

***ZEMAX***

---

**Optical Design Program**

**User's Guide**

**Version 6.0**

**ZEMAX**

---

**Optical Design Program**

**User's Guide**

**Version 6.0**

**Focus Software, Incorporated  
P. O. Box 18228  
Tucson, AZ 85731 USA  
Tel. (520) 733-0130  
Fax. (520) 733-0135**

**ZEMAX@focus-software.com  
www.focus-software.com**

## ***Important notice***

Microsoft®, Windows™, and Windows NT™ are trademarks of Microsoft Corp. ZEMAX™ is a trademark of Focus Software, Inc. Copyright © Focus Software, Incorporated 1990-1997. All rights reserved. All other product names or trademarks are property of their respective owners.

Information in this manual is subject to change without notice and does not represent a commitment on the part of the vendor. The software described in this manual is furnished under a license agreement and may be used or copied only in accordance with the terms of the agreement.

The material in this manual is copyrighted and may not be reproduced or electronically distributed without written permission from the copyright holder.

Focus Software, Incorporated (FSI) provides this publication "as is" without warranty of any kind, either express or implied, including but not limited to the implied warranties or conditions of merchantability or fitness for a particular purpose. In no event shall FSI be liable for any loss of profits, loss of business, loss of use or data, interruption of business, or for indirect, special, incidental, or consequential damages of any kind, even if FSI has been advised of the possibility of such damages arising from any defect or error in this publication or in the Software.

Published in March, 1997.

Focus Software, Incorporated  
P. O. Box 18228  
Tucson, Arizona 85731

---

# Table of Contents

---

## INTRODUCTION 1

About this document . . . . .	1-1
What does ZEMAX do? . . . . .	1-1
What doesn't ZEMAX do? . . . . .	1-1
Getting technical support . . . . .	1-2
System requirements . . . . .	1-2
Installation procedure . . . . .	1-2
Installing the hardware lock . . . . .	1-2
Running ZEMAX . . . . .	1-2
Policy on the hardware lock . . . . .	1-3
Policy on bug fixes . . . . .	1-3
Policy on upgrades . . . . .	1-3

---

## USER INTERFACE 2

Introduction . . . . .	2-1
Types of windows . . . . .	2-1
Main window operations . . . . .	2-1
Editor windows operations . . . . .	2-2
Graphic windows operations . . . . .	2-2
Using the annotation feature . . . . .	2-3
Using pan and zoom . . . . .	2-4
Text windows operations . . . . .	2-4
Dialog operations . . . . .	2-4
Aborting long computations . . . . .	2-5
Summary of useful shortcuts . . . . .	2-5
ZEMAX shortcuts . . . . .	2-5
Windows shortcuts . . . . .	2-6
Using the Windows clipboard . . . . .	2-6

---

## CONVENTIONS AND DEFINITIONS 3

Introduction . . . . .	3-1
Active configuration . . . . .	3-1
Angular magnification . . . . .	3-1
Apodization . . . . .	3-1
Cardinal planes . . . . .	3-1
Chief ray . . . . .	3-1
Coordinate axes . . . . .	3-2
Diffraction limited . . . . .	3-2
Edge thickness . . . . .	3-2
Effective focal length . . . . .	3-2
Entrance pupil diameter . . . . .	3-2
Entrance pupil position . . . . .	3-3
Exit pupil diameter . . . . .	3-3
Exit pupil position . . . . .	3-3
Extra data . . . . .	3-3

Field angles and heights . . . . .	3-3
Float by stop size . . . . .	3-3
Glasses . . . . .	3-3
Hexapolar rings . . . . .	3-3
Image space F/# . . . . .	3-4
Lens units . . . . .	3-4
Marginal ray . . . . .	3-4
Maximum field . . . . .	3-4
Non-paraxial systems . . . . .	3-4
Normalized field and pupil coordinates . . . . .	3-4
Object space numerical aperture . . . . .	3-5
Parameter data . . . . .	3-5
Paraxial and paraxial rays . . . . .	3-5
Paraxial image height . . . . .	3-6
Paraxial magnification . . . . .	3-6
Paraxial working F/# . . . . .	3-6
Primary wavelength . . . . .	3-6
Radii . . . . .	3-6
Sagittal and Tangential . . . . .	3-6
Semi-diameters . . . . .	3-6
Surface apertures . . . . .	3-7
System aperture . . . . .	3-7
Thicknesses . . . . .	3-7
Total internal reflection (TIR) . . . . .	3-7
Total track . . . . .	3-7
Vignetting factors . . . . .	3-7
Wavelength data . . . . .	3-8
Working F/# . . . . .	3-8

---

**TUTORIAL** **4**

Introduction . . . . .	4-1
Lesson 1: a singlet . . . . .	4-1
Lesson 2: a doublet . . . . .	4-5
Lesson 3: a Newtonian telescope . . . . .	4-9
Lesson 4: a Schmidt-Cassegrain with aspheric corrector . . . . .	4-12
Lesson 5: a multi-configuration laser beam expander . . . . .	4-17
Lesson 6: fold mirrors and coordinate breaks . . . . .	4-20
Lesson 7: the achromatic singlet . . . . .	4-22

---

**FILE MENU** **5**

New . . . . .	5-1
Open . . . . .	5-1
Save . . . . .	5-1
Save as . . . . .	5-1
Append . . . . .	5-1
Environment . . . . .	5-2
Address . . . . .	5-2
Directories . . . . .	5-2
Graphics . . . . .	5-2
Colors . . . . .	5-3

Text . . . . .	5-3
Editors . . . . .	5-3
Printing . . . . .	5-4
Button Bar . . . . .	5-5
Status Bar . . . . .	5-5
Exit . . . . .	5-5
Recently used files . . . . .	5-5

## EDITORS MENU

6

Lens Data . . . . .	6-1
Inserting and deleting surfaces . . . . .	6-1
Cutting, Copying, and Pasting surface data . . . . .	6-1
Entering surface comments . . . . .	6-1
Entering radii data . . . . .	6-1
Entering thickness data . . . . .	6-1
Entering glass data . . . . .	6-2
Entering semi-diameter data . . . . .	6-2
Entering conic data . . . . .	6-2
Entering parameter data . . . . .	6-2
Defining the stop surface . . . . .	6-2
Selecting surface types . . . . .	6-2
Specifying surface apertures . . . . .	6-3
Setting and removing solves . . . . .	6-3
Menu options . . . . .	6-3
Edit . . . . .	6-4
Solves . . . . .	6-4
Options . . . . .	6-4
Help . . . . .	6-4
Merit Function . . . . .	6-5
Edit . . . . .	6-5
Tools . . . . .	6-5
Help . . . . .	6-5
Multi-Configuration . . . . .	6-5
Edit . . . . .	6-6
Solves . . . . .	6-6
Tools . . . . .	6-6
Help . . . . .	6-6
Tolerance Data . . . . .	6-6
Edit . . . . .	6-6
Tools . . . . .	6-7
Help . . . . .	6-7
Extra data . . . . .	6-7
Solves . . . . .	6-7
Tools . . . . .	6-7
Help . . . . .	6-7
Undo, Redo, and Recover . . . . .	6-8
Undo: None . . . . .	6-8
Undo: Memory 1 Step . . . . .	6-8
Undo: Disk Multi Step . . . . .	6-8

## SYSTEM MENU

7

Update All . . . . .	7-1
Update . . . . .	7-1
General . . . . .	7-1
Lens Title . . . . .	7-1
Aperture Type . . . . .	7-1
Aperture Value . . . . .	7-1
Lens Units . . . . .	7-2
Glass Catalogs . . . . .	7-2
Notes . . . . .	7-2
Advanced . . . . .	7-2
Ray Aiming . . . . .	7-2
Use Ray Aiming Cache . . . . .	7-3
Pupil Shift: X, Y, and Z . . . . .	7-3
Apodization Type . . . . .	7-3
Apodization Factor . . . . .	7-4
Reference OPD . . . . .	7-4
Paraxial Rays . . . . .	7-5
Fast Asphere Trace . . . . .	7-5
Check GRIN Apertures . . . . .	7-5
Use T, P . . . . .	7-5
Temperature . . . . .	7-5
Pressure . . . . .	7-5
Semi Diameter Margin in % . . . . .	7-5
Fast Semi-Diameters . . . . .	7-6
Fields . . . . .	7-6
Wavelengths . . . . .	7-7
Next Configuration . . . . .	7-7
Last Configuration . . . . .	7-7

## ANALYSIS MENU

8

Introduction . . . . .	8-1
Layout . . . . .	8-1
2D Layout . . . . .	8-1
3D Layout . . . . .	8-2
Solid Model . . . . .	8-3
Wireframe . . . . .	8-4
Shaded Model . . . . .	8-4
Element Drawing . . . . .	8-4
Fans . . . . .	8-5
Ray Aberration . . . . .	8-5
Optical Path . . . . .	8-6
Pupil Aberration . . . . .	8-7
Spot Diagrams . . . . .	8-7
Standard . . . . .	8-7
Through Focus . . . . .	8-9
Full Field . . . . .	8-9
Matrix . . . . .	8-9
MTF . . . . .	8-10



Modulation Transfer Function (MTF)	8-10
Through Focus MTF	8-11
Surface MTF	8-11
Geometric Transfer Function	8-12
Through Focus GMTF	8-12
PSF	8-12
FFT Point Spread Function	8-12
Huygens Point Spread Function	8-15
FFT PSF Cross Section	8-15
Wavefront	8-16
Wavefront Map	8-16
Interferogram	8-17
RMS	8-17
RMS vs. Field	8-17
RMS vs. Focus	8-18
Encircled energy	8-19
Diffraction	8-19
Geometric	8-20
Line/Edge Response	8-21
Illumination	8-21
Relative illumination	8-21
Image Analysis	8-22
Vignetting Plot	8-25
Illumination XY scan	8-25
Illumination 2D surface	8-26
Miscellaneous	8-26
Field Curvature/Distortion	8-26
Grid Distortion	8-27
Longitudinal Aberration	8-29
Lateral Color	8-29
Y-Ybar Diagram	8-29
Chromatic Focal Shift	8-30
Dispersion Diagram	8-30
Internal Transmittance vs. Wavelength	8-30
Glass Map	8-31
Test Plate Lists	8-31
Calculations	8-31
Ray Trace	8-31
Gaussian Beam	8-32
Seidel Coefficients	8-34
Zernike Coefficients	8-35
YNI Contributions	8-37
Sag Table	8-37
Gradient Index	8-38
Gradium™ Profile	8-38
Polarization	8-38
Polarization Ray Trace	8-39
Polarization Pupil Map	8-39
Transmission	8-39
Phase Aberration	8-40
Transmission Fan	8-40
Coatings	8-41

Reflection vs. Angle . . . . .	8-41
Transmission vs. Angle . . . . .	8-41
Absorption vs. Angle . . . . .	8-41
Diattenuation vs. Angle . . . . .	8-42
Phase vs. Angle . . . . .	8-42
Retardance vs. Angle . . . . .	8-42
Reflection vs. Wavelength . . . . .	8-42
Transmission vs. Wavelength . . . . .	8-43
Absorption vs. Wavelength . . . . .	8-43
Diattenuation vs. Wavelength . . . . .	8-43
Phase vs. Wavelength . . . . .	8-43
Retardance vs. Wavelength . . . . .	8-44

## TOOLS MENU

9

Optimization . . . . .	9-1
Global Search . . . . .	9-1
Hammer Optimization . . . . .	9-1
Tolerancing . . . . .	9-1
Glass Catalogs . . . . .	9-1
Lens Catalogs . . . . .	9-2
Performance Test . . . . .	9-3
Reverse Elements . . . . .	9-3
Scale Lens . . . . .	9-3
Make Focal . . . . .	9-3
Ghost Focus Generator . . . . .	9-3
Test Plate Fitting . . . . .	9-4
ZPL Macros . . . . .	9-5
Remove All Variables . . . . .	9-6
Edit Coating File . . . . .	9-6
Quick Focus . . . . .	9-6
Add Fold Mirror . . . . .	9-6

## REPORTS MENU

10

Introduction . . . . .	10-1
System Data . . . . .	10-1
Surface Data . . . . .	10-1
Prescription Data . . . . .	10-2
Merit Function Listing . . . . .	10-2
Tolerance Listing . . . . .	10-2
Coating Listing . . . . .	10-3

## SURFACE TYPES

11

Introduction . . . . .	11-1
Parameter data . . . . .	11-1
Extra data . . . . .	11-1
Summary of surface types . . . . .	11-1
The standard surface . . . . .	11-3
The even asphere surface . . . . .	11-3
The odd asphere surface . . . . .	11-4

The paraxial surface . . . . .	11-4
The paraxial XY surface . . . . .	11-4
The toroidal surface . . . . .	11-5
The biconic surface . . . . .	11-5
The toroidal grating surface . . . . .	11-6
The cubic spline surface . . . . .	11-6
The hologram 1 surface . . . . .	11-7
The hologram 2 surface . . . . .	11-7
The coordinate break surface . . . . .	11-8
The polynomial surface . . . . .	11-8
The Fresnel surface . . . . .	11-9
The ABCD surface . . . . .	11-9
The alternate surface . . . . .	11-9
The diffraction grating surface . . . . .	11-10
The conjugate surface . . . . .	11-10
The tilted surface . . . . .	11-11
The irregular surface . . . . .	11-11
The gradient 1 surface . . . . .	11-12
Discussion on maximum step size for GRIN surfaces . . . . .	11-12
Restrictions on surfaces following GRIN surfaces . . . . .	11-12
The gradient 2 surface . . . . .	11-12
The gradient 3 surface . . . . .	11-13
The gradient 4 surface . . . . .	11-13
The gradient 5 surface . . . . .	11-13
The gradient 6 surface . . . . .	11-14
The gradient 7 surface . . . . .	11-15
The GRADIUM™ surface . . . . .	11-16
The gradient 9 surface . . . . .	11-17
The gradient 10 surface . . . . .	11-18
The Zernike surface . . . . .	11-18
The Zernike phase surface . . . . .	11-19
Zernike phase coefficients sign conventions . . . . .	11-20
The extended polynomial surface . . . . .	11-20
The binary optic 1 surface . . . . .	11-21
Binary optic coefficients sign conventions . . . . .	11-22
The binary optic 2 surface . . . . .	11-22
Binary optic coefficients sign conventions . . . . .	11-24
The extended cubic spline surface . . . . .	11-24
The extended asphere surface . . . . .	11-24
The extended odd asphere surface . . . . .	11-25
The variable line space grating surface . . . . .	11-26
The elliptical grating surface . . . . .	11-26
The superconic surface . . . . .	11-27
The extended Fresnel surface . . . . .	11-28
The grid sag surface . . . . .	11-29
The grid phase surface . . . . .	11-30
Phase coefficients sign conventions . . . . .	11-30
The generalized Fresnel surface . . . . .	11-30
The periodic surface . . . . .	11-31
The toroidal hologram surface . . . . .	11-31
The Jones matrix surface . . . . .	11-32
The atmospheric refraction surface . . . . .	11-33

The zone plate surface . . . . .	11-33
The user defined surface . . . . .	11-34
The UDS DLL . . . . .	11-35
Comments about refractive and reflective UDS DLL's . . . . .	11-35
Comments about gradient index UDS DLL's . . . . .	11-35
Comments about diffractive UDS DLL's . . . . .	11-35
Lenslet arrays using UDS DLL's . . . . .	11-36
Error handling and UDS's . . . . .	11-36

---

## SOLVES 12

Introduction . . . . .	12-1
Curvature : Marginal ray angle . . . . .	12-2
Curvature : Chief ray angle . . . . .	12-2
Curvature : Pick up . . . . .	12-2
Curvature : Marginal ray normal . . . . .	12-2
Curvature : Chief ray normal . . . . .	12-2
Curvature : Aplanatic . . . . .	12-2
Curvature : Element power . . . . .	12-2
Curvature : Concentric with surface . . . . .	12-3
Curvature : Concentric with radius . . . . .	12-3
Curvature : F number . . . . .	12-3
Thickness : Marginal ray height . . . . .	12-3
Thickness : Chief ray height . . . . .	12-3
Thickness : Edge thickness . . . . .	12-3
Thickness : Pick up . . . . .	12-4
Thickness : Optical path difference . . . . .	12-4
Thickness : Position . . . . .	12-4
Thickness : Compensator . . . . .	12-4
Thickness : Center of curvature . . . . .	12-4
Glass : Model . . . . .	12-4
Glass : Pick up . . . . .	12-4
Glass : Substitute . . . . .	12-4
Semi-Diameter : Pick up . . . . .	12-5
Semi-Diameter : Maximum . . . . .	12-5
Conic : Pick up . . . . .	12-5
Parameter : Pick up . . . . .	12-5
Suggestions for use . . . . .	12-5

---

## OPTIMIZATION 13

Introduction . . . . .	13-1
Selecting variables . . . . .	13-1
Defining the default merit function . . . . .	13-1
Selecting the type of optimization . . . . .	13-1
Physically significant merit functions . . . . .	13-2
Selecting the pupil integration method . . . . .	13-3
Rings . . . . .	13-3
Arms . . . . .	13-3
Grid . . . . .	13-4
Delete Vignetted . . . . .	13-4
Setting Boundary values . . . . .	13-4

Start At . . . . .	13-4
Assume Axial Symmetry . . . . .	13-4
Ignore Lateral Color . . . . .	13-5
Pitfalls with the default merit function . . . . .	13-5
Optimization with apodized beams . . . . .	13-5
Modifying the merit function . . . . .	13-5
Understanding boundary operands . . . . .	13-20
Using MTF operands . . . . .	13-21
Performing an optimization . . . . .	13-22
Defining complex operands . . . . .	13-22
Optimizing glass selection . . . . .	13-23
Optimizing zoom and multi-configuration lenses . . . . .	13-25
Optimizing extra data . . . . .	13-25
Using gradient index operands . . . . .	13-25
DLTN . . . . .	13-25
LPTD . . . . .	13-25
Optimizing with ZPL macros . . . . .	13-26
Suggestions for use . . . . .	13-27

---

## GLOBAL OPTIMIZATION 14

Introduction . . . . .	14-1
Capabilities of ZEMAX . . . . .	14-1
The global search algorithm . . . . .	14-2
The hammer algorithm . . . . .	14-2
Optimizing glass selection . . . . .	14-3
Suggestions for use . . . . .	14-3
Summary . . . . .	14-4

---

## TOLERANCING 15

Introduction . . . . .	15-1
The basic procedure . . . . .	15-1
Tolerance operands . . . . .	15-2
TRAD: Tolerance on radius . . . . .	15-2
TCUR: Tolerance on curvature . . . . .	15-3
TFRN: Tolerance on fringes . . . . .	15-3
TTHI: Tolerance on thickness . . . . .	15-3
TCON: Tolerance on conic . . . . .	15-3
TSDX, TSDY: Tolerance on surface decenters . . . . .	15-3
TSTX, TSTY: Tolerance on surface tilts . . . . .	15-4
TIRX, TIRY: Tolerance on surface TIR . . . . .	15-4
TIRR: Tolerance on surface irregularity . . . . .	15-4
TPAR: Tolerance on parameter data . . . . .	15-5
TEDV: Tolerance on Extra Data Values . . . . .	15-5
TIND: Tolerance on Index . . . . .	15-5
TEDX, TEDY: Tolerance on element decenters . . . . .	15-5
TETX, TETY: Tolerance on element tilts . . . . .	15-5
TUDX, TUDY, TUTX, TUTY, TUTZ: User Defined Tilts/Decenters . . . . .	15-6
Tolerancing with the Irregular surface type . . . . .	15-6
Tolerance control operands . . . . .	15-7
COMP: Define compensator . . . . .	15-7

CPAR: Define parameter compensator . . . . .	15-7
SAVE: Save sensitivity analysis lenses . . . . .	15-7
STAT: Define statistics . . . . .	15-8
Defining default tolerances . . . . .	15-8
Surface tolerances . . . . .	15-8
Element tolerances . . . . .	15-8
Defining compensators . . . . .	15-9
Performing the tolerance analysis . . . . .	15-9
How ZEMAX computes the tolerance analysis . . . . .	15-12
Fast Tolerancing Mode . . . . .	15-12
Sensitivity analysis . . . . .	15-12
Inverse sensitivity . . . . .	15-13
Monte Carlo . . . . .	15-14
Tolerancing multi-configuration (zoom) lenses . . . . .	15-14
Tolerancing with solves . . . . .	15-15
Trouble shooting the tolerance results . . . . .	15-15
Pitfalls when tolerancing . . . . .	15-15
Summary . . . . .	15-15

---

## MULTI-CONFIGURATIONS 16

Introduction . . . . .	16-1
The first step . . . . .	16-1
Defining the number of configurations . . . . .	16-2
Defining each configuration . . . . .	16-3
Adding and removing elements . . . . .	16-3
Changing configurations . . . . .	16-3
Optimization with multi configurations . . . . .	16-3
Using solves with multi-configuration data . . . . .	16-3

---

## USING GLASS CATALOGS 17

Introduction . . . . .	17-1
Specifying which glass catalogs to use . . . . .	17-1
Loading a glass catalog . . . . .	17-1
Description of catalog data . . . . .	17-1
Creating a new catalog . . . . .	17-2
The glass dispersion formulas . . . . .	17-2
Fitting index data . . . . .	17-3
Defining Transmission Data . . . . .	17-4
Other glass catalog options . . . . .	17-4
Using MIL number glasses . . . . .	17-4
Glass catalog sources . . . . .	17-4
Obsolete catalog data . . . . .	17-6

---

## THERMAL ANALYSIS 18

Introduction . . . . .	18-1
Defining temperature and pressure . . . . .	18-1
Index of refraction computation . . . . .	18-1
Defining multiple temperature and pressure values . . . . .	18-2
Defining which parameters consider thermal effects . . . . .	18-3

Radius of curvature (CRVT) values . . . . .	18-3
Thickness (THIC) values . . . . .	18-3
Parameter values . . . . .	18-3
Extra data values . . . . .	18-4
All other values . . . . .	18-4
Automatic thermal setup . . . . .	18-4
Adding TCE data . . . . .	18-4
Adding thermal index variation data . . . . .	18-4
Optimizing athermal lenses . . . . .	18-4
Limitations of thermal analysis . . . . .	18-5

---

## POLARIZATION ANALYSIS 19

Introduction . . . . .	19-1
Review of polarization concepts . . . . .	19-1
Reflectance Amplitude, $\rho$ . . . . .	19-1
Transmittance Amplitude, $\tau$ . . . . .	19-2
Reflectance Intensity, R . . . . .	19-2
Transmittance Intensity, T . . . . .	19-2
Absorption Intensity, A . . . . .	19-2
Diattenuation, D . . . . .	19-2
Phase, P . . . . .	19-2
Retardance, S . . . . .	19-2
Coating description shorthand . . . . .	19-2
Properties of uncoated surfaces . . . . .	19-2
Defining coatings in ZEMAX . . . . .	19-3
Default materials and coatings supplied with ZEMAX . . . . .	19-4
Specifying coatings on surfaces . . . . .	19-5
What ZEMAX does if no coating is specified . . . . .	19-5
Defining the incident polarization . . . . .	19-5
Defining polarizing components . . . . .	19-6
What ZEMAX can compute using polarization analysis . . . . .	19-7
Modeling frustrated total internal reflection . . . . .	19-7
Limitations of polarization analysis . . . . .	19-8

---

## ZEMAX PROGRAMMING LANGUAGE 20

Introduction . . . . .	20-1
Creating ZPL programs . . . . .	20-1
Running ZPL programs . . . . .	20-1
An overview of ZPL . . . . .	20-2
ZPL variables . . . . .	20-2
ZPL functions . . . . .	20-3
ZPL math operations . . . . .	20-7
ZPL logical operators . . . . .	20-7
ZPL string variables and operations . . . . .	20-8
ZPL keywords . . . . .	20-9
APMN, APMX . . . . .	20-9
APXD, APYD . . . . .	20-10
AFTP . . . . .	20-10
ATYP, AVAL . . . . .	20-11

CLOSE	20-11
CLOSEWINDOW	20-11
COLOR	20-11
CONI	20-12
CURV	20-12
DELETE	20-12
EDVA	20-13
END	20-13
FINDFILE	20-13
FLDX, FLDY, FWGT, FVDX, FVDY, FVCX, FVCY	20-14
FOR, NEXT	20-14
FORMAT	20-15
FTYP	20-15
GDATE	20-16
GETEXTRADATA	20-16
GETGLASSDATA	20-16
GETMTF	20-17
GETSYSTEMDATA	20-17
GETVARDATA	20-18
GETZERNIKE	20-19
GLAS	20-20
GLENSNAME	20-20
GOSUB, SUB, RETURN, and END	20-20
GOTO	20-21
GRAPHICS	20-21
GTEXT	20-22
GTEXTCENT	20-23
GTITLE	20-23
IF-THEN-ELSE-ENDIF	20-23
INPUT	20-24
INSERT	20-24
LABEL	20-25
LINE	20-25
LOADLENS	20-25
LOADMERIT	20-26
NEXT	20-26
NUMFIELD	20-26
NUMWAVE	20-26
OPEN	20-26
OPTIMIZE	20-27
OPTRETURN	20-27
OUTPUT	20-27
PARn	20-28
PARAXIAL	20-28
PAUSE	20-29
PIXEL	20-29
POLDEFINE	20-29
POLTRACE	20-30
PRINT	20-30
PRINTFILE	20-31
PRINTWINDOW	20-31
RADI	20-32



RAYTRACE . . . . .	20-32
READ . . . . .	20-33
READSTRING . . . . .	20-33
REM, ! . . . . .	20-33
RETURN . . . . .	20-34
SAVELENS . . . . .	20-34
SDIA . . . . .	20-34
SETAIM . . . . .	20-35
SETCONFIG . . . . .	20-35
SETTEXTSIZE . . . . .	20-35
SETTITLE . . . . .	20-36
SETUNITS . . . . .	20-36
SETVAR . . . . .	20-36
SETVECSIZE . . . . .	20-36
SETVIG . . . . .	20-37
SHOWFILE . . . . .	20-37
SOLVETYPE . . . . .	20-37
STOPSURF . . . . .	20-38
SUB . . . . .	20-38
SURFTYPE . . . . .	20-38
THIC . . . . .	20-39
TIMER . . . . .	20-39
UPDATE . . . . .	20-40
VEC1, VEC2, VEC3, VEC4 . . . . .	20-40
WAVL, WWGT . . . . .	20-40
Example 1 . . . . .	20-41
Example 2 . . . . .	20-42

---

## ADVANCED TOPICS 21

Introduction . . . . .	21-1
Simulating laser diode sources . . . . .	21-1
Simulating an elliptical source . . . . .	21-3
Modeling axicons . . . . .	21-3
Eliminating variables in zoom systems . . . . .	21-3

---

## COMMON QUESTIONS ABOUT ZEMAX 22

Introduction . . . . .	22-1
Questions about installation . . . . .	22-1
Questions about setting up lenses . . . . .	22-1
Questions about lens analysis . . . . .	22-1
Questions about optimization . . . . .	22-2

---

## SUPPORT FILES 23

Introduction . . . . .	23-1
AGF glass catalog files . . . . .	23-1
C language files . . . . .	23-2
Image files . . . . .	23-2
Note files . . . . .	23-3
Test plate fit files . . . . .	23-3

ZMF stock lens files . . . . .	23-3
ZEMAX example lens files . . . . .	23-4
ZPL example files . . . . .	23-6

---

**ERROR MESSAGES** **24**

Introduction . . . . .	24-1
------------------------	------

---

**THE ZMX FILE FORMAT** **25**

Introduction . . . . .	25-1
ZMX commands . . . . .	25-1
ZMX format for optimization merit functions . . . . .	25-3
ZMX format for multi-configuration operands . . . . .	25-3
Minimum ZMX file . . . . .	25-3
Surface type codes . . . . .	25-4
Additional examples . . . . .	25-4

# INTRODUCTION

## About this document

ZEMAX is available in three different editions: ZEMAX-SE (Standard Edition), ZEMAX-XE (Extended Edition), and ZEMAX-EE (Engineering Edition). This manual covers all three editions, however certain features are unique to one or two editions, as noted in the text. If a feature is available in both ZEMAX-XE and ZEMAX-EE, but not ZEMAX-SE, the following banner will usually be shown in the manual where the feature is described:

This feature is only available in the XE and EE editions of ZEMAX.

If a feature is available in ZEMAX-EE, but not in ZEMAX-XE or ZEMAX-SE, the following banner will usually be shown in the manual where the feature is described:

This feature is only available in the EE edition of ZEMAX.

Note that XE is a superset of SE; that is XE does everything SE does, plus additional features. Also, EE is a superset of XE. This manual covers all three editions of ZEMAX running on Microsoft Windows and Windows NT operating systems.

## What does ZEMAX do?

ZEMAX is a program which can model, analyze, and assist in the design of optical systems. The interface to ZEMAX has been designed to be simple to use, and with a little practice it can allow very rapid interactive design. Most ZEMAX features are accessed by selecting options from either pop-up or pull-down menus. Keyboard shortcuts are provided for quickly navigating or bypassing the menu structure. This manual provides explanations of the conventions used in ZEMAX, descriptions of procedures, and a description of available features.

## What doesn't ZEMAX do?

Neither the ZEMAX program nor the ZEMAX documentation will teach you how to design lenses or optical systems. Although the program will do many things to assist you in designing and analyzing optical systems, you are still the designer. The ZEMAX documentation is not a tutorial on optical design, terminology, or methodology. Technical support available to ZEMAX users includes assistance in using the program, but does not include tutoring on fundamental optical design principles. If you have little or no experience in optical design, you may want to read up on any of the many good books available on the subject. The following table lists some (but by no means all) of the books which will aid in your education.

### REFERENCES ON LENS DESIGN

Author	Title	Publisher
Bass	Handbook of Optics	McGraw-Hill
Hecht	Optics	Addison Wesley
Kingslake	Lens Design Fundamentals	Academic Press
Laikin	Lens Design, Second Edition	Marcel Dekker
Mahajan	Aberration Theory Made Simple	SPIE Optical Engineering Press
O' Shea	Elements of Modern Optical Design	John Wiley and Sons
Rutten and van Venrooij	Telescope Optics	Willmann-Bell
Smith	Modern Optical Engineering	McGraw-Hill
Smith	Modern Lens Design	McGraw-Hill
Welford	Aberrations of Optical Systems	Adam Hilger Ltd.
Welford	Useful Optics	University of Chicago Press

Most importantly, ZEMAX is no substitute for good engineering practices. No design should ever be considered finished until a qualified engineer has checked the calculations performed by the software to see if the results are

reasonable. This is particularly important when a design is to be fabricated and significant costs are involved. It is the engineers responsibility to check the results of ZEMAX, not the other way around.

## **Getting technical support**

---

If you have any questions about installing or using ZEMAX, please try the following suggestions to find the information you require:

- 1) Look in the Table of Contents to see if there is a chapter or section on the subject.
- 2) Check the index (at the back of this document) to see if it is referenced.
- 3) Look in the chapter "Common questions about ZEMAX". Several categories of commonly asked questions can be found there.
- 4) Look in the chapter "The ZEMAX example files" for a sample lens of the type you are trying to set up, if appropriate. If your question involves setting up a tilted component system, there are several such examples included in the directory in which ZEMAX was installed.

If you still cannot find the information you require, you may need to call, FAX, E-Mail, or write for technical support. The phone numbers and address are on the title page of this manual. If you telephone, try to be in front of your computer when you call. Please note that technical support is available for only 90 days after your initial purchase of ZEMAX. After the 90 day period, technical support and upgrades may be purchased on an annual basis.

Technical support is defined as any question that requires an engineer to resolve. This includes answering telephone calls, E-Mail, faxes, or letters regarding how to use ZEMAX, ZEMAX features, problems with your printer or network, or reporting suspected bugs.

If you think you have found a bug, read the section "Policy on bug fixes" later in this chapter.

## **System requirements**

---

ZEMAX requires the current version of Windows (which is Windows 95 as of this writing) or Windows NT, 16 Mb of RAM (32 Mb or more preferred), 40 Mb (or more) of free hard disk space, and a CD-ROM drive. ZEMAX is available for Intel and DEC Alpha based computers.

## **Installation procedure**

---

Before you install the ZEMAX software, please verify that you have the required hardware, defined in the previous section.

### ***Installing the hardware lock***

ZEMAX is copy-protected using a hardware device called a hardware lock. The hardware lock only permits one copy of ZEMAX to run at a time. Attempting to run ZEMAX on more than one computer at a time, or attempting to defeat the intention of the hardware lock is a violation of the owner's copyright, and is punishable by severe criminal and civil penalties.

Attach the hardware lock supplied with the program on the parallel printer port at the rear of the computer. If you do not have a printer, place the hardware lock in the port by itself. If you have a printer, disconnect the printer cable, then attach the hardware lock to the port, then attach the printer cable to the hardware lock. Your printer, and all other computer operations, will be unaffected by the presence of the hardware lock.

You do not need to hook up a printer to the hardware lock, but if you do, the printer must be turned on. If you do not have the hardware lock installed correctly, ZEMAX will issue a error message and not allow the program to run. If you experience difficulty, see the chapter "Common questions about ZEMAX".

### ***Running ZEMAX***

The ZEMAX installation program will create a new program group. To run ZEMAX, double click on the ZEMAX icon within this program group.

## **Policy on the hardware lock**

---

ZEMAX is multiple-use protected by a device called a hardware lock. The hardware lock must be attached to the computer on which ZEMAX is running. The hardware lock prevents more than one copy of ZEMAX from running at one time. The registered user of the ZEMAX software is allowed to install the ZEMAX software on more than one computer, as long as there is no possibility that more than one copy of ZEMAX will be running at one time. For example, the user is allowed to install the ZEMAX software on a computer at work and another computer at home, as long as the two installations of ZEMAX are never used at the same time. The hardware lock must be physically moved from one computer to another, so that only one copy of ZEMAX can run at one time.

The hardware lock supplied with the ZEMAX software is worth the full purchase price of the software. If the hardware lock is lost or stolen, it will not be replaced without payment of the full purchase price. Insure the hardware lock as you would any other business or personal asset of comparable value.

If the hardware lock is lost or stolen, it will not be replaced without payment of the full purchase price.
---

If the hardware lock ceases to function correctly, it will be replaced free of charge if returned to Focus Software. Please call or fax for authorization prior to returning any defective hardware lock.

## **Policy on bug fixes**

---

All non-trivial computer programs have bugs. Every effort is made to find and fix all bugs before a new version is released. However, the program is sufficiently complex such that even a talented team of pre-release program testers will not find every bug. Focus Software therefore may provide periodic bug fix releases.

If you find an apparent bug, please report it to us. Try to isolate the exact actions which cause or lead up to the bug. Try to see if the error is unique to the lens file you are working with. Bugs are usually easier to find if they also appear on the example files provided with the program.

If you find a bug which causes significant degradation of program performance within 60 days of purchasing the program, the bug fix will be provided free of charge. Focus Software reserves the right to determine what bugs constitute significant degradation. After 60 days, an upgrade charge may apply; in this case you will receive the latest program version.

Bug fixes will not be provided if the problem is traced to incompatible, non-standard, or insufficient hardware, or if you are using an outdated version of the operating system.

Please also see the section "What doesn't ZEMAX do?" earlier in this chapter.

## **Policy on upgrades**

---

Periodically you may receive an upgrade notification which will list the new features available. Upgrades are only sold as part of an annual technical support and upgrade package.

This page intentionally left blank.

## Introduction

---

This chapter describes the conventions used by the ZEMAX user interface, and describes some keyboard shortcuts for common window operations. ZEMAX is easy to use once you learn the simple conventions used throughout the program. For step by step sample sessions using ZEMAX, see the "Tutorial" chapter.

## Types of windows

---

ZEMAX has different types of windows, each of which serves a different purpose. The types of windows are:

**The main window:** This window has a large blank area, with a title bar, menu bar, and toolbar at the top. The commands available on this menu bar apply in general to the current optical system as a whole.

**Editor windows:** There are five different editors: the Lens Data Editor, the Merit Function Editor, the Multi-Configuration Editor, the Tolerance Data editor, and (in ZEMAX-EE only) the Extra Data Editor.

**Graphic windows:** These windows are used to display graphic data such as layouts, ray fans, and MTF plots.

**Text windows:** Text windows are used to display text data such as prescription data, aberration coefficients, and numerical data.

**Dialogs.** Dialogs are pop-up windows which cannot be resized. Dialogs are used to change options or data such as field angles, wavelengths, apertures, and surface types. They are also used extensively to change options on graphic and text windows, such as for changing the number of rays on a layout plot.

All windows can be moved or resized (except dialogs) using the standard mouse or keyboard commands. If you are not familiar with these procedures, see any good book on using Windows or the Windows documentation.

## Main window operations

---

The main window bar has several menu headings. Most menu headings correspond to a chapter of the same name later in this manual. See the specific chapter for detailed instructions on using the features available from each menu option. These are the menu bar headings:

**File:** Used to open, close, save, and rename (save as) lens files.

**Editors:** Used to invoke (make appear) any of the editor windows.

**Analysis:** Analysis groups together features which do not change the lens data, but compute numerical or graphical results from the lens data. These features include layouts, ray fans, spot diagrams, diffraction calculations, and more.

**Tools:** Tools are features which can change the lens data or perform advanced computations on the system as a whole. These include optimization, tolerancing, test plate fitting, macro language execution, and more.

**Reports:** Reports are used to document a lens design. These features include system data summaries and surface data summaries.

**Help:** Provides access to the online documentation.

Most of the commonly used menu options have keyboard equivalents which may be faster to use. For example, typing Ctrl-Q will quit ZEMAX. The keyboard shortcuts are listed along side the menu options.

A handy shortcut key for switching between windows within the main window is Control-Tab. This will advance control to the next window in the list of windows maintained by ZEMAX.

The main window also displays a row of buttons just below the menu bar. This row of buttons is called a "toolbar". The toolbar buttons are used to quickly select common operations. All of the buttons represent functions available on the menus. The button definitions may be changed on the File, Environment, Button Bar menu option. The three letter mnemonics used by the buttons are defined on that environment window. If the screen resolution is not sufficiently high, then not all of the buttons will appear. A screen resolution of 1024 x 768 or higher is recommended.

## Editor windows operations

---

The editor windows are used primarily for entering lens and merit function data. Each editor is similar to a spreadsheet, with rows and columns. The intersection of a row and a column forms a cell. If the editor is the active window (the one with a highlighted title bar) then one cell will be shown highlighted or in reverse colors. This cell is called the active cell, and it has what is called the input "focus". The inverted color of the cell is called the cursor, although it is not a cursor in the usual sense.

Having the input focus means that any data typed on the keyboard will be sent to the active cell. The exceptions are control commands such as cursor keys or control key combinations, which are sent directly to the main window. To modify the data in the active cell, simply type in the new data and press the Enter key when finished.

To add an incremental value to a cell, type a plus sign and then the increment, then Enter. For example, to change a 12 to 17, type "+5" and Enter. The "\*" multiply and "/" divide symbols also work. To subtract a value, type a minus sign and a space followed by the value to subtract. The space is required to distinguish between subtraction and simply entering a negative number.

To modify a portion of the contents of a cell without retyping the entire value, first highlight the cell, then press the back space key. The left cursor, right cursor, home, and end keys can then be used to navigate within the cell for editing. The mouse may also be used to select and replace portions of text. Once changes have been made to the data in the cell, pressing Enter will complete editing and leave the cursor on that cell. Pressing up or down cursors will also complete editing, and move the cursor accordingly. Pressing Tab or Shift-Tab will complete editing and move the cursor right or left.

To abort editing of any cell, press the escape key.

The left, right, up and down cursor keys will move the cursor accordingly. Pressing the control key and the left, right, up and down keys simultaneously will move the editor display one page at a time in the appropriate direction. The Tab and Shift-Tab keys also move the cursor right and left.

The page up and page down keys move the cursor one screen at a time. Control-page up and control-page down move the cursor to the top and bottom of the current column. The home and end keys will move the cursor to the first column first row and first column last row, respectively. Control-home and control-end will move the cursor to first and last row of the last column, respectively.

Clicking once in any cell will move the cursor to that cell. Double clicking on a cell will invoke the solve dialog box for that cell if one exists. Right mouse clicking will also invoke the solve dialog box for that cell if one exists.

## Graphic windows operations

---

Graphic windows have the following menu items:

Update: This recomputes the data displayed in the window with the current settings.

Settings: Invokes a dialog box which controls the options for that window.

Text: Displays a text listing of the data in a new window. Not all graphics windows support this option.

Print: Prints the contents of the window.

Window: There are three submenus under Window:

Copy Clipboard: Copies the window contents to the Windows clipboard. See a following section for details.

Export Metafile: Exports the displayed graphic as a Windows Metafile. The WMF format is used for 16-bit applications, while 32-bit applications use the "enhanced" metafile format, EMF. Metafiles are widely supported as a means for exchanging graphics among Windows applications. This option is particularly useful for transferring graphics from ZEMAX to other applications when the use of the Clipboard fails to work.

Lock Window: If selected, the window will be converted into a "static" window whose data cannot change. The locked window contents may be printed, copied to the clipboard, or saved in a file. The application for this feature is for comparing the results of different lens files. Once a window is locked, it cannot be updated, and so any new lens files which are subsequently loaded may be analyzed and compared to the locked



window results. Once the window is locked, it cannot be unlocked. To recompute the data in the window, the window must be closed and another window opened.

Annotate: See the section "Using the annotation feature" for details. There are four submenus under Annotate:

Line: Draws a single line on the graphic window.

Text: Prompts for and then draws text on the graphic window.

Box: Draws a box on the graphic window.

Edit: Allows extended editing of annotations.

Unzoom: Restores a previously panned and zoomed graphic to the full size. See "Using pan and zoom" below.

There are two mouse shortcuts available when using graphic windows:

Double clicking anywhere in the body of graphic window will update the contents. This is the same as selecting Update.

Clicking with the right mouse button anywhere in the body of a graphic window will invoke the Settings dialog box.

### **Using the annotation feature**

Selecting Annotate, Edit from the graphic menu will invoke the annotation editor, which consists of a simple text editor and several buttons. There is also a single checkbox for enabling or disabling annotations for the graph.

The text edit field is used to define the annotations to be applied to the graph. To insert a new line, use the keyboard command Ctrl-Enter.

There are several supported commands, each with a specific syntax:

```
TEXT "string" x y angle fontx fonty
```

The TEXT command will write any text within the double quotes at the location specified by x and y, at an angle in degrees given by angle, using a fixed font whose width and height are given by fontx and fonty. The coordinates are in normalized units, where the left edge of the graph has a coordinate of x = 0.0, the right edge x = 100.0, the bottom edge is y = 0.0, and the top edge is y = 100.0. The origin is the lower left hand corner of the screen. The "angle" value is in degrees. The fontx and fonty units are in arbitrary units. The angle, fontx, and fonty values may all be left undefined, and then default values will be used.

```
LINE x1 y1 x2 y2
```

The LINE command draws a straight line from x1, y1 to x2, y2. The units and coordinate system are as described for x and y in the description of the TEXT command.

```
BOX x1 y1 x2 y2
```

The BOX command draws a box with opposite corners from x1, y1 to x2, y2. The units and coordinate system are as described for x and y in the description of the TEXT command.

```
ELLIPSE x y rx ry
```

ELLIPSE draws an ellipse centered at x and y with x half width of rx and half height of ry. If ry is the same as rx or if ry is omitted, a circle of radius rx results.

There are also several buttons on the annotation dialog box:

OK: Accepts the annotations as displayed and exits.

Cancel: Reverts back to the last annotations and exits.

Save: Opens a "Save As" type box where the file name may be specified. The annotations are saved in the user named file.

Load: Opens a "Load: type box where the file name to load may be selected. The loaded file contains the annotations to use.

Reset: Clears the edit buffer.

Help: Invokes the online help function.

### **Using pan and zoom**

Any graphic window may be panned (scrolled left, right, up and down) or zoomed (magnified). To activate the pan and zoom, choose any ZEMAX graphic window, then click the left mouse button and hold the button down for 1/2 second anywhere in the window. The cursor will change from an arrow to a cross. Now drag the mouse down and right to define a rectangle of the desired size which covers the area to zoom in on. Now, let go of the left mouse button. The selected area will be magnified to fill the window, while maintaining the aspect ratio of the plot.

To pan, drag the scroll bars on the sides of the window. Panning is only allowed after a graphic has been zoomed.

To restore the graphic to the original size, select Unzoom from the Graphic window menu bar.

## **Text windows operations**

---

Text windows have the following menu items:

Update: This recomputes the data displayed in the window with the current settings.

Settings: Invokes a dialog box which controls the options for that window.

Print: Prints the contents of the window.

Window: There are three submenus under Window:

Copy Clipboard: Copies the window contents to the Windows clipboard. See a following section for details.

Save Text: Saves the displayed text data in an ASCII file.

Lock Window: If selected, the window will be converted into a "static" window whose data cannot change. The locked window contents may be printed, copied to the clipboard, or saved in a file. The application for this feature is for comparing the results of different lens files. Once a window is locked, it cannot be updated, and so any new lens files which are subsequently loaded may be analyzed and compared to the locked window results. Once the window is locked, it cannot be unlocked. To recompute the data in the window, the window must be closed and another window opened.

There are two mouse shortcuts available when using text windows:

Double clicking anywhere in the body of graphic window will update the contents. This is the same as selecting Update.

Clicking with the right mouse button anywhere in the body of a graphic window will invoke the Settings dialog box.

## **Dialog operations**

---

Most dialog boxes are self explanatory. Usually there are buttons for OK and Cancel which are common in Windows dialog boxes.

Analysis features, such as ray fan plots, have dialog boxes which allow selection of various options. All of these dialog boxes have six buttons:

OK: Causes the window to recompute and redisplay the data with the currently selected options.

Cancel: Reverts to those options selected before the dialog was invoked, and does not recompute the data.

Save: Saves the currently selected options for use as the default settings in the future.

Load: Loads the default values that were previously saved.

Reset: Resets the default values to the "factory default" settings.

Help: Calls the ZEMAX help system. The help page displayed will contain information about the options on the active dialog box.

The other options that appear on dialog boxes can be selected using either the keyboard or the mouse. For keyboard control, use the Tab and Shift-Tab key combinations to move from option to option. The space bar can be used to toggle the currently selected checkbox. The cursor keys can be used select items within a drop down list. Pressing the first letter of an entry in a drop down box will also select that entry.

## **Aborting long computations**

Certain ZEMAX tools may require relatively long computation times. For example, the optimization, global optimization, and tolerancing tools may run from several seconds to many days. To terminate the execution of these tools, there is a "Terminate" button displayed which can be clicked. After the terminate button is clicked, ZEMAX gracefully exits the computation and returns control to the main program. Usually, the results of the computation are not available, and are not displayed.

Some analysis features, such as MTF and the image analysis feature, will run for long periods of time in some circumstances. For example, large MTF grids or large image analysis ray densities require longer computation times. However, analysis features do not display a status box or terminate button, because analysis features display their output directly in a window. For this reason, the keyboard command "Escape" is used to terminate lengthy analysis computations. There is no mouse equivalent for this feature; only the Escape key is used.

The Escape key will terminate MTF, PSF, encircled energy, and other diffraction computations. If the Escape key is pressed, control will return to the main program (it may take 1 or 2 seconds) and the data displayed in the window will be invalid. For the image analysis feature, the escape key will terminate the tracing of new rays, however, rays that have already been traced will be displayed, and data from those rays is accurate, if incomplete.

## **Summary of useful shortcuts**

This section summarizes useful shortcuts, using both the keyboard and the mouse.

### **ZEMAX shortcuts**

Action	Result
Ctrl + Tab	Moves the input focus from window to window.
Ctrl + letter	Shortcut for many ZEMAX tools and functions. For example, Ctrl-L invokes the 2D layout plot. All the keyboard shortcuts are listed alongside the menu items.
F1..F10	Function keys are also used as shortcuts for many functions. All the keyboard shortcuts are listed alongside the menu items.
Backspace	When an editor window has the input focus, the highlighted cell can be edited using the backspace key. Once the backspace key is pressed, the mouse and left-right cursors can be used for editing.
Left double click	If the mouse is positioned over any graphic or text window, double clicking will cause the window contents to be recalculated and redrawn. This is the same as selecting Update. For editor windows, invokes the solves dialog box.
Right click	If the mouse is positioned over any graphic or text window, right clicking will invoke the settings box for that window. This is the same as selecting Settings. For editor windows, invokes the solves dialog box.
Tab	Moves to the next cell in the editor windows, or the next field in the dialog boxes.
Shift + Tab	Moves to the previous cell in the editor windows, or the previous field in the dialog boxes.
Home/End	Moves to the top left/bottom left of the current editor, or the top and bottom of the text window.
Ctrl + Home/End	Moves to the top right/bottom right of the current editor.
Page Up/Down	Moves up or down one screen at a time.
Ctrl + Page Up/Down	Moves to the top/bottom of the current column.

## **Windows shortcuts**

Action	Result
Alt + Tab	Toggles between all currently running applications. Extremely useful for quickly moving between ZEMAX and other running applications.
Ctrl + Esc	Invokes the Windows task list. The task list allows selection of another running application.
Alt	Selects the top menu bar of the current application.
Alt + letter	Selects the menu option with that letter. For example, Alt+F selects the File menu.
Tab	Moves to the next option or field.
Shift + Tab	Moves to the previous option or field.
Space bar	Toggles selected checkboxes on or off.
Enter	Equivalent to pressing the highlighted or default button in a dialog box.
letter	Pressing the first letter of a word in a drop down box will select that entry in the drop down box.

## **Using the Windows clipboard**

One of the most useful features in Windows is the clipboard. The clipboard is "holding area" for graphics and text. The advantage to using the clipboard is that virtually all Windows programs can either import or export to the clipboard.

Since ZEMAX is primarily used to generate graphical and text data, ZEMAX only supports exporting to the clipboard. Once the desired data has been copied to the clipboard, it is easy for another application, such as a word processor, graphics editor, or desktop publishing system to retrieve the data. For example, the graphics within this manual were generated within ZEMAX, copied to the clipboard, and then pasted in to a desktop publishing program from the clipboard.

To get ZEMAX graphics and text into the clipboard is extremely simple. Choose the graphic or text window desired, then select Window, Copy Clipboard. Nothing will appear to happen (the data transfer is extremely fast), however the data will now be available to other applications.

To now get the clipboard data into a document processing application, run that application, and choose Paste, usually from the Edit menu of that application. See the documentation for that application for details. Note that both ZEMAX and the receiving application can be left running at all times. If you find yourself closing an application, then opening another, then closing again; you are not getting the point of Windows! Leave all your applications that you are using running at once, and use Alt-Tab to switch between them. ZEMAX pastes graphics in full color at full resolution. The clipboard graphics format is a device-independent vector file, which will print at full resolution on any print device.

With a little practice, moving graphics and text from ZEMAX to other applications is extremely fast. Older DOS programs often rely on using a "print to file" technique to save graphics and text. Often, a HPGL plot file was generated, and then imported into another application at a later time. This method is now considered obsolete, and is rarely used. However, you can still print to a file in whatever file format you have selected for a printer driver. To generate an HPGL file, use the HP 7470A printer driver (you may need to install it first, see the Windows documentation for details) and choose "Print To File" on the print dialog box.

Some Windows applications cannot import ZEMAX graphics, even if they appear correctly in the Windows Clipboard Viewer. In this case, the work around is to use the "Export Metafile" option described in the section on graphics windows operations earlier in this chapter. Once the metafile is created, most Windows applications can import the graphic.

This page intentionally left blank.

---

# CONVENTIONS AND DEFINITIONS

## Introduction

---

This chapter describes conventions and defines terminology used throughout this manual. Most of the conventions and terms ZEMAX uses are common in the optics industry, however there may be some important differences.

## Active configuration

---

The active configuration is the configuration currently being displayed in the lens data editor. For details see the Chapter "Multi-Configurations".

## Angular magnification

---

The ratio of the paraxial image space chief ray angle to the paraxial object space chief ray angle. The angles are measured with respect to the paraxial entrance and exit pupil locations.

## Apodization

---

Apodization refers to the uniformity of illumination in the entrance pupil of the system. By default, the pupil is always illuminated uniformly. However, there are times when the pupil should have a non-uniform illumination. For this purpose, ZEMAX supports pupil apodization, which is a variation of amplitude over the pupil.

Three types of pupil apodization are supported: uniform, Gaussian, and tangential. For each type (except uniform), an apodization factor determines the rate of variation of amplitude in the pupil. See the discussion on apodization types and factors in the Chapter "System Menu".

## Cardinal planes

---

The term cardinal planes (sometimes called cardinal points) refers to those special conjugate positions where the object and image surfaces have a specific magnification. The cardinal planes include the principal planes, where the magnification is +1, the anti-principal planes, where the magnification is -1, the nodal planes, where the angular magnification is +1, and the focal planes, where the magnification is 0 for the image space focal plane and infinite for the object space focal plane.

Except for the focal planes, the cardinal planes are conjugates with each other, that is, the image space principal plane is conjugate with the object space principal plane, etc. If the lens has the same index in both object space and image space, the nodal planes are identical to the principal planes.

ZEMAX lists the distance from the image surface to the various image space planes, and lists the distance from the first surface to the various object space planes.

## Chief ray

---

If there is no vignetting, and there are no aberrations, the chief ray is defined to be the ray that travels from a specific field point, through the center of the entrance pupil, and on to the image plane. Note that without vignetting or aberrations, any ray passing through the center of the entrance pupil will also pass through the center of the stop and the exit pupil.

When vignetting factors are used, the chief ray is then considered to be the ray that passes through the center of the vignetted pupil, which means the chief ray may not necessarily pass through the center of the stop.

If there are pupil aberrations (and there virtually always are), then the chief ray may pass through the center of the paraxial entrance pupil (if ray aiming is not used) or the center of the stop (if ray aiming is used), but generally, not both.

If there are vignetting factors which decenter the pupil, then the chief ray will pass through the center of the vignetted entrance pupil (if ray aiming is not used) or the vignetted stop surface (if ray aiming is used).

The common convention used is that the chief ray passes through the center of the vignetted pupil, while the principal ray passes through the center of the unvignetted stop. ZEMAX never uses the principal ray. Most calculations are referenced to the chief ray or the centroid. Note the centroid reference is generally superior because it is based upon the aggregate effect of all the rays that actually illuminate the image plane, and not on the arbitrary selection of one ray which is "special".

## Coordinate axes

---

The optical axis is the z-axis, with the initial direction of propagation from the object being the positive z direction. Mirrors can subsequently reverse the direction of propagation. The coordinate system is right handed, with the sagittal x-axis being oriented "into" the monitor on a standard layout diagram. The tangential y-axis is vertical.

The direction of propagation is initially left-to-right, down the positive z axis. After an odd number of mirrors (1, 3, 5..) the beam physically propagates in a negative z direction. Therefore, all thicknesses after an odd number of mirrors should be negative.

## Diffraction limited

---

The term diffraction limited implies that the performance of an optical system is limited by the physical effects of diffraction rather than imperfections in either the design or fabrication. A common means of determining if a system is diffraction limited is to compute or measure the optical path difference. If the peak to valley OPD is less than one quarter wave, then the system is said to be diffraction limited.

There are many other ways of determining if a system is diffraction limited, such as Strehl ratio, RMS OPD, standard deviation, maximum slope error, and others. It is possible for a system to be considered diffraction limited by one method and not diffraction limited by another method.

## Edge thickness

---

ZEMAX uses two different definitions for the term "edge thickness". Usually, the edge thickness is computed for a specific surface by:

$$E_i = Z_{i+1} - Z_i + T_i$$

where  $z_j$  is the sag of the surface at the +y semi-diameter of the surface,  $z_{i+1}$  is the sag of the next surface at the +y semi-diameter of the next surface, and  $T_i$  is the axial thickness of the surface. Note that the edge thicknesses are computed accounting for the sag at the respective semi-diameter of each surface, which in general are different.

Note also that edge thickness is normally computed for the +y radial aperture, which may be inadequate if the surface is not rotationally symmetric, or if surface apertures have been placed upon either of the surfaces.

The exception to this rule is when computing edge thickness solves. Because the edge thickness solve can change the center thickness, the edge thickness solve can change where rays strike the following surface, which in turn means the semi-diameter of the next surface may change. If the semi-diameter of the next surface is used in the edge thickness computation, an "infinite loop" or circular definition may occur.

For this reason, edge thickness solves compute the edge thickness strictly at the semi-diameter of the first surface, for both surfaces. The semi-diameter of the second surface is never used, although the curvature or shape of the surface is used.

## Effective focal length

---

The distance from the rear principal plane to the paraxial image plane. This is calculated for infinite conjugates. Principal plane calculations are always based upon paraxial ray data. The effective focal length is always referenced to an index of refraction of 1.0, even if the image space index is not unity.

## Entrance pupil diameter

---

The diameter in lens units of the paraxial image of the stop in object space.

## Entrance pupil position

---

The paraxial position of the entrance pupil with respect to the first surface in the system. The first surface is always surface 1, not the object surface, which is surface 0.

## Exit pupil diameter

---

The diameter in lens units of the paraxial image of the stop in image space.

## Exit pupil position

---

The paraxial position of the exit pupil with respect to the image surface.

## Extra data

---

Extra data values are used to define certain non-standard surface types. For example, extra data values are used to define the phase of diffractive optic surfaces such as the Binary 1 surface type. For a complete discussion of the extra data values see the section "Extra Data" in the Chapter "Surface Types".

## Field angles and heights

---

Field angles are always in degrees. The angles are measured with respect to the paraxial entrance pupil position. Positive field angles refer to negative object coordinates, and positive ray slopes. If object or image heights are used to define the field points, the heights are measured in lens units. When image heights are used as the field definition, the heights are the paraxial image heights. If the optical system has distortion, then the actual image points will be at different locations.

## Float by stop size

---

Float by stop size is one of the system aperture types supported by ZEMAX. This phrase refers to the fact that the entrance pupil position, object space numerical aperture, image space F/#, and stop surface radius all are specified if just one of them is specified. Therefore, setting the stop radius, and then allowing the other values to be whatever they are, is a perfectly valid way of defining the system aperture. It is particularly handy when the stop surface is a real, unchangeable aperture buried in the system, such as when designing null corrector optics.

## Glasses

---

Glasses are entered by name in the glass column. Blanks are treated as air, with unity index. Mirrors can be specified by entering "MIRROR" for the glass type, although this name will not appear in the glass catalog. Available glasses may be reviewed, and new ones entered using the glass catalog tool. See the Chapter "Using Glass Catalogs" for details.

## Hexapolar rings

---

ZEMAX usually selects a ray pattern for you when performing common calculations such as spot diagrams. The ray pattern refers to how a set of rays is arranged on the entrance pupil. The hexapolar pattern is a rotationally symmetric means of distributing a set of rays. The hexapolar pattern is described by the number of rings of rays around the central ray. The first ring contains 6 rays, oriented every 60 degrees around the entrance pupil with the first ray starting at 0 degrees (on the x-axis of the pupil). The second ring has 12 rays (for a total of 19, including the center ray in ring "0"). The third ring has 18 rays. Each subsequent ring has 6 more rays than the previous ring.

Many features which require a sampling parameter to be specified (such as the geometric MTF calculation) use the number of hexapolar rings as a convenient means of specifying the number of rays. If the hexapolar sampling density is 5, it does not mean that 5 rays will be used. A sampling of 5 means  $1 + 6 + 12 + 18 + 24 + 30 = 91$  rays will be used.



## Image space F/#

---

Image space F/# is the ratio of the paraxial effective focal length calculated at infinite conjugates over the paraxial entrance pupil diameter. Note that infinite conjugates are used to define this quantity even if the lens is not used at infinite conjugates.

## Lens units

---

Lens units are the primary unit of measure for the lens system. Lens units apply to radii, thicknesses, apertures, and other quantities, and may be millimeters, centimeters, inches, or meters.

## Marginal ray

---

The marginal ray is the ray that travels from the center of the object, to the edge of the entrance pupil, and on to the image plane.

If there is vignetting, ZEMAX extends this definition by defining the marginal ray to be at the edge of the vignetted entrance pupil. If ray aiming is on, then the marginal ray is at the edge of the vignetted stop.

See also the definition of the chief ray.

## Maximum field

---

Displays the maximum radial field angle in degrees if "field angle" mode is selected, the maximum radial object coordinate in lens units if "object height" mode is selected, or the maximum radial image coordinate in lens units if "image height" mode is selected. Field modes are set on the Field Data dialog box available under the "System" menu.

## Non-paraxial systems

---

The term non-paraxial system refers to any optical system which cannot be adequately represented by paraxial ray data. This generally includes any system with tilts or decenters (implemented with coordinate break surfaces), holograms, gratings, paraxial lenses, cubic splines, ABCD matrices, gradient index, or diffractive components.

A great deal of optical aberration theory has been developed for systems with conventional refractive and reflective components in rotationally symmetric configurations. This includes Seidel aberrations, distortion, Gaussian beam data, and virtually all first order properties such as focal length, F/#, and pupil sizes and locations. All of these values are calculated from paraxial ray data.

If the system being analyzed contains any of the non-paraxial components described, then any data computed based upon paraxial ray tracing cannot be trusted.

## Normalized field and pupil coordinates

---

Normalized field and pupil coordinates are often used in both the ZEMAX program and documentation. There are four normalized coordinates: Hx, Hy, Px, and Py. The Hx and Hy values are the normalized field coordinates; Px and Py are the normalized pupil coordinates.

The normalized field and pupil coordinates represent points on a unit circle. The radial size of the field of view (or object height if fields are defined in terms of object height) is then used to scale the normalized field coordinates, and the entrance pupil radius is used to scale the normalized pupil coordinates. For example, suppose the maximum object height is 10 mm. This would be the case if you defined 3 fields, at 0, 7, and 10 mm. The coordinate (Hx = 0, Hy = 1) would refer to a ray which starts at the top of the object (x = 0 mm, y = 10 mm). The coordinate (Hx = -1, Hy = 0) would refer to a ray which starts at (x = -10 mm, y = 0 mm) on the object surface.

The pupil coordinates work the same way. Suppose the entrance pupil radius (Not diameter) is 8 mm. Then (Px = 0, Py = 1) refers to a ray which is aimed to the top of the entrance pupil. On the entrance pupil surface, the ray will have a coordinate of (x = 0 mm, y = 8 mm).

Note that the normalized coordinates should always be between -1 and 1, and that

$$H_x^2 + H_y^2 \leq 1, P_x^2 + P_y^2 \leq 1.$$

The advantage of using normalized coordinates is that certain rays always have the same coordinates, independent of the size or location of either the object or the entrance pupil. For example, the marginal ray is the ray which travels from the center of the object to the top of the entrance pupil, or  $(H_x = 0, H_y = 0, P_x = 0, P_y = 1)$ . The chief ray travels from the top of the field to the center of the pupil, or  $(0, 1, 0, 0)$ .

Another significant advantage of this system is scalability. Suppose prior to optimizing a lens design, you define a ray set to compute the system merit function. By using normalized coordinates, the same ray set will work unaltered if the entrance pupil size or position or object size or position is changed later, or perhaps even during the optimization procedure.

The normalized field coordinates work even if the field positions are defined in terms of angles. For example, suppose you have selected y-field angles of 0, 7, and 10 degrees. This implies your maximum field "radius" in angular space is 10 degrees. Normalized field coordinates of  $H_x = 0$  and  $H_y = 1$  select the x-field of 0 and the y-field of 10 degrees. The coordinates  $H_x = -.5$  and  $H_y = .4$  select the x-field angle of -5 degrees, and a y-field angle of 4 degrees. Note that even if no x-angles are defined, you may trace rays from these fields using non-zero values for  $H_x$ . The  $H_x$  and  $H_y$  values always refer to points on a circle in object angle space whose radius is determined by the maximum radial field. If you define a single field point whose x-field angle is 10 degrees and y-field is -6 degrees, then the maximum radial field is 11.66 degrees, and  $H_x$  and  $H_y$  will then be normalized to this radius. Notice that the coordinates are relative field angles if you define your object in terms of field angles. If you are using object height then the  $H_x$  and  $H_y$  are normalized object heights.

## **Object space numerical aperture**

---

Object space numerical aperture is a measure of the rate of divergence of rays emanating from the object surface. The numerical aperture is defined as the index of refraction times the sine of the marginal ray angle, measured in object space. The marginal ray defines the boundary of the cone of light diverging from an object point.

## **Parameter data**

---

Parameter data values are used to define certain non-standard surface types. For example, parameter data may include aspheric coefficients, grating spacings, or tilt and decenter data. For a complete discussion of the parameter data values see the section "Parameter Data" in the Chapter "Surface Types".

## **Paraxial and paraxial rays**

---

The term paraxial strictly speaking means "near the axis". Paraxial optics are optics that are well described by the linear form of Snell's law. Snell's law is:

$$n \sin \theta = n' \sin \theta'$$

For small angles this becomes

$$n \theta = n' \theta'$$

A great number of definitions in optics is based upon the assumption of linearity. Aberrations are deviations from this linearity, and so the paraxial properties of optical systems are often considered the properties the system has in the absence of aberrations.

Although there are numerous simple formulas for computing paraxial entities, such as focal length,  $F/\#$ , magnification, etc., ZEMAX generally does not use these formulas. Internally, ZEMAX traces "paraxial" rays which are real (real means using Snell's law explicitly) that make small angles with respect to the base ray, which is usually an axis or chief ray.

The reason ZEMAX uses paraxial rays rather than paraxial formulas is because so many optical systems include non-paraxial components. Non-paraxial means these components are not well described by conventional axial first-order theory. This includes tilted or decentered systems, and systems using holograms, diffractive optics, general aspheres, and gradient index lenses.

ZEMAX computes many paraxial entities, and these values should be used with caution when the optical system has non-standard components. Often, the generalization of using paraxial rays is acceptable, but for very unusual systems, a few first-order numbers may be inadequate to describe the imaging properties.

## Paraxial image height

---

The paraxial radial size of the image in lens units of the full field image at the paraxial image plane.

## Paraxial magnification

---

The radial magnification, being the ratio of paraxial image height to object height. The paraxial magnification is measured at the paraxial image plane. The value is always zero for infinite conjugate systems.

## Paraxial working F/#

---

The paraxial working F/# is defined as

$$W = \frac{1}{2n\theta}$$

where  $\theta$  is the paraxial marginal ray angle in image space and  $n$  is the index of refraction of image space. The paraxial marginal ray is traced at the specified conjugates. For non-axial systems, this parameter is referenced to the axis ray and is averaged over the pupil. The paraxial working F/# is the effective F/# ignoring aberrations. See also the definition for working F/#.

## Primary wavelength

---

The primary wavelength in microns is displayed. This value is used for calculating most other paraxial or system values, such as pupil positions.

## Radii

---

The radius of curvature of each surface is measured in lens units. The convention is that a radius is positive if the center of curvature is to the right (a positive distance along the local  $z$  axis) from the surface vertex, and negative if the center of curvature is to the left (a negative distance along the local  $z$  axis) from the surface vertex. This is true independent of the number of mirrors in the system.

## Sagittal and Tangential

---

The term "tangential" refers to data computed in the tangential plane, which is the plane defined by a line and one point: the line is the axis of symmetry, and the point is the field point in object space. The sagittal plane is the plane orthogonal to the tangential plane, which also intersects the axis of symmetry at the entrance pupil position.

For typical rotationally symmetric systems with field points lying along the  $Y$  axis, the tangential plane is the  $YZ$  plane and the sagittal plane is the plane orthogonal to the  $YZ$  plane which intersects the center of the entrance pupil.

The problem with this definition is that it is not readily extended to non-rotationally symmetric systems. For this reason, ZEMAX instead defines the tangential plane to be the  $YZ$  plane regardless of where the field point is, and tangential data is always computed along the  $y$  axis in object space. The sagittal plane is the orthogonal to the  $YZ$  plane, and intersects the center of the entrance pupil in the usual way, and sagittal data is always computed along the  $x$  axis in object space.

The philosophy behind this convention is as follows. If the system is rotationally symmetric, then field points along the  $y$  axis alone define the system imaging properties, and these points should be used. In this case, the two different definitions of the reference planes are redundant and identical. If the system is not rotationally symmetric, then there is no axis of symmetry, and the choice of reference plane is arbitrary.

## Semi-diameters

---

The size of each surface is described by the semi-diameter setting. The default setting is the radial distance to the aperture required to pass all real rays without clipping any of the rays. Typing a value for the semi-diameter column results in the character "U" being displayed next to the value. The "U" indicates that the semi-diameter is user defined. This setting only affects how the lens is drawn in the layout features, and does not effect whether

or not any ray passes that surface. Note this applies in particular to the stop surface and the image surface. To clip a ray, either use the vignetting factors feature or use a surface aperture.

Semi-diameters on any surface for axial symmetric systems are computed very accurately, as long as the surface does not lie within the caustic of the ray bundle. ZEMAX estimates semi-diameters by computing a few marginal pupil rays. For non-axial systems, ZEMAX estimates the required semi-diameters using either a fixed number of rays or by an iterative technique, which is slower but more accurate. See the "System Menu" chapter for details. It is important to note that the "automatic" semi-diameter computed by ZEMAX is an estimate, although it is generally a very good one.

## **Surface apertures**

---

Surface apertures include circular, rectangular, elliptical, and spider shaped apertures which can vignette rays. Surface apertures do not effect ray launching or tracing, except for the termination of a ray if it does not pass the surface aperture. Surface apertures have no effect on the system aperture. For more information, see the Chapter "Editors Menu" in the section "Specifying surface apertures".

## **System aperture**

---

The system aperture is the overall system F/#, Entrance Pupil Diameter, Numerical Aperture, or Stop Size. Any of these 4 quantities is sufficient to define the other 3 for a particular optical system. The system aperture is used to define the object space entrance pupil diameter, which in turn is used to launch all rays. The system aperture is always circular. Rays may be vignetted after being launched by various surface apertures. There is only one system aperture, although there may be many surface apertures.

## **Thicknesses**

---

Thicknesses are the relative distance to the next surface vertex in lens units. Thicknesses are not cumulative, each one is only the offset from the previous vertex along the local z axis.

Thicknesses always change sign after a mirror. After an odd number of mirrors, all thicknesses should be negative. This sign convention is independent of the number of mirrors, or the presence of coordinate breaks. This fundamental convention cannot be circumvented through the use of coordinate rotations of 180 degrees.

## **Total internal reflection (TIR)**

---

TIR refers to the condition where a ray makes too large an angle with respect to the normal of a surface to meet the refraction condition as specified by Snell's Law. This usually occurs when a ray with a large angle of incidence is refracting from a high index media to a lower index media, such as from glass to air. Rays which are TIR are considered errors, and are terminated. Physically, the ray would reflect rather than refract from the boundary, but ZEMAX does not consider this effect.

## **Total track**

---

Total track is the length of the optical system as measured by the vertex separations between the "left most" and "right most" surfaces. The computation begins at surface 1. The thickness of each surface between surface 1 and the image plane is considered, ignoring any coordinate rotations. The surface which lies at the greatest z coordinate defines the "right most" surface, while the surface with the minimum z coordinate defines the "left most" surface. Total track has little value in non-axial systems.

## **Vignetting factors**

---

Vignetting factors are coefficients which describe the apparent entrance pupil size and location for different field positions. ZEMAX uses four vignetting factors: VDX, VDY, VCX, and VCY. These factors represent decenter x, decenter y, compression x, and compression y, respectively. The default values of all four factors are zero, which indicates no vignetting.

Both the field of view and the entrance pupil of an optical system can be thought of as unit circles. The normalized field and pupil coordinates, defined earlier in this chapter, are the coordinates on these two unit circles. For example, the pupil coordinates ( $p_x = 0$ ,  $p_y = 1$ ) refer to the ray which is traced from some point in the field to the

top of the entrance pupil. If there is no vignetting in the system, ZEMAX will trace rays to fill the entire entrance pupil during most computations.

Many optical systems employ deliberate vignetting. This means a portion of the rays are intentionally "clipped" by apertures other than the stop surface. There are two reasons for introducing vignetting in an optical system. First, vignetting decreases the size of the lenses, particularly in wide angle lenses. Second, vignetting may remove a portion of the beam which would be excessively aberrated. Vignetting usually increases the F/# as a function field angle (which darkens the image), but the image quality may improve if the most severely aberrated rays are clipped.

Vignetting factors redefine the entrance pupil for a specific field position. The normalized pupil coordinates are modified according to:

$$P'_x = VDX + P_x(1 - VCX) , \text{ and}$$

$$P'_y = VDY + P_y(1 - VCY) .$$

The VDX term can shift the apparent pupil left and right, while VCX makes the pupil larger or smaller in the x direction. Similar results hold for the VDY and VCY values. Note that if the vignetting factors are all zero, the pupil coordinates are left unmodified. Vignetting factors provide a convenient way of designing optics which employ vignetting. However, there are drawbacks to using vignetting factors that must be understood:

- 1) Some ZEMAX features are capable of tracing rays from arbitrary field positions where no vignetting factors have been assigned. These features include the image analysis feature, optimization, field curvature and distortion, RMS vs. field, relative illumination, and others. These features may not provide completely accurate results for data computed at field positions other than the defined fields. For rotationally symmetric systems, or systems with field points entirely along the y axis, ZEMAX interpolates between adjacent field points to estimate the vignetting factors to use at intermediate field points. For more general optical systems with X field values, ZEMAX uses the closest defined field point for determining the vignetting factors for an arbitrary field point. Features such as ray fans, MTF, spot diagrams, and the default merit functions all use only defined fields and are therefore completely accurate.
- 2) Once a vignetting factor is defined, it is up to the designer to insure that rays beyond the apparent pupil are in fact vignetted! If the vignetting factor is used to shrink the size of the lenses, then the lenses should be made no larger than is required to pass the rays which are at the edge of the apparent pupil. If rays from beyond the vignetted aperture are allowed to pass in the real optical system, then the lens performance will not correlate with the computer model.

The vignetting factors work with and without ray aiming turned on. If ray aiming is off, then the paraxial entrance pupil is remapped according to the equations given earlier. If ray aiming is turned on, then the remapping is done at the stop surface.

One possible application of vignetting factors is to account for pupil aberration without using the ray aiming feature. This is an advanced trick which can be used to speed up ray tracing in wide angle systems.

Vignetting factors may be defined on the "Field Data" dialog box. See the Chapter "System Menu" for more information. Vignetting factors may also be zoomable parameters; see the Chapter "Multi-Configurations". For more information on the use of vignetting as a design tool, see any of the good books referenced in the first chapter.

## **Wavelength data**

---

Wavelength data are always measured in microns. Wavelength data is entered on the "Wavelength Data" dialog box; see the Chapter "System Menu" for details.

## **Working F/#**

---

Working F/# is defined as

$$W = \frac{1}{2n \sin \theta} ,$$

where  $\theta$  is the marginal ray angle in image space and  $n$  is the index of refraction of image space. The marginal ray is traced at the specified conjugates.

For non-axial systems, this parameter is referenced to the axis ray and is averaged over four rays, at the top, bottom, left, and right side of the vignetted pupil. The average is of the square of the numerical aperture of the four rays, and the resulting RMS numerical aperture is converted back to  $F/\#$ .

Working  $F/\#$  is generally much more useful than image space  $F/\#$  because it is based upon real ray data at the actual conjugates of the lens. See also the definition for paraxial working  $F/\#$ .

This page intentionally left blank.

## Introduction

---

This chapter is intended to teach you how to use ZEMAX. Each section in this chapter will walk you through a different design problem. The first few design examples are very simple, and may not seem worth the trouble if you are an experienced lens designer. However, if you take the time to go through the lessons you will learn how to run ZEMAX, and then you can move on to your particular design interests.

The first few example sessions do provide some tutorial comments regarding lens design theory to aid those who lack familiarity with the terminology. However this manual in general, and these examples in particular, will not promote the novice designer to an expert. If you cannot follow these examples, or you do not understand the mathematics behind the calculations the program performs, see any of the good books referenced in the chapter "Introduction". You must have properly installed ZEMAX before beginning the lessons.

## Lesson 1: a singlet

---

What you will learn: Starting ZEMAX, entering wavelength and lens data, generating rayfan, OPD, and spot diagrams, defining thickness solves and variables, running simple optimizations.

Suppose you needed to design a F/4 singlet lens with a 100 mm focal length, for use on-axis in the visible spectrum using BK7 glass. How do you start?

First of all, run ZEMAX. The ZEMAX main screen will appear, as will the lens data editor (LDE). You can move and resize the LDE to suit your taste. The LDE consists of various rows and columns arranged like a spreadsheet. The Radius, Thickness, Glass, and Semi-diameter columns are the ones you will use the most, the others will be used only for certain types of optical systems.

One cell in the LDE is highlighted in "reverse video". This means it displays letters on a background in different colors than the other cells. If no cell is highlighted, then click with the mouse on any cell to highlight it. This reverse video bar is referred to as the "cursor" in this tutorial. You can navigate around in the LDE by clicking on the cell you wish to move the cursor to with the mouse, or you can simply use the cursor keys. The operation of the LDE is simple, and with a little practice you will master it.

To begin, we will enter the wavelengths for our system. This does not always have to be done first; we simply elect to do so now. On the main screen menu bar, select System then Wavelengths.

The "Wavelength Data" dialog box will appear in the center of the screen. ZEMAX is full of these dialog boxes; they are used to enter data and to present you with options. Using the mouse, click on the "Use" box on the second and third rows. This will add two more wavelengths to bring the total to three. Now enter in the first "Wavelength" row .486. This is the wavelength, in microns, of the Hydrogen F-line.

ZEMAX always uses microns for wavelength. Now go to the second row, wavelength column, and type .587. Finally, enter .656 in the third row. This is essentially all there is to entering data in ZEMAX; move to the appropriate field and type in the data. On the far right edge of the screen, you can see the primary wavelength indicator column. This indicator points to the primary wavelength, currently .486 microns. Click on the second row of the primary wavelength indicator. The indicator moves down to the .587 position. The primary wavelength is used to calculate the paraxial parameters such as focal length, magnification, and so on.

ZEMAX always uses microns for wavelength.

The "Weight" column is used for optimization, and for calculating wavelength weighted data such as RMS spot size and Strehl ratio. For now leave all the weights at 1.0. Click on OK to save the changes and exit the wavelength data dialog box.

Now we need to define an aperture for the lens. This tells ZEMAX how large to make all of the lenses, among other things. Since we wanted a F/4 lens, we will need a 25 mm aperture (a 100 mm focal length divided by F/4). To set this aperture, select System then General. The "General Data" dialog box appears. Click on the cell



showing "Aper Value" and enter a value of 25. Note the aperture type is by default "Entrance Pupil Diameter", other types of aperture settings may be selected. The other options listed are not needed yet, so click on OK.

We still do not have a lens. The important surface data still needs to be added. ZEMAX models optical systems using a series of surfaces. Each surface has a radius of curvature, a thickness (the axial distance to the next surface), and a glass. Some surfaces may have other data, we will cover that later. Note three surfaces are shown in the LDE: The object surface, denoted OBJ on the left edge, the stop, denoted STO, and the image plane, denoted IMA. For our singlet, we need a total of four surfaces: The object, the front surface of the lens (which will also be the stop), the back surface of the lens, and then the image plane. To insert a fourth surface, move the cursor on top of the word "Infinity" on the image (the last) surface. Now press the insert key. This inserts a new surface at that row, and moves the image surface down. The new surface is numbered 2. Note that the object is surface 0, and then there is surface 1 (denoted STO because it is the stop), 2, and 3 (denoted IMA).

Now we will enter the glass to be used. Move the cursor over to the "Glass" column of surface 1, which is denoted "STO" on the left hand column. Type "BK7" and press enter. ZEMAX has a very extensive glass catalog online. All we need to do is decide to use BK7; ZEMAX will go look up the glass and calculate the indices at each of the wavelengths we have defined.

Since we have a 25 mm aperture, a reasonable thickness for the lens is 4 mm; so move the cursor to the thickness column of surface 1 (where we just entered BK7) and type "4" and enter. Note the default units are millimeters. Other units (centimeters, inches, and meters) are also available.

Now we need to enter some values for the radius of curvature on each side of the lens. Let's guess that the front and rear radii are going to be 100 and -100, respectively. Enter these values on surfaces 1 (STO) and 2, respectively. The sign convention is plus if the center of curvature is right of the lens, minus if it is left. These signs (+100, -100) will yield an equi-convex lens. Also, we need to place the image plane at the focus of the lens, so enter a value of 100 for the thickness of surface 2 as well.

How do we know how good the lens is? Probably the most useful diagnostic tool in lens design is the ray fan plot. To generate a ray fan plot, select Analysis then Fans, then Ray Aberration. You should see the plot appear in a small window (if you see any error messages, go back and make sure you entered in all the data exactly as described). The ray fan should look like Figure E1-1. The plot shows transverse ray aberrations (referenced to the chief ray) as a function of the pupil coordinate. The graph on the left is labeled "EY" for epsilon Y. This is the aberration in the Y direction, sometimes called the tangential, or YZ plane. The graph on the right is labeled "EX" for epsilon X, sometimes called the sagittal, or XZ plane. This ray fan exhibits one glaring design error. The tilt through the origin indicates defocus.

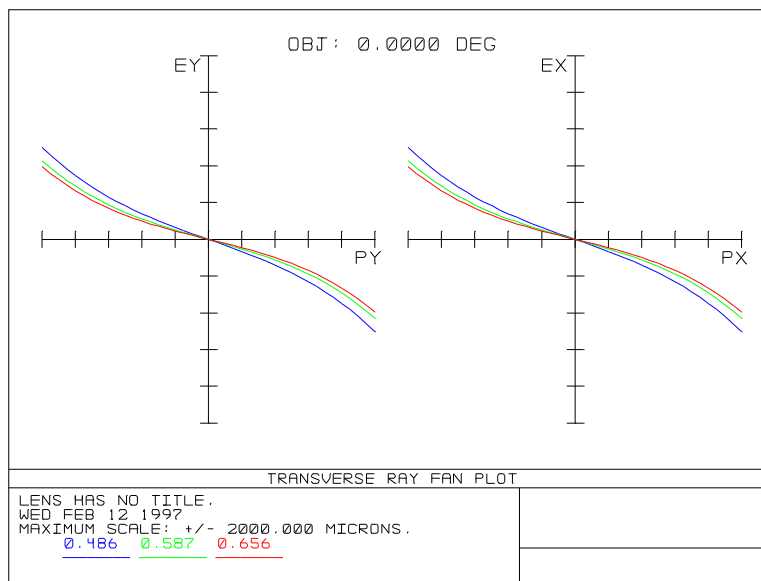


Figure E1-1.

To correct the defocus, we will use a solve placed on the rear side of the lens. Solves (see the chapter "Solves") dynamically adjust certain lens data. To place the image plane in paraxial focus, double click on the thickness of surface 2. The solve dialog box appears; it simply says "Fixed". Click on the drop-down box and change the solve type to "Marginal Ray Height", then click on OK. This solve will adjust the thickness to achieve a marginal ray

height of zero, which is paraxial focus. Note the thickness of surface 2 is automatically adjusted to about 96 mm. Now we need to update the ray fan plot to see the change. From the ray fan window menu, click on Update (double clicking anywhere in the window will also update it). The ray fan as shown in Figure E1-2 appears. Now the defocus is gone, and the dominant aberration is spherical; note the change in plot scale.

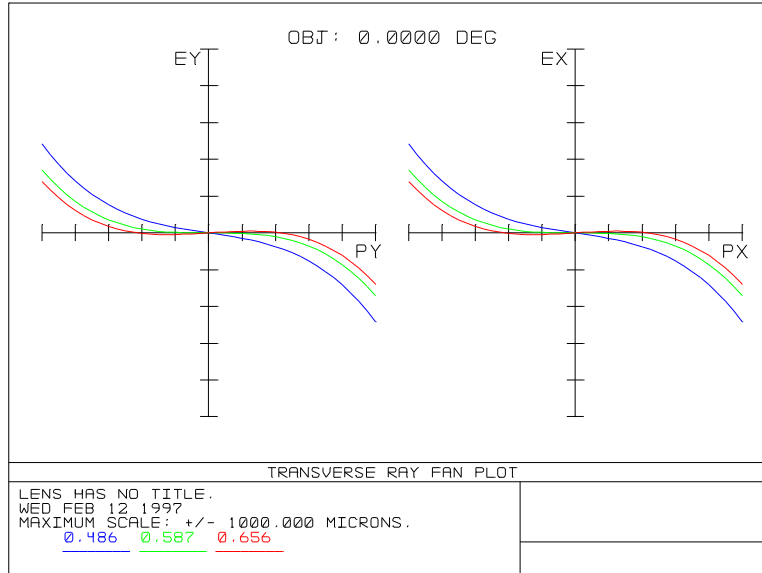


Figure E1-2.

Is this the best design available? We are going to use optimization to do the rest of the design for us. First we will tell ZEMAX which parameters it is free to design with (these are called variables) and then we will tell it the design requirements (these are called targets or operands). There are three variables here that will help us; the front and back curvatures of the lens, and the thickness of the second surface, which will allow defocus to compensate the spherical. Move the cursor to the radius column of surface 1, and then type Ctrl-V (if you prefer the menu interface, click once on the radius, and then select Solves then Variable toggle from the LDE menu; you can also double click on the radius for a drop-down list of options, which includes the variable status). Note the "V" appears to indicate a variable parameter. Typing Ctrl-V works as a toggle. Also set a variable on the radius of surface 2, and the thickness of surface 2. The variable on the thickness of surface 2 overrides the solve placed earlier.

Now we need to define a "Merit Function" for the lens. A merit function defines in a mathematical sense what is a good lens. Merit functions are like golf scores, the lower the better. An ideal lens (for a given application) has a merit function of zero.

To define the merit function, select Editors then Merit Function from the main menu bar. A spreadsheet editor similar to the LDE will appear. From the menu bar of this new window, select Tools then Default Merit Function. On the dialog box that appears, just click on Reset, then OK. You will eventually learn what all the options are for, but for now just accept the defaults. ZEMAX is very good at determining an appropriate default merit function.

ZEMAX has constructed a default merit function for you, consisting of a list of rays to trace which will minimize the RMS wavefront error. This is not quite sufficient, because we also want a 100 mm focal length in addition to minimizing the spot size. Without a specific focal length constraint, ZEMAX will quickly discover that making the focal length infinite (the lens becomes a window) will produce very good wavefront aberration.

Move the cursor to the first row of the Merit function Editor by a single click of the mouse anywhere in the first row. Now press the insert key to add a new row. Now, underneath the "Type" column, enter "EFFL" and press enter. This operand controls the effective focal length. Move the cursor over to the "Target" column, and enter "100" and press enter. For a "Weight", enter a value of 1. We are done defining the merit function; you can remove the merit function editor from the display by double clicking on the upper left corner of the window. The merit function will not be lost, ZEMAX retains it automatically.

Now select Tools then Optimization from the main menu bar. This brings up the optimization tool dialog box. Note the checkbox labeled "Auto Update". If this option is selected, the windows currently on display (such as the ray fan plot) will be automatically updated as the lens changes during optimization. Select auto update by clicking once in the box, and then click on "Automatic". ZEMAX will decrease the merit function rapidly. Click on exit to close the optimization dialog.

The result of this optimization is a "bending" of the lens. The resulting curvatures of the lens have given us approximately a 100 mm focal length, and as small a RMS wavefront error as possible with this simple system. ZEMAX may not optimize to exactly a 100 mm focal length because the EFFL constraint was a "weighted" target to be considered like any other aberration.

We can now investigate the performance using the ray fan plot. The optimized design has a maximum aberration of about 200 microns, as shown in Figure E1-3.

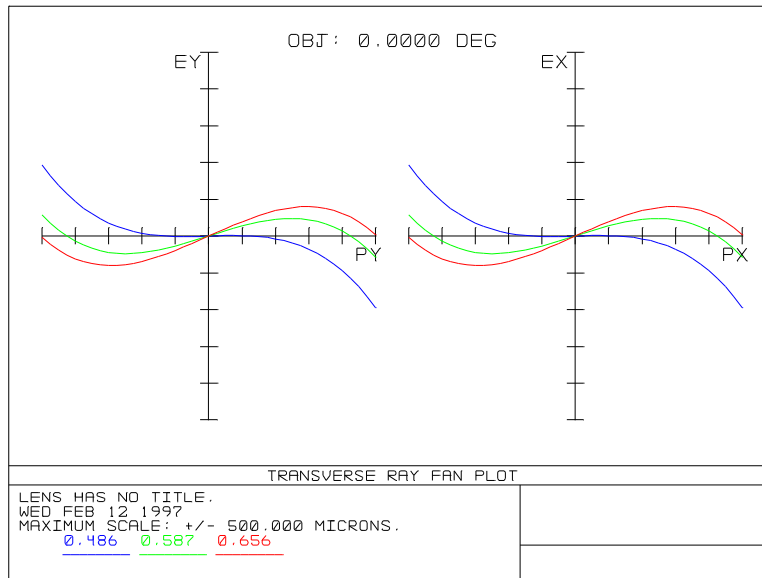


Figure E1-3.

Another way of measuring optical performance is to generate a spot diagram. To see the spot diagram, select Analysis, then Spot Diagrams, then Standard. The spot diagram will appear in another window. This spot is about 400 microns across. For comparison, the Airy diffraction disk is roughly 6 microns across.

Another useful diagnostic tool is the OPD fan. This is a plot of optical path difference (with respect to the chief ray) as a function of the same pupil coordinates used for the rayfan. To see the OPD fan, select Analysis then Fans then Optical Path. You should see the OPD plot as shown in Figure E1-4. This system suffers from roughly 20 waves of aberration, mostly focus, spherical, spherochromatism, and axial color.

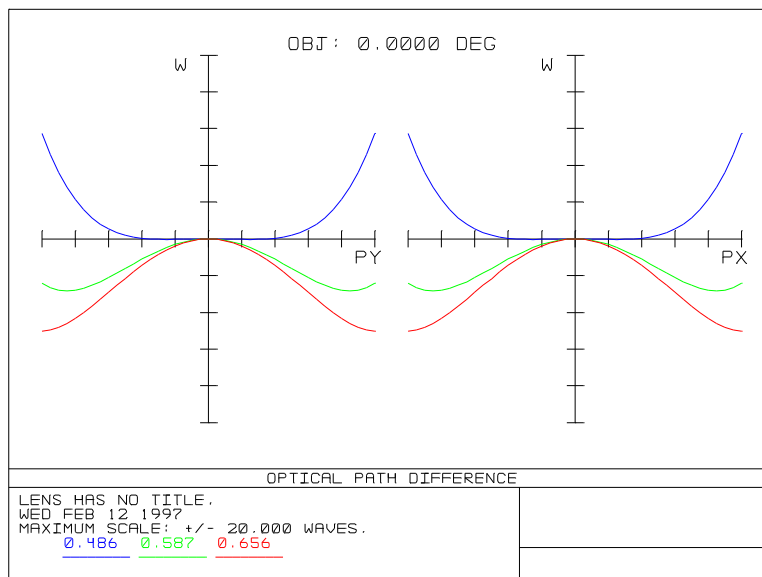


Figure E1-4.

You are probably aware that lenses are considered "diffraction limited" when the wave aberration is roughly one-quarter wave or less (see the references for a more complete discussion of this concept). Clearly our singlet

is not diffraction limited. To improve the performance of this (or any) optical system, the designer must determine which aberrations limit the performance, and what options are available for correcting them. From the ray fans (particularly E1-3) it may be apparent that chromatic aberration is the limiting aberration. (On the other hand, it may not be apparent; again see any of the good books available for advice on interpreting ray fans.)

ZEMAX offers another convenient tool for determining the magnitude of the first-order chromatic aberration: the chromatic focal shift plot. This plots back focal length as a function of wavelength, which indicates the variation in paraxial focus. To see the chromatic focal shift plot, select Analysis, Miscellaneous, Chromatic Focal Shift.

You should see the plot shown in Figure E1-5. Note the wavelength scale is vertical, covering the range of defined wavelengths, and that the maximum change in focal length is about 1540 microns. This monotonic type of curve is typical for singlet lenses.

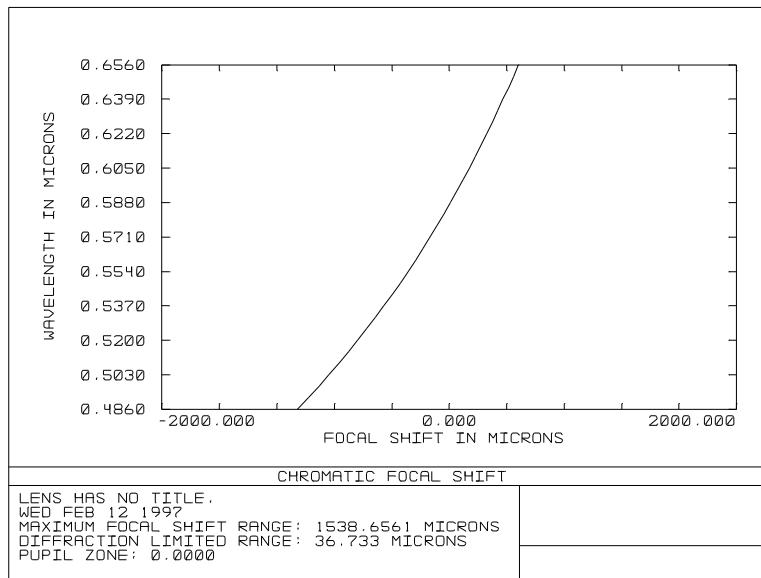


Figure E1-5.

To correct the first-order chromatic aberration, a second glass material is required. This approach leads us to the next example, the design of a doublet. If you would like to save this lens for later evaluation, select File, Save As. ZEMAX will prompt you for a filename which you can enter. Any lenses saved in this manner can be retrieved by selecting File, Open. To quit ZEMAX now, select File, Exit.

## Lesson 2: a doublet

What you will learn: Generating layouts and field curvature plots, defining edge thickness solves, defining field angles.

A doublet lens consists of two glasses, usually cemented (but not always) so they have one common curvature. By using two glasses with different dispersion characteristics, the chromatic aberration may be corrected to first order. This means we should be able to get a parabolic curve instead of a line in the chromatic focal shift plot. This will in turn yield much better imaging. For now we maintain the previous design requirements of 100 mm focal length and on-axis use only; we will add field angles later.

Selecting which two glasses to use requires some finesse; see for example the excellent discussion in Smith, Modern Optical Engineering. Since the purpose of this example is teach you how to use ZEMAX, and not how to design lenses, we will simply suggest the glass choice of BK7 and SF1. If you have just completed the previous lesson, and the singlet lens is still loaded, you do not have to re-enter the wavelengths for this design. Otherwise, enter the wavelengths and apertures according to the instructions in the previous example. Now insert new surfaces as necessary until your lens data editor (LDE) looks like the following table. Not all of the columns are shown for simplicity. If you need to move the stop position so that the first surface is the stop, you can do so by double clicking on the surface type column of the row you wish to make the stop, then choose the "Make Surface Stop" button.

## STARTING PRESCRIPTION FOR DOUBLET EXAMPLE

Surf	Radius	Thickness	Glass
OBJ	Infinity	Infinity	
STO	100.000000 V	3.000000	BK7
2	-100.000000 V	3.000000	SF1
3	-100.000000 V	100.000000 V	
IMA	Infinity		

Because there is no separation between the BK7 and the SF1 media, this is a cemented doublet. ZEMAX does not model the cement itself; it simply models the glasses as being in contact.

If you still have the merit function from the previous example loaded, then there is no need to recreate the merit function. Otherwise, recreate the merit function, including the EFFL operand, as described in the previous example.

Now select Tools, Optimization from the main menu bar, and click on "Automatic". The merit function will begin to decrease; click on "Exit" when it stops. To see if we have made any progress, display the chromatic focal shift plot (if it is not already on your screen select Analysis, Miscellaneous, Chromatic Focal Shift). It should look like Figure E2-1.

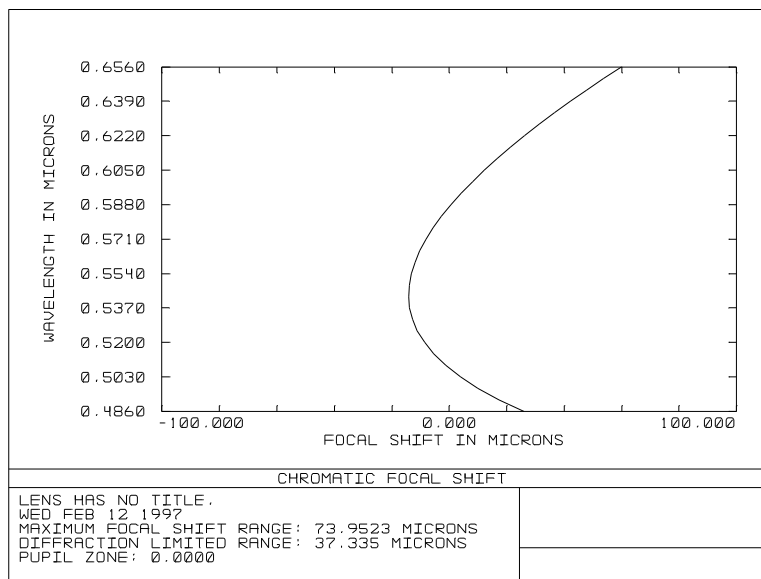


Figure E2-1.

We now have reduced the linear term so that the second-order term dominates, as seen by the parabolic shape. Note also the magnitude of the shift is 74 microns (it was 1540 microns for the singlet).

There are other glass choices that yield better designs. To see the other glasses ZEMAX has in the glass catalog, select Tools, Glass Catalogs. Click on "Exit" when you are done browsing in the glass catalog the catalog.

Now update the rayfan plot by selecting Update from the ray fan window (if the ray fan window is not shown, select Analysis, Fans, Ray Aberration from the main menu bar). The fan is shown in Figure E2-2. The maximum transverse ray aberrations have been reduced to roughly 20 microns. This is a substantial improvement over the 200 microns for the singlet. Note that the slope of the rayfan plot through the origin for each wavelength is roughly the same; this indicates that the relative defocus for each wavelength is small. The slope is not zero, however. This implies that the defocus is being used to balance the spherical aberration. The sideways "S" rayfan curve is typical of a lens with spherical balanced by defocus.

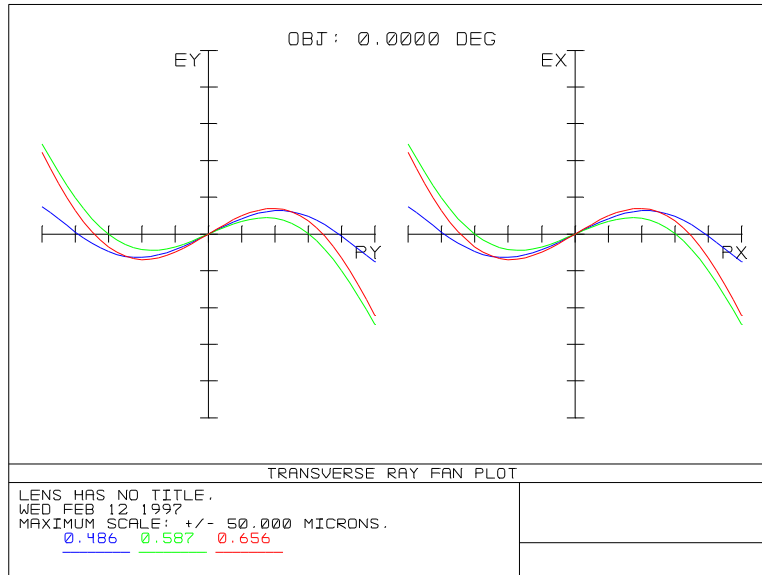


Figure E2-2.

Now that we have designed a lens with good performance characteristics, what does the lens look like? To see a simple, two-dimensional cross section of the lens, select Analysis, Layout, 2D Layout. The layout shown in Figure E2-3 will appear. The layout shows the lens from the first surface to the image plane (by default, other options are available). It also draws three rays (by default) traced from each field at the primary wavelength to the image plane. The rays are at the top, center, and bottom of the entrance pupil, which in this case is surface 1. The first lens has a rather sharp edge, apparently. From the drawing it is difficult to tell if the edge thickness is even positive or negative. Also, it would be nice if the lenses were slightly oversized, so that the useful clear aperture of the lens was smaller than the diameter. This would allow some rim space for polishing and mounting, for example.

We can improve the design with regard to both of these problems. To determine the exact edge thickness, move the cursor to any column on surface 1 (for example, click on the word "BK7" in the LDE). Now type Ctrl-X (the control key and the "x" key) or select Reports, Surface Data. A window will appear which tells you the edge thickness of that surface. The value given is 0.17, which is a little on the thin side.

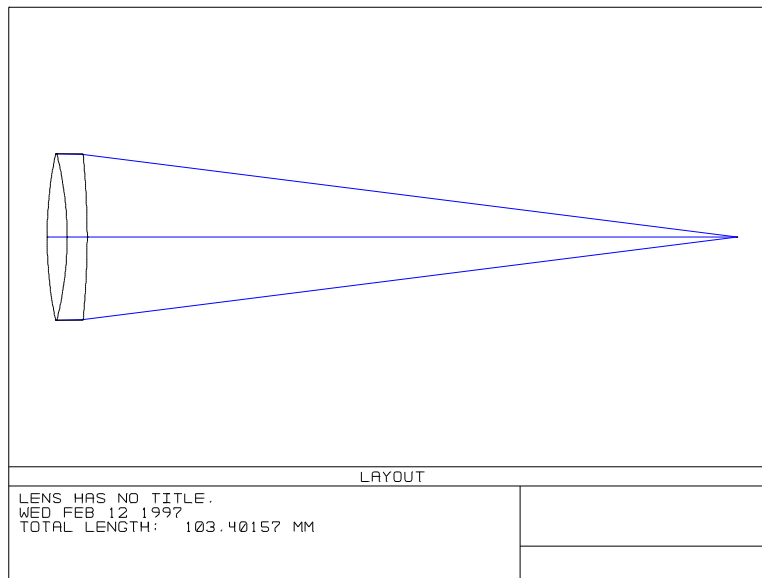


Figure E2-3.

Before we fix the thin edge thickness, we will first oversize the lenses. Move the cursor over to the "Semi-Diameter" column of surface 1. Type in "14" over the 12.5 shown. ZEMAX clears away the 12.5, and displays "14.000000 U". The "U" signifies that this aperture is user-defined. If no "U" is shown, then ZEMAX is free to size the aperture as required. You can remove the "U" by typing Ctrl-V, or by double clicking on the semi-diameter and selecting

"Automatic" for a solve type. After making the change, select System, Update to update the apertures. The value 14 is the semi-diameter, which means the full aperture is 28 mm. Enter an aperture of 14 on surfaces 2 and 3 as well.

Update the layout plot. Now the apertures are oversized, but the first edge thickness is negative! Update the surface data window to see the new edge thickness; it should change to a negative number. To provide a more reasonable edge thickness, we could increase the center thickness. However, there is a more powerful method of maintaining the edge thickness at a particular value.

Suppose we wanted to maintain the edge thickness at 3 mm. To do this, double click on the thickness column of surface 1. This brings up the solve control screen. Select "Edge Thickness" from the list of solves that appears. Two values are shown, a "Thickness", and a "Radial Height". Set the thickness to 3 and the radial height to 0 (If the radial height is zero, ZEMAX uses the defined semi-diameter), then click on OK. On the LDE, the thickness of surface 1 has been adjusted. The letter "E" appears in the box to indicate an active edge thickness solve on that parameter.

Update the surface data window again, and the edge thickness of 3 will be listed. You may also update all of the windows at once by selecting System, Update All. This will refresh the layout and ray fan plots. By adjusting the thickness, we have changed the focal length of the lens slightly. If you like, look at the rayfan plot now. Then optimize again (select Tools, Optimization, then select "Automatic". After optimization, click on "Exit" then select System, Update All, to refresh the plots once more.

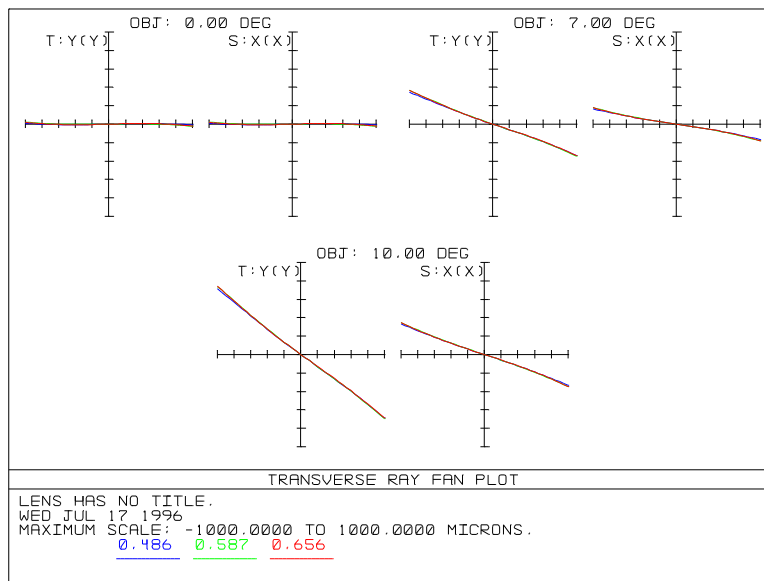


Figure E2-4.

Now we would like to test the off-axis performance of the doublet. From the main menu, select System, Fields to reach the "Field Data" dialog box. Select three fields by clicking on the "Use" column on rows 2 and 3. In the second row of the y field column below, enter 7 (for seven degrees) and in the third row enter 10. Leave the first row at zero, for on-axis. Leave the x field values at zero as well; for a rotationally symmetric system they have little value. Click on OK to close the box.

Now select System, Update All. The rayfan plot is shown in Figure E2-4. Your plot may be slightly different, depending on how you re-optimized after setting the solve.

As you may have inferred from these plots, the off-axis performance is very poor. The reason is that we optimized the design for on-axis performance. What aberration limits us now? Studying the rayfan plot identifies the primary aberration as field curvature. This aberration can be evaluated by the field curvature plot. Select Analysis, Miscellaneous, Field Curv/Dist. The field curvature plot shown in Figure E2-5 appears. Note the left plot indicates the shift in paraxial focus as a function of field angle, while the right plot indicates the real ray distortion, referenced to the paraxial ray. All of the information on the field curvature plot is available from the rayfans. The field curvature plot is proportional to the slopes through the origin of the rayfan plot.

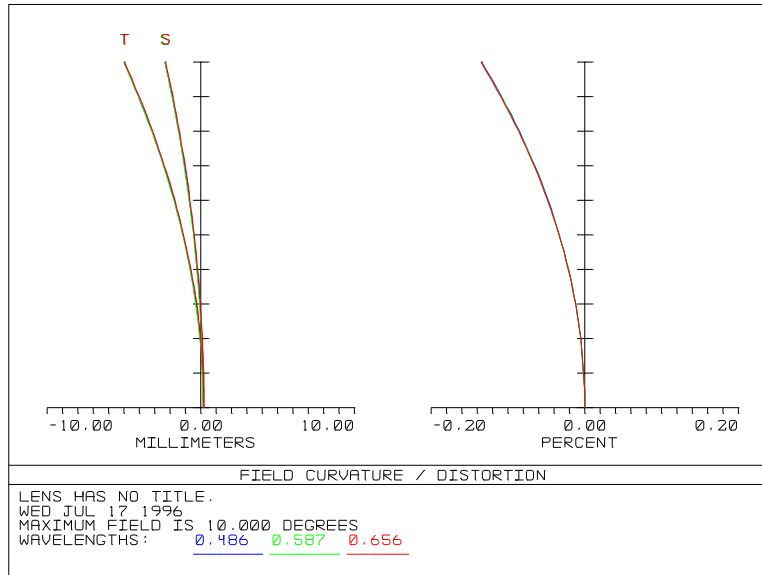


Figure E2-5.

There are techniques for correcting the field curvature, and in fact, the spherical aberration and coma as well. These techniques are also described in Smith, Modern Optical Engineering.

### Lesson 3: a Newtonian telescope

What you will learn: Using mirrors, conic constants, coordinate breaks, three dimensional layouts, obscurations.

If you have already worked the previous two examples, you are ready to move on to a more complicated design. The Newtonian is the simplest telescope corrected for all aberrations on axis, and it serves well to illustrate some fundamental ZEMAX operations. First of all, a Newtonian is a telescope consisting of a simple parabolic mirror, and nothing else. The parabola corrects perfectly all orders of spherical aberration, and since we will only use our telescope on axis, there are no other aberrations. To start fresh, close all windows except the LDE, and select File, New.

Assume we want a 1000 mm F/5 telescope. This implies a mirror with a radius of curvature of 2000 mm, and an aperture of 200 mm. Move the cursor to the radius of curvature column on surface 1, which is the stop, and enter -2000.0. The minus sign indicates concave to the object. Now enter a thickness of -1000 on this same surface. The minus sign indicates we are moving "backwards" due to the reflection off the mirror surface. Now enter "MIRROR" for the "Glass" column on this same surface. Select System, General, then enter an aperture value of 200 on the general data dialog box, and click on OK.

ZEMAX uses one default wavelength of .550, and one field angle of 0, and this is acceptable for our purposes. Now open up a layout window. The rays are drawn from the first surface to the image plane, which now lies to the left of the mirror. If you now perform a standard spot diagram (pull down Analysis, Spot Diagrams, Standard; or type Ctrl-S) you will see a large spot diagram of 77.6 microns RMS. One convenient way of assessing image quality is to superimpose the Airy diffraction ring on top of the spot diagram. To do this, select Settings from the spot diagram menu bar, and on the option "Show Scale", select "Airy Disk", then click on OK. The resulting spot diagram is shown in Figure E3-1.



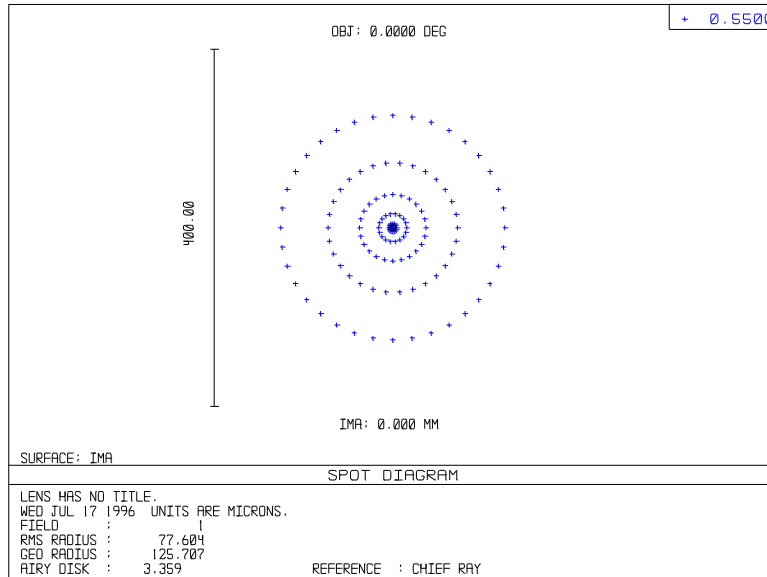


Figure E3-1.

The RMS spot size listed is 77.6 microns. The reason the rays are not diffraction-limited is because we have not yet entered the conic constant. The -2000 radius of curvature we entered earlier just defines a sphere, we need a conic constant of -1 to define a parabola. In the "Conic" column on surface 1, enter -1, and press enter. Now refresh all windows using System, Update All. On the updated spot diagram, you should see the large Airy ring with a tiny cluster of rays in the center. The RMS spot size is zero.

Unfortunately, this high quality image is in a poor location. By being in the incoming beam path, the image is not readily accessible. This is commonly fixed by using a fold mirror placed after the primary mirror. The fold mirror is tipped at 45 degrees, relaying the image up away from the optical axis. To implement the fold mirror, we must first decide where it should go. Since the incoming beam is 200 mm wide, we want the image plane at least 100 mm above the axis. We will select 200 mm, and therefore the fold mirror must be 800 mm from the primary.

Start by changing the thickness on surface 1 to -800. Now move the cursor to the image surface, and press insert to insert a dummy surface between the primary and the image plane. The new surface will be converted to a fold mirror shortly. The purpose of the dummy surface is simply to locate the fold mirror. Enter a thickness of -200 on the new dummy surface, to maintain the total distance from mirror to image plane at -1000.0.

Now click on Tools, Add Fold Mirror, and set the "fold surface" to 2, and click OK. The resulting spreadsheet display should be similar to the following table (some of the columns have been omitted for clarity).

PRESCRIPTION FOR NEWTONIAN TELESCOPE EXAMPLE

Surf	Surface Type	Radius	Thickness	Glass	Conic	Tilt About X
OBJ	Standard	Infinity	Infinity		0.00000	
STO	Standard	-2000.000	-800.000	MIRROR	-1.00000	
2	Coord Break		0.000			45.00000
3	Standard	Infinity	0.000	MIRROR	0.00000	
4	Coord Break		200.000			45.00000
IMA	Standard	Infinity			0.00000	

Now we would like to view our new folded mirror Newtonian system. The layout used previously will no longer work (it only works for rotationally symmetric systems), instead we use the 3D-layout by choosing Analysis, Layout, 3D Layout. Once the 3D layout appears, use the left, right, up, down, page up, and page down keys to rotate the image around. ZEMAX allows the image to be rotated interactively. One possible projection is shown in Figure E3-2.

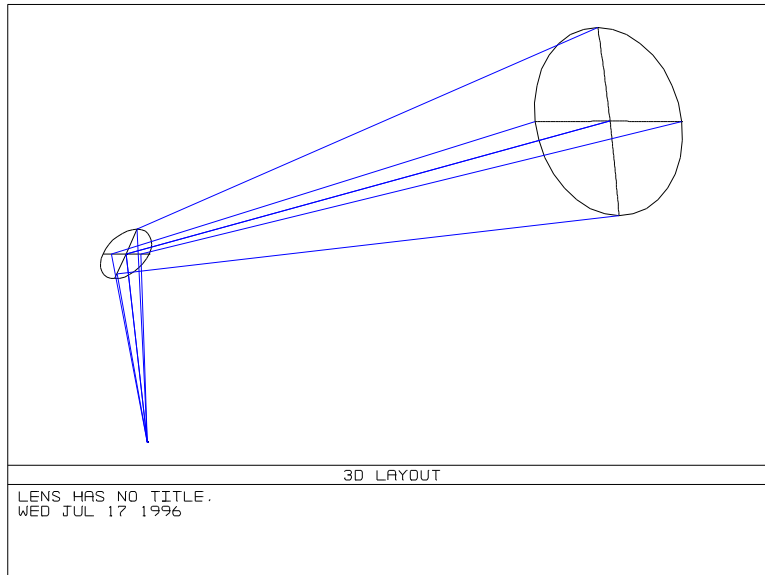


Figure E3-2.

This design projection could be improved several ways. First of all, the rays which are coming from the object to the mirror could be shown. Also, the rays which strike the back of the fold mirror should be vignettted, and not allowed to proceed to the image plane. This is an important effect for real systems, since the rays cannot physically travel through the mirror in usual optical systems.

First, we add a dummy surface in front of the stop by inserting a new surface with the cursor on surface 1. Now, make the thickness of this dummy surface 900. Now, double click on the word "Standard" on surface 1. On the dialog box, select "Circular Obscuration" for an "Aperture Type". This allows the placement of an obscuration in the beam, which accounts for the fold mirror blocking some of the beam. For a "Max Radius" enter 40, then click on OK. Now update the 3D layout. The system now looks more realistic, as shown in Figure E3-3. If not all of the surfaces are visible, select Settings, and change the first and last surface options to 1 and 6, respectively, or click on Reset, then OK.

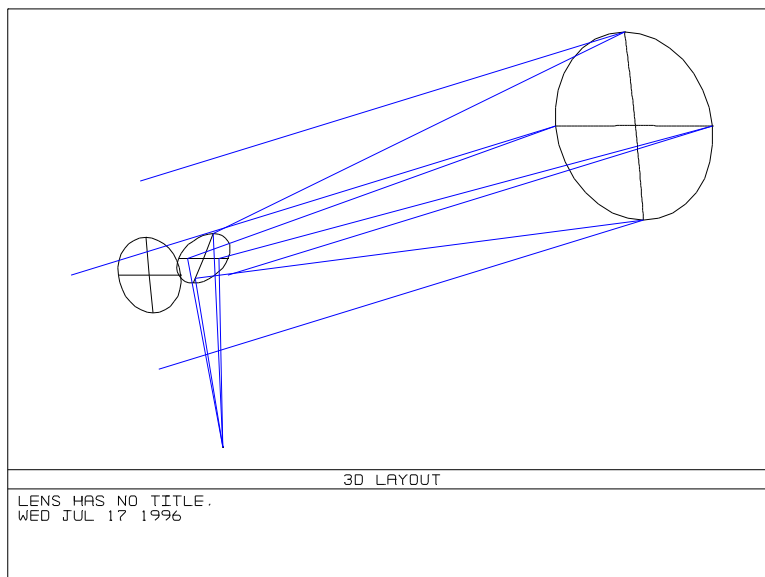


Figure E3-3.

The procedure outlined here is all that is required to use folding mirrors. Coordinate breaks can also be used to tip, tilt, and decenter optical components other than mirrors. Entire optical sub-assemblies may be moved, see some of the example designs included with ZEMAX for sample systems.

## Lesson 4: a Schmidt-Cassegrain with aspheric corrector

What you will learn: Using polynomial aspheric surfaces, obscurations, apertures, solves, optimization, layouts, MTF plots.

This lesson is the complete design of a Schmidt-Cassegrain with a polynomial aspheric corrector plate. The design will be for use in the visible spectrum. We would like an aperture of 10 inches, and a back focus of 10 inches (from the back of the primary to focus).

It is easier to start this design with just the corrector and the primary, and so we begin by inserting two surfaces after the stop. Select System, General, and enter 10 for an aperture value. On this same screen, change the units from "Millimeters" to "Inches".

Select System, Wavelengths to reach the Wavelength Data screen, and set the three wavelengths of .486, .587, and .656 with the .587 being the primary wavelength. This can be done with a single step: click on the "Select ->" button on the bottom of the wavelength dialog box.

For now, we will use the default field angle of 0 degrees. Enter the data in the following table on the spreadsheet. The stop is placed at the center of curvature of the primary; this is done to eliminate field aberrations (like coma) and is characteristic of the Schmidt design.

### STARTING PRESCRIPTION FOR SCHMIDT CASSEGRAIN

Surf	Radius	Thickness	Glass
OBJ	Infinity	Infinity	
STO	Infinity	1.000000	BK7
2	Infinity	60.000000	
3	-60.000000	-30.000000	MIRROR
IMA	Infinity		

Now perform a layout plot to verify everything is correct. The standard 2D layout will work fine. You should see a plot like Figure E4-1.

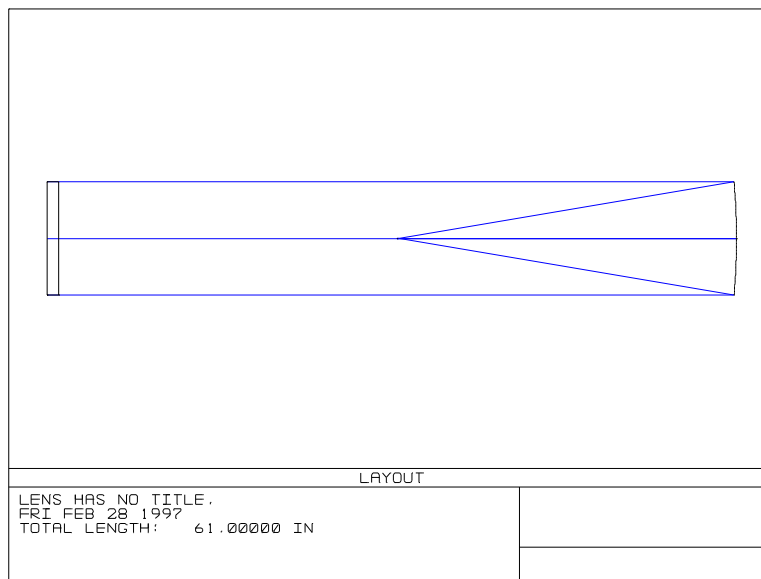


Figure E4-1.

Now we will add the secondary mirror, and place the image plane. We will let ZEMAX calculate the correct curvature for the secondary later. For now, modify the spreadsheet to support one new surface as shown in the following table.

## INTERMEDIATE PRESCRIPTION FOR SCHMIDT CASSEGRAIN

Surf	Radius	Thickness	Glass
OBJ	Infinity	Infinity	
STO	Infinity	1.000000	BK7
2	Infinity	60.000000	
3	-60.000000	-18.000000	MIRROR
4	Infinity V	28.000000	MIRROR
IMA	Infinity		

Notice we have decreased the distance from the primary to -18. This will keep the size of the secondary down. The distance to the image plane is now 28, which is exactly 10 inches behind the primary. A variable has been added to the radius of surface 4; we will allow ZEMAX to find the correct curvature. Since no curvature has been entered yet, the image is out of focus. Check your work by updating the layout. It should look like Figure E4-2.

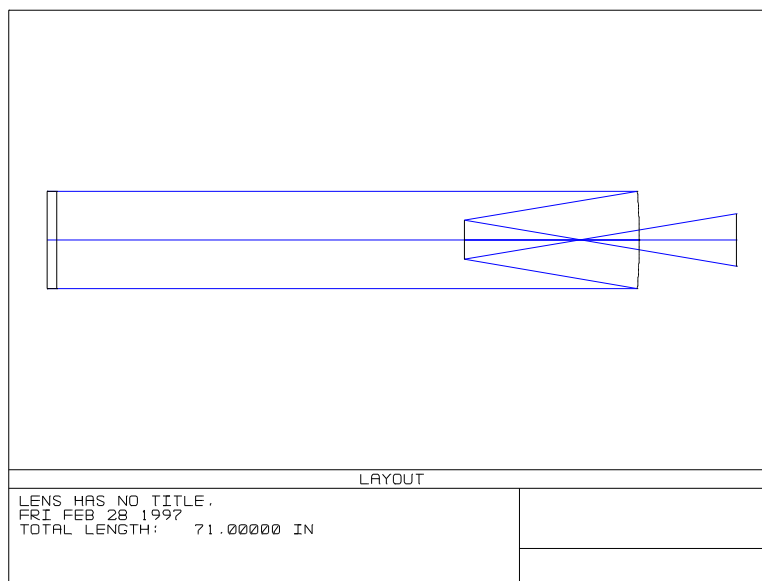


Figure E4-2.

Now select Editors, Merit Function, to display the merit function editor. Select Tools, Default Merit Function from the merit function editor window menu. Click on "Reset", then change the "Rings" option to "5". Now click on OK. The rings option determines the sampling density of the rays, and this design requires more than the default 3.

Select Tools, Optimization. Select "Automatic" and the merit function will quickly drop to about 1.3. This is the residual RMS wave error. Click on "Exit", and then select System, Update All. The radius of the secondary has been changed from infinity to -41.83. Now perform an OPD plot by selecting Analysis, Fans, Optical Path Difference. The OPD plot shows defocus and spherical, and looks like Figure E4-3. Note there is roughly 4 waves of aberration still to correct.

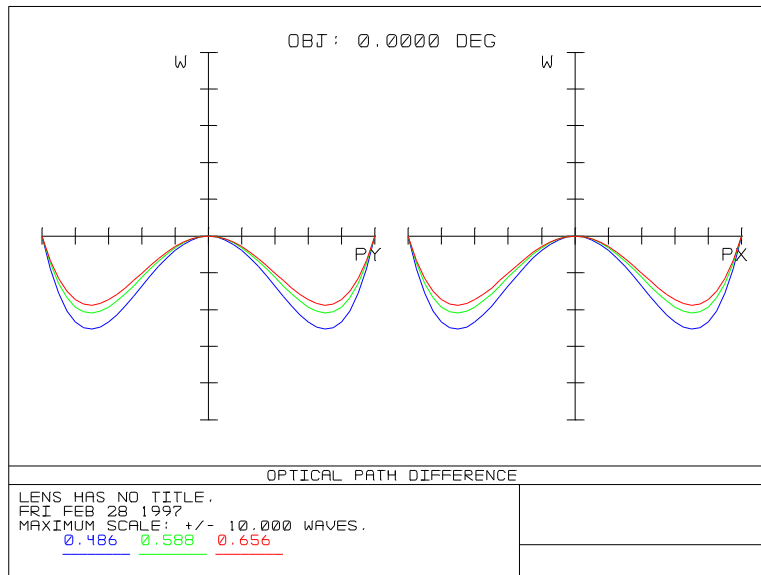


Figure E4-3.

Now click on the "Standard" surface type of surface 1 (the stop surface). From the dialog box that appears, select "Even Asphere". This surface type permits designation of polynomial aspheric coefficients for aspheric correction. Click on OK. Now, move the cursor to the right until you reach the column "4th Order Term" on surface 1, and type Ctrl-V. This sets a variable on this parameter, currently zero. Also place variables on the "6th Order Term" and "8th Order Term" coefficients. Now select Tools, Optimization, and click on "Automatic". In a few seconds, the merit function will drop substantially as ZEMAX balances the various orders of spherical aberration. Click on Exit.

Now update the OPD plot again, it is shown in Figure E4-4. The spherical aberration has been reduced substantially. Note the limiting aberration now is chromatic aberration; with each wavelength having a different amount of spherical. This is called spherochromatism, and we will correct it shortly.

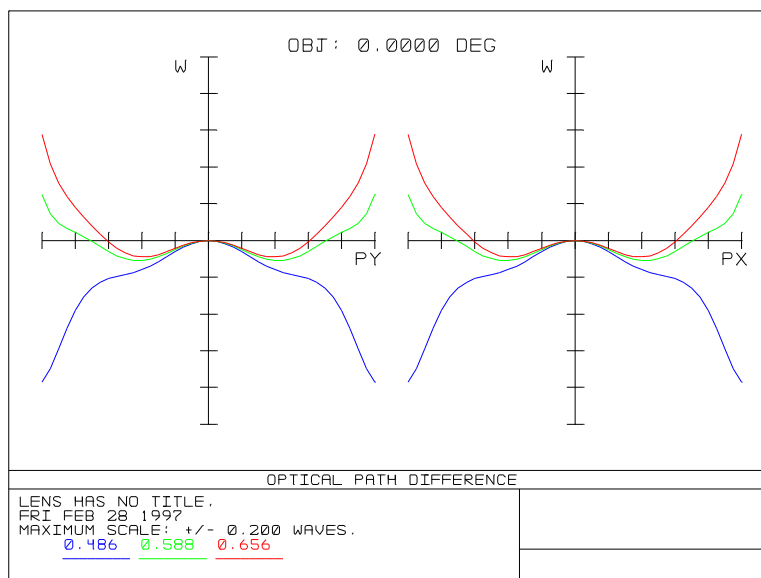


Figure E4-4.

There is a little bit of experience required now to get this design where we want it. To correct the spherochromatism, we need to balance it with axial color. This is a common design approach; balancing higher-order aberrations with lower-order ones in the same family. Here, spherochromatism is the higher-order cousin of first-order axial color. To introduce axial color, we will vary the curvature of surface 1, the front of the corrector plate (this also makes the plate easier to fabricate, for reasons we will not discuss). Set the radius of surface 1 to a variable now.

Now optimize again (Tools, Optimization, Automatic). The merit function will drop again. Now click on Exit, and update the OPD plot. It should look like Figure E4-5.

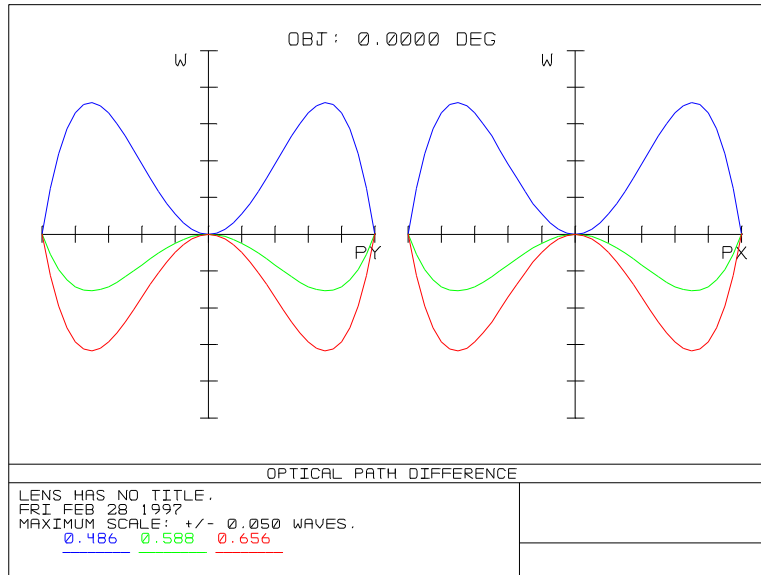


Figure E4-5.

This is the solution we wanted; the residual is less than one-twentieth of a wave! Now we can open up the field angles and fine tune the design. From the main menu, select System, Fields, and set the number of field angles to three. Enter y- angles of 0.0, 0.3, and 0.5 degrees.

If you update and look at the OPD plot now, you will see roughly 1/2 wave of coma at full field. We can easily correct this by re-optimizing. Because we have altered the fields, we must reconstruct the merit function. This is very important! You must appreciate that the default merit function is based upon the fields and wavelengths you have defined; if you change these values, you must reconstruct the merit function.

On the merit function editor, select Tools, Default Merit Function, and change "Rings" to "4". Click on OK.

Now select Tools, Optimization, then click on "Automatic". When it has converged, click on Exit, and update the OPD plot once more. The plot is shown in Figure E4-6, and is a well balanced design.

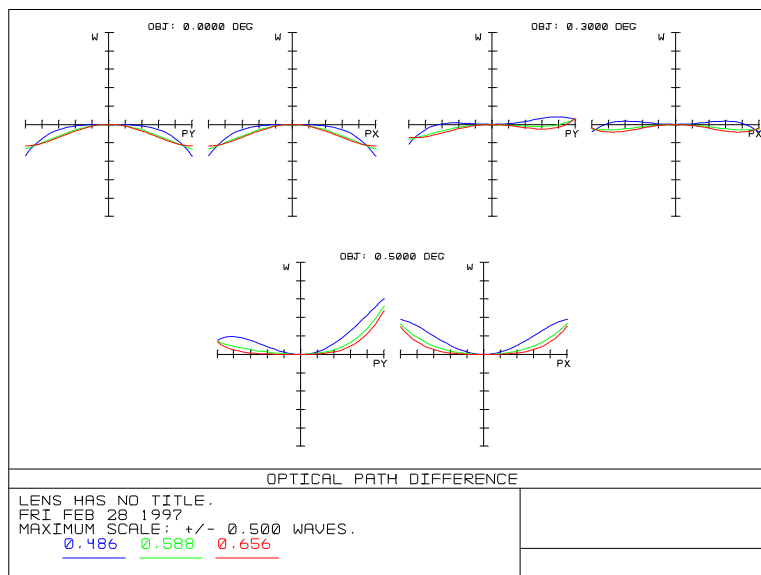


Figure E4-6.

Suppose we are going to use this telescope for taking photographs. We might be interested in the modulation transfer function (MTF), which specifies image contrast as a function of spatial frequency (usually given in inverse

millimeters). To see the MTF plot, from the main menu select Analysis, Diffraction, Modulation Transfer Function. The MTF appears, as shown in Figure E4-7.

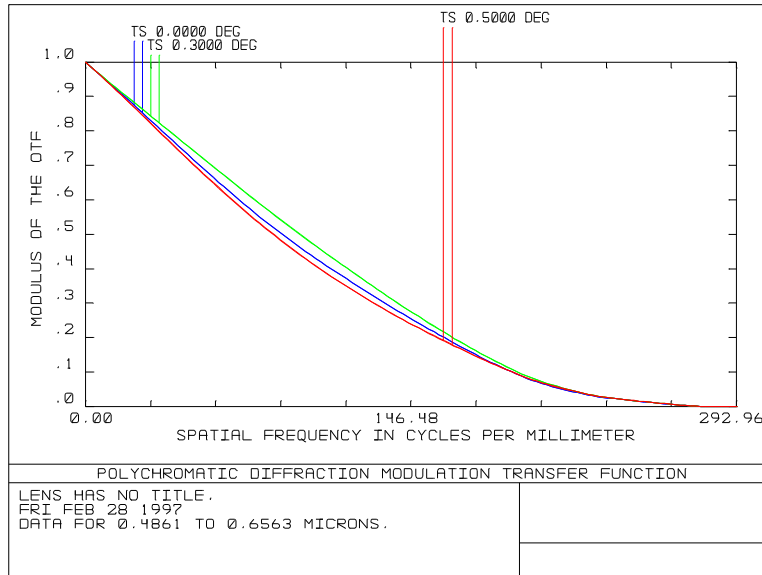


Figure E4-7.

The MTF plot is a very powerful analysis tool. The graph shows tangential and sagittal response for all defined fields. There is, however, something wrong with this plot. A perceptive designer will realize that the data shown is the autocorrelation of a circular pupil. The problem lies in the fact that we have not accounted for the several apertures and obscurations in the system. There is the obscuration caused by the secondary, and there is a hole in the primary. If we add these effects, the performance will drop, particularly at the medial spatial frequencies.

To correct this shortcoming in our analysis, return to the LDE, double click on the first column of surface 3, and select "Circular Aperture" from the list of aperture types. Now move to the "Min Radius" and enter 1.7. This says that all rays must be greater than 1.7 inches away from the axis to pass this surface; this is the "hole" in our primary. Change the "Max Radius" to 6.

The obscuration on the secondary is more complex. It needs to be placed optically in front of the secondary. Since ZEMAX traces rays sequentially, we must place it before the primary. This is easier to do than to explain, so have a little faith and try the following.

While still on surface 3, press the insert key. This inserts a new surface between the corrector and the primary. Change the thickness of this new surface (which is now surface 3) from 0 to 20. Move up one row, and change the thickness of surface 2 from 60 to 40. The total corrector to primary is thus unchanged at 60; we have simply added an intermediate surface. Double click on the first column on this new surface 3, and set the aperture type to "Circular Obscuration", and set the "Max Radius" (not the minimum) to 2.5, then click on OK. Set the semi-diameter of surface 3 to be 2.5 as well. Now update the layout plot. If you did all this correctly, you will see the layout shown in Figure E4-8. The slight gap between the obscuration and the secondary is purely a matter of taste; it is easier to see it this way. You could move it on top of the secondary if you prefer.

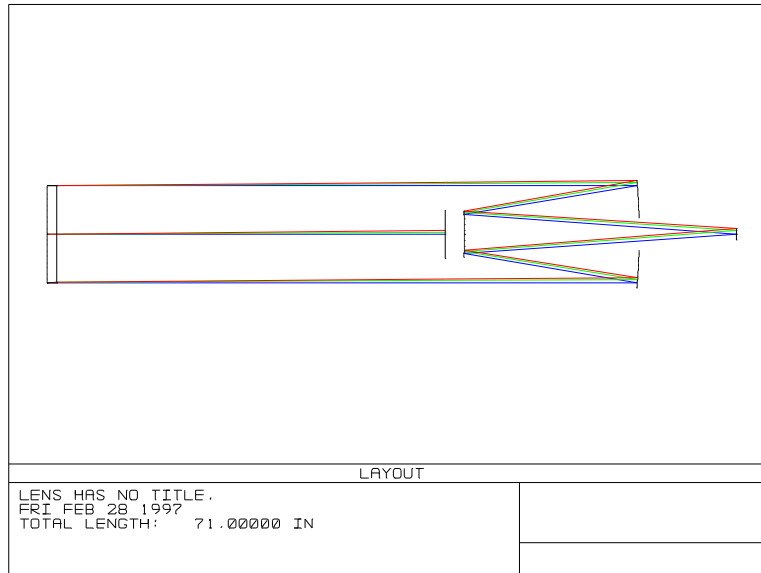


Figure E4-8.

The MTF is now altered by the obscurations, primarily the secondary. Update the MTF window to see the new MTF, it is shown in Figure E4-9.

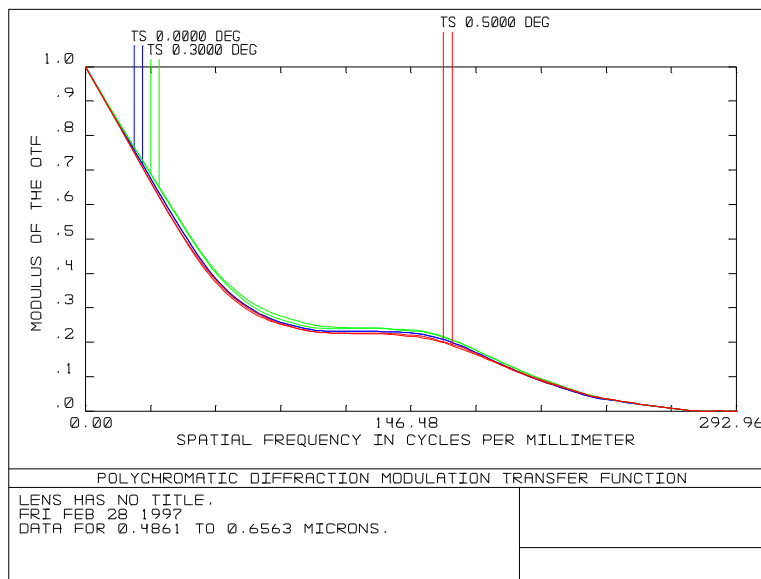


Figure E4-9.

## Lesson 5: a multi-configuration laser beam expander

What you will learn: Using the multi-configuration capability

This example assumes you have mastered the operation of ZEMAX, and understand how to perform basic diagnostics such as ray fans and OPD plots.

Suppose you needed to design a beam expander for use with a laser which operates at  $\lambda = 1.053\mu$ . The input diameter is 100 mm, and the output diameter is 20 mm. Both the input and output are collimated. This is an easy design to do if there is no constraint on the overall length, however, to make it difficult, we will add several constraints:

- 1) Only two lenses may be used.
- 2) The design must be Galilean in form (no internal focus).



- 3) The separation between the lenses must be no more than 250 mm.
- 4) Only 1 aspheric surface is permitted.
- 5) The system must be tested at  $\lambda = .6328\mu$ .

The design task involves not only correcting the aberrations, but doing so at two different wavelengths. However, the system is not used at both wavelengths simultaneously, and therefore, we can shift the conjugates during testing. To begin the design, run ZEMAX. Insert several surfaces, and place variables on them until your lens data editor looks like the following table. Note that the "Focal Length" column is several columns to the right of "Glass". The heading "Focal Length" will only appear after you have changed the surface type from "Standard" to "Paraxial". Not all columns are shown for clarity.

### STARTING PRESCRIPTION FOR LASER BEAM EXPANDER

Surf	Surface Type	Radius	Thickness	Glass	Focal Length
OBJ	Standard	Infinity	Infinity		
STO	Standard	Infinity V	10.000	BK7	
2	Standard	Infinity V	250.000		
3	Standard	Infinity	5.000	BK7	
4	Standard	Infinity V	10.000		
5	Paraxial		25.000		25.000
IMA	Standard	Infinity			

Note the use of the paraxial lens to bring collimated light to focus. Set both the thickness and the focal length to 25.00. Also, set the entrance pupil diameter to 100, and the wavelength (just one) to 1.053 microns. DO NOT enter any other wavelength yet.

The lens thicknesses were set arbitrarily, but will be sufficient for our purposes. The 250 mm is due to our requirement #3. Now select Editors, Merit Function. The merit function should be empty, with just a "BLNK" (for blank, meaning unused) operand shown. At the first line, change the operand type to "REAY". This real ray Y constraint will be used to control the 5:1 beam compression required. Enter "5" for Int1 (this is the surface on which we want to control the ray height) and 1.00 for Py. Enter a target value of 10. This will give us a collimated output beam of 20 mm diameter. For the weight, enter a value of 1.0. Now select Tools, Update; you should see a value of 50 appear in the value column. This is just the entrance pupil radius because we only have parallel plates at this point.

Now from the editor menu bar select Tools, Default Merit Function. Select "Reset", then change the "Start At" field value to 2, then click on OK. This will add operands starting at row 2 in the spreadsheet, which will prevent loss of the REAY operand already entered.

Now from the main menu bar select Tools, Optimization, then click on Automatic. When the optimization is finished, click on Exit. Now from the main menu select Analysis, Fans, Optical Path. You should see the OPD plot shown in Figure E5-1. The performance is poor at roughly 7 waves.

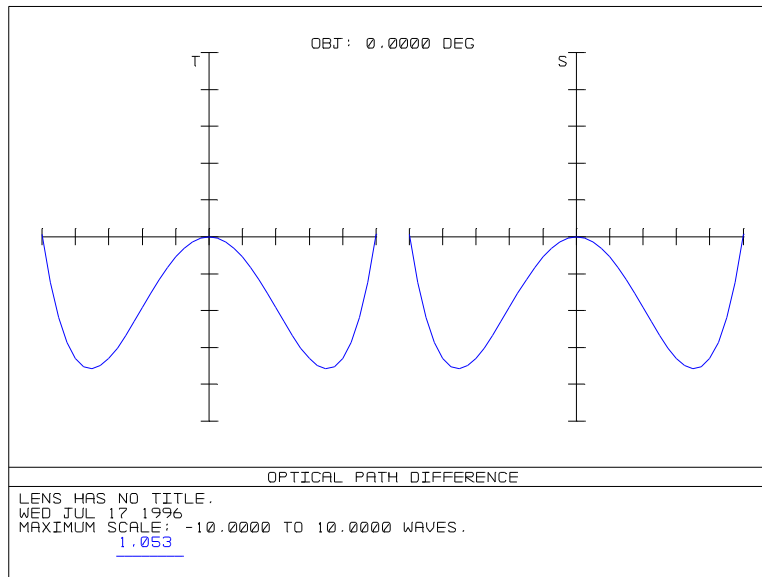


Figure E5-1.

The limiting aberration is spherical (ZEMAX has added defocus in an attempt to compensate it), and fortunately, we have a very effective way of eliminating spherical. Move the cursor up to the conic on surface 1, and make it variable via Ctrl-V. Return to the optimization tool and click on Automatic. The merit function should drop dramatically. Now click on Exit, and update the OPD plot. The performance is easily diffraction limited now, with a reasonable conic. Remove the variability on the three curvatures and the conic, by typing Ctrl-V at each cell.

Before we go any further, save the file by selecting File, Save As. Enter the name MC1.ZMX for the lens, and click on OK. You can open the file later if you make a mistake in the next few steps. Be sure you have removed the variability of the conic as well as the three curvatures.

Go to the Wavelength Data dialog box by selecting System, Wavelengths. Change the wavelength from 1.053 to .6328, and click on OK. Update the OPD plot again. The performance is terrible due to the dispersion of the glass. The aberration shown is simple defocus.

To correct it, we could adjust the lens spacing. Make the 250 mm thickness on surface 2 variable, and then select Tools, Optimization. You should see just one variable listed. Select Automatic. The merit function should decrease. Click on Exit, and then update the OPD plot again. The system has about one wave of aberration at this new wavelength and conjugate. Remove the variable on the thickness of surface 2 by using Ctrl-V again.

We will now use the multi-configuration capability. From the main menu, select Editors, Multi-configuration. Insert a new configuration by selecting Edit, Insert Config. Now double click on the first column of the first row, and select "WAVE" from the drop-down box which appears. On this same dialog box, select "1" for "Wavelength #", then click on OK. This will allow us to define different wavelengths for each configuration. Enter 1.053 under "Config 1" and .6328 under "Config 2". Now press insert to add a new row to the Multi Configuration Editor. Double click on the first column, on the new row 1, then select "THIC" for an operand type. The "THIC" operand will allow us to define different thickness for each configuration. Select "2" from the "Surface" list, and click on OK. Now enter 250 under "Config 1", and 250 under "Config 2". The "2" refers to surface 2, so we are defining the thickness of surface 2 as a multi-configuration value. Make the thickness of surface 2 under configuration 2 a variable by using Ctrl-V.

Now return to the merit function editor. Select Tools, then Default Merit Function. On the dialog box which appears, change the "Start At" value to 1. This will cause the default merit function to begin at row 1. Now click on OK. This will reconstruct the merit function for the multi-config case.

Now the REAY constraint entered earlier needs to be added to the new multi-config merit function. On the first row of the merit function editor, note there is a CONF operand with a 1 in the Int1 column. This operand changes the active configuration to number 1. Under this line, 3 OPDX operands follow. Between the CONF and the first OPDX, insert a new row. Change the operand type on this new row to "REAY". Enter "5" for Int1 (this is the surface on which we want to control the ray height) and 1.00 for Py. Enter a target value of 10. This will give us a collimated output beam of 20 mm diameter. Any operand listed under the CONF 1 operand will be enforced in that

configuration. Under CONF 2, no operand is needed, since we cannot have exactly a 5:1 beam compression at both wavelengths.

Now go back to the lens data editor, and make the curvatures of surfaces 1, 2, and 4 variable, like at the beginning of the session. Also make the conic of surface 1 variable. Go to the optimization screen by selecting Tools, Optimization. Note 5 variables are active (3 curvatures, 1 conic, 1 multi-config thickness). Now click on Automatic, and when finished, click on Exit.

Now double click on the "Config 1" column heading of the multi-configuration editor, and update the OPD plot, and note the performance is excellent at a wavelength of 1.053. Now double click on the "Config 2" column heading, and update the OPD plot (this time for .6328). It is also well corrected. Note that the multi-configuration editor displays the two thicknesses to use for the two wavelengths. The keyboard shortcut Ctrl-A is handy for quickly changing between the two configurations.

More complex multi-configuration applications are possible, but the procedure is the same.

## Lesson 6: fold mirrors and coordinate breaks

What you will learn: A better understanding of coordinate breaks, sign conventions for tilted and decentered systems, implementing fold mirrors. Most of the techniques described in this tutorial are automated by the "Add Fold Mirror" tool, however, a detailed explanation is provided for manual entry of coordinate breaks for completeness.

A previous lesson showed how to design a Newtonian telescope. That lesson introduced the concept of mirrors as well as the coordinate break surface. The import points of that lesson are:

- 1) Thicknesses always change sign after a mirror. After an odd number of mirrors, all thicknesses should be negative. This sign convention is independent of the number of mirrors, or the presence of coordinate breaks.
- 2) Coordinate breaks usually come in pairs, and "sandwich" fold mirrors.

This lesson will show how to manually add fold mirrors to a simple converging beam. Start by selecting File, New to clear out any current lens data. The New function will set the number of fields to one, and the number of wavelengths to one, which suits our purpose. On the Lens Data Editor (LDE), there are three surfaces shown: OBJ, STO, and IMA, which are surfaces 0, 1, and 2. Change the surface type on the STO surface to a paraxial lens by double clicking on the surface type column, and then select "paraxial" from the drop down list. Next, set the thickness of the STO surface to 100, which is the default focal length for the paraxial lens. Next, select System, General, and on the dialog box which appears enter an aperture value of 20 (this will yield an F/5 lens). Click on OK to close the dialog box. Now perform a 3D layout by selecting Analysis, Layout, 3D Layout. You should see a plane surface on the left, and rays converging to a point focus on the right. Any optical system could have been used to generate the converging beam; a paraxial lens surface was used for simplicity.

Now we will add a single fold mirror to direct the converging beam upward. The fold mirror will initially be oriented at 45 degrees. Suppose we want the fold mirror to be 30 mm away from the paraxial lens. Three new surfaces are required: a coordinate break to rotate the coordinate system 45 degrees, then a mirror surface, and finally another 45 degree rotation to follow the reflected beam. This is a crucial point: three surfaces are required to implement a single fold mirror.

To add the three surfaces, click anywhere on the image surface row to relocate the cursor, then press the insert key three times. Change the thickness of surface 1 (the STO surface) to 30. Enter in "MIRROR" for the glass column on surface 3. Change the thickness of surface 4 (the surface prior to IMA) to -70. Note the 70 is negative because after an odd number of mirrors all thicknesses change sign.

Now update the 3D layout window. The graphic will be redrawn, and you will see the light begin to focus, hit the mirror, and bounce backward to a focus 40 mm to the left of the paraxial lens. The fold mirror is not yet tilted.

To tilt the fold mirror 45 degrees, change surfaces 2 and 4 to coordinate breaks by double clicking on the surface type column of those surfaces, and select "Coordinate Break" from the drop down list. Now scroll to the right (using the cursor keys or the scroll bars at the bottom of the LDE) until the parameter columns appear. On surfaces 2 and 4 a series of zero values will appear. Click on the parameter 3 column of surface 4, and the column heading should then indicate "Tilt About X". Double click on this cell (make sure you are on surface 4) and select "Pickup" for a solve type on the drop down list. Set the "From Surface" to 2 and the "Scale Factor" to 1.0. This

will cause the second coordinate break rotation angle to always be the same as the first. Click on OK. Note a "P", indicating a pickup solve, appears next to the cell value.

Now move up to surface 2, and enter a value of 45 in the "tilt about x" column. From the main menu select System then Update All, and you should see the layout shown in Figure E6-1.

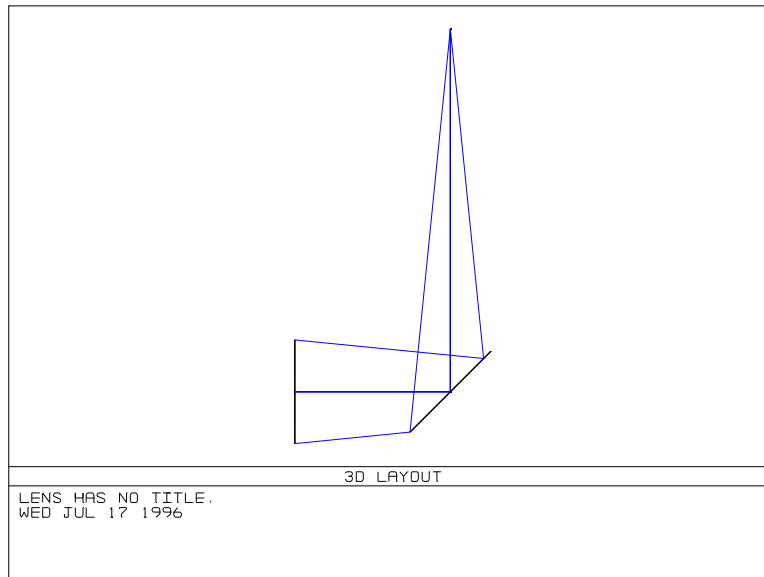


Figure E6-1.

Note that the thickness of the paraxial lens, 30, locates the vertex of rotation for the first coordinate break. The thickness of the coordinate break is 0, which means the mirror is located at the same point. However, the coordinate break has rotated the coordinate system 45 degrees. The mirror itself is not rotated, it is the coordinate system in which it lies that has been rotated. The mirror also has a zero thickness because we want to rotate another 45 degrees before moving on to the next surface. The second coordinate break first rotates another 45 degrees, and then moves the -70 units to focus. Note all tilts and decenters precede the thickness step.

To implement another fold mirror, click on the image surface to locate the cursor there, and press the insert key three times. Change the thickness of surface 4 from -70 to -30, the glass type of surface 6 to MIRROR, and the thickness of surface 7 to +40 (note again the sign change after a mirror). Change the surface types of surfaces 5 and 7 to coordinate breaks, and enter a tilt about x of -45 on surface 5. On surface 7, double click on the tilt about x and place a pick up solve on that parameter. Use a pick up from surface 5, with a scale factor of 1. Update the 3D Layout. It should look like Figure E6-2.

The second series of break-mirror-break folds the beam another 90 degrees, so it is parallel with the original axis. Since we have added pickup solves on the second coordinate break in each series, it is easy to tilt the beam by changing just two numbers. Try entering 30 and -60 on the tilt about x on surfaces 2 and 5, respectively, then choose System, Update All to see the effects on the 3D Layout.

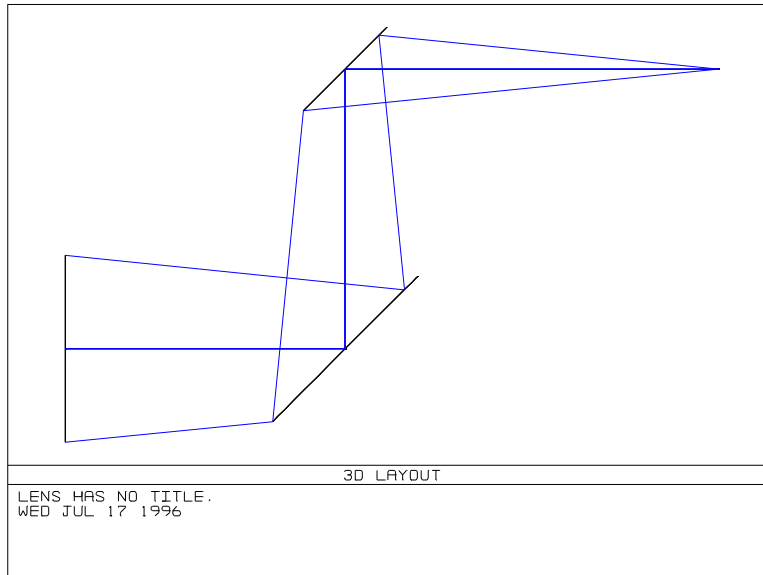


Figure E6-2.

## Lesson 7: the achromatic singlet

What you will learn: Using the Extra Data Editor, optimizing with Binary surfaces. The features used in this lesson are only available in ZEMAX-EE. This lesson assumes you have successfully completed Lessons 1 and 2, and that you know how to perform basic operations like defining wavelengths and apertures.

If you have been designing refractive optics exclusively, you may not believe there is such a thing as an "achromatic singlet". Of course, a mirror is an achromatic singlet, but is also possible to design a hybrid refractive/diffractive component which is corrected for first-order chromatic aberration. The trick is to use a conventional refractive singlet which has a diffractive surface etched on one side. The curvatures on the faces of the singlet provide the bulk of the focusing power, while the weak diffractive component provides enough dispersion to compensate for the dispersion of the glass.

To review, a singlet lens of focal length  $f$  has an optical power of  $\phi = f^{-1}$ . The variation in power over the  $\lambda_F - \lambda_C$  wavelength range is given by the Abbe number  $V$  of the glass of which the singlet is composed:

$$\Delta\phi = \frac{\phi}{V}.$$

Note that the dispersion is relatively weak for most glasses; for example, BK7 has a  $V$  value of 64.2. The variation in power is therefore about 2% of the total power.

Diffractive optics use the direct manipulation of the phase of the wavefront to add optical power to the beam. For a diffractive surface with a quadratic phase profile, the phase is given by

$$\Psi = A r^2,$$

where  $A$  has units of radians per unit length squared and  $r$  is the radial coordinate. It is easy to show the power of a surface with this phase profile is given by

$$\phi = \frac{\lambda A}{\pi}.$$

Note that the power of this surface varies linearly with wavelength! Over the same wavelength range that the refractive singlet power varies by 2%, the diffractive optic power varies by nearly 40%. Furthermore, the sign of the dispersion can be adjusted by changing the sign of the constant  $A$ . This property can easily be exploited by adding a slight amount of positive power to the refractive component, and then compensating by adding a slight negative power to the diffractive component. The amount of power added can be selected to balance the axial color.

To see how easily this can be done, start ZEMAX and choose File, New. Then in the Lens Data Editor (LDE) change the Standard STO surface to a Binary 2 by double clicking on the word "Standard", choose "Binary 2" from the drop down list of surface types, and then click on "OK". Move the cursor down to the IMA row and press insert to add one more surface. Change the thickness of the new surface 2 to 100, and the thickness of surface 1 (the STO surface) to 10. Enter in "BK7" for a glass type on surface 1. Then select System, General and enter an Aperture Value of 20, and click on "OK". Finally, choose System, Wavelengths and enter in 3 wavelengths: .486, .587, and .656. Make the .587 wavelength the primary wavelength.

First we will see how good the performance is if we design a convex-plano singlet. Make the radius of surface 1 variable using Ctrl-V. Then choose Editors, Merit Function. On the Merit Function Editor (MFE) choose Tools, Default Merit Function. The default is fine, so click on "OK". Close the MFE window.

Now select Tools, Optimization. Click on "Automatic", and it will drop down. Click on "Exit" and select Analysis, Fans, Optical Path. The plot will appear, and it should show about 8 waves of aberration. Note the axial color completely dominates this design. There is also considerable spherical aberration and defocus.

That is about as good as it gets for a simple convex-plano singlet. To improve the design, select Editors, Extra Data. On the EDE, enter in a "1" for the "Max Term #", and a "10" for the "Norm Aper". Make the third "Coeff on P<sup>2</sup>" column variable using Ctrl-V. Now from the main menu select Tools, Optimization, and it should show two variables (the first radii and the diffractive power). Click on "Automatic" and the merit function will drop quickly as ZEMAX uses the diffractive power to correct the axial color. Click on "Exit".

Now choose System, Update All, and the OPD fan will be redrawn with a maximum aberration of about 1 wave. Most of what remains is secondary spectrum and spherical aberration. What about that spherical aberration? We can correct it using a higher order diffractive term. Return to the EDE, change the maximum term number to 2, make the new fourth order term a variable, and reoptimize. Update the OPD plot, and see the wavefront aberration is well below 1 wave.

Diffractive optics hold extraordinary promise for optical designers. Systems which incorporate diffractive elements have the potential to be lighter and of superior performance than purely refractive systems. However, as of this writing diffractive optics are difficult to fabricate for use in the visible spectrum, and suffer from higher scattering losses than conventional optics.

This page intentionally left blank.

---

**New**

---

*Purpose:*

Clears the current lens.

*Discussion:*

This option restores ZEMAX back to the start-up condition. Windows currently open will remain open. If the current lens has not been saved, ZEMAX will prompt to save the lens before it is destroyed.

---

**Open**

---

*Purpose:*

Opens an existing ZEMAX lens file.

*Discussion:*

This option opens a new lens file. Windows currently open will remain open. If the current lens has not been saved, ZEMAX will prompt to save the lens before it is destroyed. See also "Append".

---

**Save**

---

*Purpose:*

Saves lens file.

*Discussion:*

This option saves the current lens file. To save the lens under a different name or directory, use Save As.

---

**Save as**

---

*Purpose:*

Saves lens file, optionally with another name.

*Discussion:*

This option saves the current lens file with the option to change the current file name or directory.

---

**Append**

---

*Purpose:*

Appends previously stored lens data to the current lens file.

*Discussion:*

This option is similar to the "Open" command, but the current lens data is not disturbed. After the file to append is selected, ZEMAX will prompt for the surface number where the new lens data should be inserted. New surfaces will be inserted into the lens prescription to make room for the new data. The dialog box also has an "Ignore Object" checkbox which by default will ignore the object thickness of the new lens. Therefore, the new lens data to be appended will start at surface 1 rather than surface 0.

Although this feature can save a lot of typing, the resulting lens may have some extra surfaces, or may need a little hand editing to achieve the desired result.



## Environment

---

ZEMAX allows users a number of options that can be set and saved so that these options are selected automatically when ZEMAX is run. The main configuration file is ZEMAX.CFG; this file can be deleted to return the startup configuration to the default state. The environment options are divided into groups: Address, Directories, Graphics, Text, Editors, Printing, and Button bar.

### **Address**

#### *Purpose:*

These settings determine how the "address" box will appear. The address box can be used to display user defined text such as a company name or drawing number. The address box appears on the bottom right corner of most graphics.

#### *Settings:*

Item	Description
Address Line 1	The first line of text to appear in the address box.
Address Line 2	The second line of text to appear in the address box.
Address Line 3	The third line of text to appear in the address box.
Address Line 4	The fourth line of text to appear in the address box, unless file name or zoom position is chosen instead.
Show Line 4 As	Choose either the entered text, the lens file name, or the zoom position.
Hide Address	If checked the address box will not appear.

### **Directories**

#### *Purpose:*

These settings determine where ZEMAX will place or search for certain types of files.

#### *Settings:*

Item	Description
ZEMAX Path	The default directory where ZEMAX will look for files used by the program, such as glass catalogs and stock lens catalogs.
Output Path	The default directory where all text and graphics output will be sent.
Lens Path	The default directory for lens files.
ZPL Path	The default directory for ZPL Macros.
Stock Path	The default directory for the stock lens directories. All stock lenses must be stored in subdirectories within the Stock Path directory.

### **Graphics**

#### *Purpose:*

These settings determine the size, color, and behavior of most ZEMAX graphics windows. See also the text windows section for the Date/Time option.

#### *Settings:*

Item	Description
B/W Screen	By default, ZEMAX draws color graphics to the screen. If this checkbox is selected, then all graphics will be black and white.
B/W Plots	By default, ZEMAX prints color graphics to printers. If this checkbox is selected, then all printed graphics will be black and white. Only printers which support color will print in color, otherwise, gray scales will be used to simulate color.
Show Options First	If this checkbox is selected, then the settings box will appear before any analysis graphic is computed and displayed.

Item	Description
Win X,Y Size	The default x and y size for graphics windows in pixels. This can be adjusted to suit the monitor size and resolution.
Background	The background color on graphics windows can be selected from this drop down list.
Metafiles	ZEMAX generates several different types of Windows Metafile formats. Metafiles are used to copy graphics to the clipboard, or to copy graphics to files on disk, where the graphics may then be pasted or imported into other Windows programs. Most 16 bit Windows 3.1 applications used the "16 Bit Standard" format. However, some Windows 3.1 applications use a variation called the "16 Bit Placeable" format. Newer, 32 bit applications use the "32 Bit Enhanced" format. When a 32-bit format is used, the extension used is EMF, for enhanced metafile format.
Metafile Pen Width	Determines the pen width in device dependent pixels for graphics which are exported as Windows Metafiles, via either the clipboard or as disk files.
Aspect Ratio	The default aspect ratio for ZEMAX graphic windows is 3 x 4, which fits well on a standard 8.5 x 11 inch paper printer. For 11 x 17 inch printing, a 3 x 5 aspect ratio fits better. This option selects the aspect ratio for both printing and screen graphics.

### **Colors**

The colors dialog box is used to define the pen colors used by ZEMAX for drawing graphics. Each pen is used by the corresponding wavelength of field position when graphing ray fan data, spot diagrams, and other data. Wavelength number 1 uses pen 1, wavelength 2 uses pen 2, and so on. Field position 1 also uses pen 1, field position 2 uses pen 2, and so on. The Red, Green, and Blue values define the pen color. Each entry must be between 0 and 255. Pens are defined using the RGB values for 24 bit color (16 million colors), but will only display up to the resolution provided by the graphics hardware present. The resulting colors are shown to the right of each RGB value.

### **Text**

#### *Purpose:*

These settings determine characteristics of the text windows. The Date/Time setting also affects graphic windows.

#### *Settings:*

Item	Description
Screen Font Size	The font size in points to be used for text displayed in a window. The default is an 8 point text.
Date/Time	Select either None for no date or time, Date for the date only, and Date/Time for both the date and the time to be printed on plots and graphics.

### **Editors**

#### *Purpose:*

These settings determine characteristics of the spreadsheet editors.

#### *Settings:*

Item	Description
Decimals	Decimals will alter the number of decimal digits displayed on the lens data editor. Selecting "Compact" will vary the number of decimals displayed to minimize the space required to display numbers.
Font Size	The font size in points to be used for text. The default is an 8 point text.
Auto Update	Controls how and when ZEMAX performs updates on the editor data. The "None" setting means pupil positions, solves, and other lens data editor data are not updated until "Update" is selected from the "System" menu. The "Update" setting will cause an update to be performed whenever new data is typed in to the lens, extra, or multi-configuration editors. "Update All" causes all windows to be updated whenever new data is typed in the editors. See the "System menu" chapter for details on Update and Update All.

Item	Description
Show Comments	If checked, then the surface comments column is displayed in the Lens Data Editor. Otherwise, the column is hidden from view.
Undo	The Undo feature has three seletable states: None, Memory 1 Step, and Disk Multi Step. See the Chapter "Editors Menu" for details on the Undo feature.
LDE Cell Size	The width in characters of a single edit cell in the Lens Data Editor. Larger cells will mean fewer columns can be displayed, but the data is generally more readable.
MFE Cell Size	The width in characters of a single edit cell in the Merit Function Editor.
MCE Cell Size	The width in characters of a single edit cell in the Multi-Configuration Editor.
EDE Cell Size	The width in characters of a single edit cell in the Extra Data Editor.
TOE Cell Size	The width in characters of a single edit cell in the Tolerance Data Editor.

## **Printing**

### *Purpose:*

These settings determine characteristics of the printed output.

### *Settings:*

Item	Description
Skip Print Dialog	If this box is checked, then ZEMAX will not present the "Print" dialog box which allows selection of printer type and other options when "Print" is selected from any window. If the box is unchecked, then the default printer settings will be used as specified by the Windows Print Manager.
Rotate Plots	If selected, this will cause all printed graphics to be rotated by 90 degrees. This allows a landscape type orientation while printing in portrait mode. See the discussion section below.
Pen Width	The width of the pen in pixels. Values of 0 result in fine lines, larger values yield thicker lines.
Plot Width	See the discussion section below.
Left Margin %	The left margin of the plot as a percentage of the maximum possible plot width. Only affects graphics printing.
Right Margin %	The right margin of the plot as a percentage of the maximum possible plot width. Only affects graphics printing.
Top Margin %	The top margin of the plot as a percentage of the maximum possible plot height. Only affects graphics printing.
Bottom Margin %	The bottom margin of the plot as a percentage of the maximum possible plot height. Only affects graphics printing.
Print Font Size	The font size in points to be used for text when a text window is printed. The default is an 8 point text.
Text Margin	The left margin to be used when printing text files, measured in characters.

### *Discussion:*

The "Plot Width" control is different from most other settings in ZEMAX, because it is actually used to tell ZEMAX how wide the plots are rather than how wide they should be. Every printer will print ZEMAX graphics at a different size. To get accurate scaling on layout plots and scale bars, ZEMAX must be informed as to how big the plots are when printed. With this information, ZEMAX can print exactly at 1:1, or 2:1, and so on.

Pressing the "Plot Width" button will invoke the Print dialog box. This is the same box that is displayed when printing graphics from ZEMAX. This box allows selection of the printer driver, and usually allows selection of printer specific items such as resolution, orientation, and perhaps other options which vary from printer to printer. Using the Print dialog box, select the driver and mode you normally like to print in. Note ZEMAX prints graphics in "landscape mode" by rotating the plot 90 degrees and then using the portrait setting. This is done because virtually all printers default to portrait mode. It is therefore better to stay in portrait mode all the time, and use the rotate plots feature. ZEMAX will use these settings to determine how wide the actual plot will be when printed, and place this width, in inches, in the "plot width" edit box. Note that the plot width is automatically recalculated

for the default printer settings whenever the print orientation or margins are adjusted, or the "Reset" button is pressed.

Once the actual plot width has been computed, the scaling on layout plots will be very accurate. However, it will only be accurate if the same printer driver and mode settings are actually used for printing. If a different printer driver or mode is selected at print time, then the plot width scaling will not automatically be recalculated. To get correct scaling when printing to a new printer or using a different printer mode, the plot width environment variable must be reset using the procedure described above.

Lastly, there are times when it may be required to override the default settings for the plot width. For example, if the final printed output is to be reduced to a certain size for inclusion in another document, then the known final size can be used to get exact scaling at the final image size. To accomplish this, simply enter the known final image width in inches in the plot width edit box, and press Save. All subsequent printing will be done assuming the final image size specified.

Note that the precise scaling controls specified only affect the Layout and Element Drawing features, since all other plots are scale independent.

### **Button Bar**

*Purpose:*

These settings determine which functions are available from the button bar at the top of the ZEMAX main screen.

*Discussion:*

There are 20 buttons which can be set to invoke any of the ZEMAX main menu features. Each button has an identical list of menu options which can be associated with the button. Selecting "Off" will cause the button to disappear.

### **Status Bar**

*Purpose:*

These settings determine which values are displayed in the status bar at the bottom of the ZEMAX main screen.

*Discussion:*

There are 4 regions which can be set to display various data, such as EFL, EPD, F/#, etc.

## **Exit**

---

*Purpose:*

Exits ZEMAX.

*Discussion:*

If the lens has been modified, ZEMAX will prompt to save the lens. If not, the program terminates.

## **Recently used files**

---

The six most recently used lens files are listed at the bottom of the file menu. Selecting one of these files from the list will cause the file to load. This is simply a shortcut for using the File, Open command.

This page intentionally left blank.

## Lens Data

---

The Lens Data Editor is the primary spreadsheet where the majority of the lens data will be entered. This data includes the radius of curvature, thickness, and glass for each surface in the system. Single lenses are defined by two surfaces (a front and a back), and the object and image also each require one surface. This fundamental data can be entered directly on the spreadsheet. When the Lens Data Editor is displayed, data can be entered on the spreadsheet by simply typing in the required values at the highlighted bar. Each column is labeled by the data type, and each row represents a single optical (or perhaps a dummy) surface. The cursor keys move the highlighted bar to whichever column is desired. Moving the cursor continuously to the right or left causes the display to scroll, which provides access to the other columns of data such as semi-diameters, conic constants, and "parameter" terms whose values depend upon the surface type. The display will scroll left to right or right to left. The page up and page down keys will move the bar to the top and bottom of the display. The display will also scroll up and down if required, when the number of lens surfaces is sufficiently large.

### **Inserting and deleting surfaces**

Note that initially (unless a lens has been loaded) three surfaces are shown; the object, stop, and image surfaces. The object and image surfaces are permanent, and cannot be deleted. However, other surfaces can be inserted and deleted by using the Insert and Delete keys. No surface can be inserted in front of the object, or behind the image. In this context "in front of" means a smaller surface number, and "behind" means a larger surface number, in the sense that the light reaches the various surfaces sequentially. ZEMAX numbers surfaces from the object, being surface zero, upwards to the last surface, being the image surface.

To enter values in the spreadsheet, simply move the cursor to the correct cell, and start typing. To edit values currently displayed, press the backspace key. Once you are editing the cell contents, you can use the left/right cursor keys and the home and end keys to navigate around in the text. When the data has been changed, press any cursor key, or click on any other cell, or press enter.

There are also a few shortcuts available only on the Lens Data Editor. To add a value to the current value, type a plus sign before the number. For example, if the displayed number is 10, typing "+5" and pressing enter will change it to 15. The symbols "\*" and "/" also work. To subtract a value, use the minus sign followed by a space. For example entering "- 5" would change a 17 to a 12. Note the space between the "-" and the "5". If no space was entered, the program assumes you are entering a new value which happens to be negative. Entering "\*-1" will change the sign of the number.

### **Cutting, Copying, and Pasting surface data**

See the discussion on the "Edit" menu below.

### **Entering surface comments**

Each surface has a comment field which can be used to enter up to 32 characters of user text. The comment column is used for improving the readability of the lens prescription, and has no affect on ray tracing. Some analysis features also display the surface comments. The entire comment column can be hidden, see the "Options" menu description below.

### **Entering radii data**

To enter or change the radius of curvature of a surface, move the cursor to the desired cell, and type in the new value. Radii data is always entered and displayed in lens units, which have dimensions of length.

### **Entering thickness data**

To enter or change the thickness of a surface, move the cursor to the desired cell, and type in the new value. Thickness data is always entered and displayed in lens units, which have dimensions of length. The thickness of a surface is the distance to the next surface. The only thickness that is not used is the thickness of the image plane.

Thicknesses always change sign after a mirror. After an odd number of mirrors, all thicknesses should be negative. This sign convention is independent of the number of mirrors, or the presence of coordinate breaks. This fundamental convention cannot be circumvented through the use of coordinate rotations of 180 degrees.

### **Entering glass data**

The material used for each surface is usually specified by entering the name of the glass in the "Glass" column on the Lens Data Editor. The glass name entered must be in one of the currently loaded glass catalogs. The default catalogs are the "Schott" and "Misc" catalogs; others are available. To use multiple glass catalogs, or to review, edit, or append the glass catalogs, see the Chapter "Using Glass Catalogs". To specify that a particular surface be a mirror, use the glass name "mirror".

There is an optional "/P" command that may be appended to the glass name when entering a new glass. This option will cause ZEMAX to alter the curvatures before and after the surface to maintain a constant power at the front and back vertices of the surface. For example, if the glass type is already BK7, entering a new glass type of "SF1/P" will change the glass type to SF1 while adjusting the radii of the surface and the surface after, to maintain constant power. ZEMAX keeps the power at the vertices constant, however, the overall lens power will change slightly due to the change in the optical thickness of the glass. This effect is generally very small for thin lenses.

### **Entering semi-diameter data**

The default semi-diameter is automatically calculated to be the radial clear aperture required to pass all rays from all field points. If any value is entered for the semi-diameter, then the new number will be retained and a "U" will be placed next to the value. This indicates the semi-diameter is user defined. The semi-diameter value only affects how the surface is drawn on a layout plot, not whether or not that surface vignettes rays. See the section "Specifying surface apertures".

### **Entering conic data**

Conic data is permitted on many different surface types. To enter or change the conic constant of a surface, move the cursor to the desired cell, and type in the new value. Conic constants are always dimensionless. See the section "The standard surface" in the Chapter "Surface Types" for a discussion of how conic surfaces are defined.

### **Entering parameter data**

Parameter data consists of up to 8 additional numeric values which define the properties of certain surface types. For more information on parameter data, See the Chapter "Surface Types".

### **Defining the stop surface**

The stop surface can be any surface in the system, except for the object and image surfaces. To change the stop surface, double click on the left most column (the one with the surface numbers listed) on the surface which will be the new stop surface. The surface type dialog box will appear. Click on the button labeled "Make surface stop". The dialog box disappears and the surface now will display the "STO" label instead of the surface number.

It is important to define the stop surface such that the entrance pupil is on the same axis as the object surface. You can insure this condition by placing the stop surface of the optical system before any coordinate breaks, obscuration decenters, holograms, gratings, or other components which can alter the optical axis. If your system is symmetric for rotations about the optical axis, then this limitation does not apply. Only systems that use surfaces which can tilt or decenter the optical axis should have the stop placed before any such surfaces. If coordinate breaks are used, but only to implement fold mirrors in an otherwise axial system, then the pupil locations will be correctly computed even if the stop is placed after the fold mirrors.

In certain systems it is not possible to place the aperture stop before coordinate breaks. In this case, ray aiming must be used. Ray aiming is discussed in the Chapter "System Menu".

### **Selecting surface types**

ZEMAX models planes, spheres, and conics; all of these surface types are grouped under the category of standard surface. By double clicking on the left most column in the Lens Data Editor, the surface data dialog box will appear. On this dialog box is a list of surface types. Select the appropriate surface type from the drop down list. ZEMAX supports many different types of surfaces in addition to the standard surface. The types are discussed in detail in the Chapter "Surface Types". Many optical designs only use the standard surface type.

## **Specifying surface apertures**

Surface apertures are used to correctly account for the effects of vignetting. There are eight types of surface apertures in ZEMAX: no aperture, circular aperture, circular obscuration, rectangular aperture, rectangular obscuration, elliptical aperture, elliptical obscuration, and spider. The aperture and obscuration types define regions which pass or stop rays, respectively. When an aperture is defined on a surface, ZEMAX will display an asterisk "\*" symbol next to the surface number or descriptor in the Lens Data Editor.

More than one aperture may be described at a given optical element by inserting a dummy surface with zero thickness at the desired location, and then setting additional apertures on that surface. This is useful for constructing complex apertures.

The default aperture type is "none" which is shown on the surface data dialog box. The surface data dialog box can be reached by double clicking on the left most column in the Lens Data Editor. When "none" is selected for the aperture type, all rays which can be refracted or reflected by this surface are allowed to proceed. Passing a surface is completely independent of the "Semi-Diameter" value in the Lens Data Editor. That setting is only used for drawing the lens elements, not for determining vignetting. To clear any aperture to this default state, or to change the current aperture type, select another aperture type on the surface data dialog box.

One of the options is "Circular Aperture". An aperture is a hard stop for the rays which strike the surface outside of the specified radial apertures. Any ray passing beyond (at a greater radius than) the outer radii and within (at a lesser radius than) the inner radii will be terminated, and not allowed to proceed. If the ray is between the inner and outer radii, then the ray will be allowed to proceed.

The next aperture type is called "Circular Obscuration". An aperture can be used to describe a blocked ring, or central obscuration; rays are not allowed to pass unless they are at a greater radial distance than the maximum specified aperture, or at a smaller radial distance than the inner aperture. The "Circular Obscuration" is the complement of the "Circular Aperture". To set a central obscuration in a Cassegrain telescope for example, use the "Circular Obscuration" type and set the minimum radius to zero and the maximum radius to the desired obscuration size.

The "Rectangular Aperture", "Rectangular Obscuration", "Elliptical Aperture", and "Elliptical Obscuration" are very similar to the corresponding circular types, except the x- and y-half-widths are specified instead of the minimum and maximum radius.

The next aperture type is the "Spider". The spider is defined by the width of each arm, and by the number of arms. ZEMAX assumes the arms are all the same width, and that they are spaced in equal radial angles. The first arm starts at a radial position of zero degrees, which is along the local positive x-axis. More complex spiders, which contain arms of different widths at unequal angles can be constructed using several spiders on adjacent dummy surfaces. The coordinate break surfaces can be used to rotate the spider(s) to any desired angle.

All of the preceding apertures are modeled as a surface projected from the vertex tangent plane onto the optical surface. For example, the actual ray-surface intercept coordinates are used to determine vignetting. Slightly different results may be calculated for steep optical surfaces if the aperture is placed on a dummy surface in front of the optic, instead of being directly entered on the curved surface. This will only occur if the ray incidence angles are steep. Usually it is best to place the apertures directly on the optical surface, unless of course the dummy surface more accurately represents your situation.

All types of apertures may be decentered from the current optical axis by specifying either X- or Y-decenters, or both. The decenters are given in lens units. It is important to remember that decenters do not move the chief ray; the stop must be on the same axis as the object. To design an off axis telescope, for example, set the stop on axis and decenter the system.

## **Setting and removing solves**

Most data columns (such as radii and thickness) support one or more solves. To set a solve on a cell, double click with the left mouse button on that cell. Alternatively, single click with the right mouse button, or select the menu option from the Lens Data Editor menu. Solves are described in the Chapter "Solves".

## **Menu options**

The Lens Data Editor menu options are used for inserting and deleting surfaces, selecting surface types, and setting solves and variables.



## **Edit**

The edit menu offers these options:

**Surface Type:** This option allows the type of a surface to be changed.

**Insert Surface:** This inserts a new surface row in the spreadsheet at the current row. The Insert key is a shortcut for this menu option.

**Insert After:** This inserts a new surface row in the spreadsheet after the current row. The Ctrl-Insert key combination is a shortcut for this menu option.

**Delete Surface:** This deletes the current row in the spreadsheet. The Delete key is a shortcut for this menu option.

**Cut Surfaces:** Copies all the data for a single surface or a range of surfaces to the Windows Clipboard, then deletes the surfaces. The surface or range of surfaces must first be selected using either of the following techniques:

**Using the mouse:** Click on the first surface to be selected. Hold down the left mouse button, and drag the mouse cursor to cover the range of surfaces desired. The selected surfaces will be highlighted in inverse colors. To select only one surface, drag the mouse up or down from the surface desired until two surfaces are selected, then move the mouse back to the desired surface.

**Using the keyboard:** Move the cursor to any cell on the desired surface. Then hold down the shift key, while moving the cursor up or down until the desired range of surfaces is selected. The selected surfaces will be highlighted in inverse colors. To select only one surface, move the cursor up or down from the surface desired until two surfaces are selected, then move the cursor back to the desired surface.

**Copy Surfaces:** Copies all the data for a single surface or a range of surfaces to the Windows Clipboard. To select a single surface or a range of surfaces, see the discussion under "Cut Surfaces" above.

**Paste Surfaces:** Copies all the data for a single surface or a range of surfaces from the Windows Clipboard to the current cursor location in the Lens Data Editor. The surface data must first have been copied to the Windows Clipboard using either "Cut Surfaces" or "Copy Surfaces" described above.

## **Solves**

Solves and variables may be placed on many Lens Data Editor values:

**Radius:** Sets solves on the radius of curvature.

**Thickness:** Sets solves on the thickness.

**Glass:** Sets solves on the glass.

**Semi-Diameter:** Sets solves on the semi-diameter.

**Conic:** Sets solves on the conic constant.

**Parameter:** Sets solves on the parameter columns.

**Variable Toggle:** This will toggle the variable status of the currently selected cell. Ctrl-V is a shortcut for this operation.

## **Options**

**Show Comments:** If this menu option is checked, then the comment column is displayed. If it is unchecked, then the comment column is hidden. The hide/show status of the comment column is only used for the current session. To set this option automatically at the start of a ZEMAX session, see the Chapter "File Menu" under "Environment".

## **Help**

**Using the LDE:** Invokes the online help system for using the Lens Data Editor.

## Merit Function

---

The Merit Function Editor is used to define, modify, and review the system merit function. The system merit function is used for optimization, and is described in the Chapter "Optimization".

### Edit

**Insert Operand:** Inserts a new row in the spreadsheet at the current row. The Insert key will perform this same function.

**Insert After:** Inserts a new row in the spreadsheet after the current row. The Ctrl-Insert key combination will perform this same function.

**Delete Operand:** Deletes the row at the current cursor location. The Delete key will perform this same function.

**Cut Operands:** Copies all the data for a single operand or a range of operands to the Windows Clipboard, then deletes the operands. The operand or range of operands must first be selected using either of the following techniques:

Using the mouse: Click on the first operand to be selected. Hold down the left mouse button, and drag the mouse cursor to cover the range of operands desired. The selected operands will be highlighted in inverse colors. To select only one operand, drag the mouse up or down from the operand desired until two operands are selected, then move the mouse back to the desired operand.

Using the keyboard: Move the cursor to any cell on the desired operand row. Then hold down the shift key, while moving the cursor up or down until the desired range of operands is selected. The selected operands will be highlighted in inverse colors. To select only one operand, move the cursor up or down from the operand desired until two operands are selected, then move the cursor back to the desired operand.

**Copy Operands:** Copies all the data for a single operand or a range of operands to the Windows Clipboard. To select a single operand or a range of operands, see the discussion under "Cut Operands" above.

**Paste Operands:** Copies all the data for a single operand or a range of operands from the Windows Clipboard to the current cursor location in the Merit Function Editor. The operand data must first have been copied to the Windows Clipboard using either "Cut Operands" or "Copy Operands" described above.

### Tools

**Update:** This option recomputes the merit function. All operands are evaluated, and the new values are displayed.

**Default Merit Function:** Invokes the dialog box which used to define one of the default merit functions. See the Chapter "Optimization".

**Save:** Saves the current merit function in a \*.MF file. This step is only required if the merit function is to be subsequently loaded into another lens. ZEMAX automatically stores the merit function with the lens when the entire lens is saved.

**Load:** Loads a merit function previously stored in a \*.MF file or in a \*.ZMX file. Either file type may be selected; only the merit function portion of the file will be loaded into the spreadsheet. The current merit function is destroyed.

### Help

**Help on operands:** Invokes the online help system.

## Multi-Configuration

---

The Multi-Configuration editor is very similar to the Lens Data Editor. To edit the contents of a cell, simply move the cursor to that cell and type in the new data. To set a solve on a cell, double click with the left mouse button, or select the solve type menu option.

## **Edit**

Operand Type: This option allows the type of multi-configuration operand to be changed. For a complete description of multi-configuration operands, see the Chapter "Multi-Configurations".

Insert Operand: This inserts a new row in the spreadsheet at the current row. The new operand type is "OFF" which means the operand is ignored. The Insert key is a shortcut for this menu option.

Insert After: This inserts a new row in the spreadsheet after the current row. The new operand type is "OFF" which means the operand is ignored. The Ctrl-Insert key combination is a shortcut for this menu option.

Delete Operand: This deletes the current row in the spreadsheet. The Delete key is a shortcut for this menu option.

Insert Configuration: Selecting this item will insert a new column, which corresponds to a new configuration.

Delete Configuration: Deletes the configuration indicated by the current position of the cursor. This eliminates the column and all the data it contained.

## **Solves**

Solve Type: Selecting this option will invoke the solve dialog box for the cell which is currently indicated by the cursor.

Variable Toggle: This will toggle the variable status of the currently selected cell.

## **Tools**

Auto Thermal: This tool is used to perform the bulk of the tedious work in setting up a multi-configuration thermal analysis. A dialog box will appear which allows setting of the number of configurations, as well as the minimum and maximum temperatures.

The tool will then create 1 nominal configuration at the current temperature and pressure. Additional configurations are then defined to cover the temperature range specified. If 3 configurations are requested, then there will be 1 nominal configuration (configuration 1), and 3 configurations that span the specified temperature range in equal increments, for a total of 4 configurations. The air pressure will be assumed to be the same as the nominal pressure.

For each radius and thickness affected by temperature effects, appropriate operands will be inserted with TCE pickup solves.

This tool will erase any multi-configuration data already defined in the lens.

## **Help**

Help on operands: Invokes the online help system.

## **Tolerance Data**

---

The Tolerance Data editor is used to define, modify, and review the system tolerance values. See the Chapter "Tolerancing" for details.

## **Edit**

Insert Operand: Inserts a new row in the spreadsheet at the current row. The Insert key will perform this same function.

Insert After: Inserts a new row in the spreadsheet after the current row. The Ctrl-Insert key combination will perform this same function.

Delete Operand: Deletes the row at the current cursor location. The Delete key will perform this same function.

Cut Operands: Copies all the data for a single operand or a range of operands to the Windows Clipboard, then deletes the operands. The operand or range of operands must first be selected using either of the following techniques:

Using the mouse: Click on the first operand to be selected. Hold down the left mouse button, and drag the mouse cursor to cover the range of operands desired. The selected operands will be highlighted in inverse

colors. To select only one operand, drag the mouse up or down from the operand desired until two operands are selected, then move the mouse back to the desired operand.

Using the keyboard: Move the cursor to any cell on the desired operand row. Then hold down the shift key, while moving the cursor up or down until the desired range of operands is selected. The selected operands will be highlighted in inverse colors. To select only one operand, move the cursor up or down from the operand desired until two operands are selected, then move the cursor back to the desired operand.

Copy Operands: Copies all the data for a single operand or a range of operands to the Windows Clipboard. To select a single operand or a range of operands, see the discussion under "Cut Operands" above.

Paste Operands: Copies all the data for a single operand or a range of operands from the Windows Clipboard to the current cursor location in the Tolerance Data Editor. The operand data must first have been copied to the Windows Clipboard using either "Cut Operands" or "Copy Operands" described above.

## **Tools**

Default Tolerances: Invokes the default tolerances dialog box. See the Chapter "Tolerancing" for details.

Loosen 2X: Increases all tolerance ranges by a factor of 2. This is a quick way of loosening the tolerances if they are all too tight.

Tighten 2X: Decreases all tolerance ranges by a factor of 2. This is a quick way of tightening the tolerances if they are all too loose.

## **Help**

Help on operands: Invokes the online help system.

## **Extra data**

This feature is only available in the EE edition of ZEMAX.
--

The extra data editor is only used by special surfaces supported by ZEMAX-EE. The extra data editor is very similar to the lens data spreadsheet editor, except only the extra data values are displayed and edited. See the Chapter "Surface Types" for a discussion of special surfaces supported by ZEMAX-EE. Surfaces cannot be inserted or deleted on the extra data editor.

## **Solves**

Only one type of "solve" is allowed on the extra data values. Each extra data value may be defined as a variable or fixed value. The variable state for an extra data value is not a solve at all, but the menu option is called "Solves" to be consistent with the other spreadsheet editors and to permit the future addition of solves for extra data values.

## **Tools**

Two tools are available on the extra data editor: Solves and Import.

The Solves tool invokes a dialog box which allows selection of the status of the highlighted parameter. If the current cell is not used by the current surface type, then the solve dialog box will not appear. If the cell is used by the surface type, then the dialog box which appears can be used to change the status of that value between fixed, variable, and pick up solve.

The import tool is used to load extra data values for extra data surfaces from an ASCII file rather than by typing the numbers in directly. This menu option will invoke a dialog box which presents a list of ASCII data files which end in the extension .DAT. The dialog box also permits specification of which surface number should receive the data. Numerical data must be in the ASCII file exactly as it appears in the extra data spreadsheet. The format of the ASCII file is a single column of free-format numbers, and the file must end in the DAT extension. ZEMAX will look for the file in the default directory for ZEMAX specified on the environment screen.

## **Help**

Using the EDE: Invokes the online help system for using the Extra Data Editor.

## **Undo, Redo, and Recover**

---

There are three different states to the Undo capability in ZEMAX: None, Memory 1 Step, and Disk Multi Step. The Undo state is set on the Environment Editors box described in the Chapter "File Menu".

### **Undo: None**

If the Undo feature state is set to None, then no Undo capability is supported. Use this option on computers which do not have sufficient system memory or disk space to support the Undo feature.

### **Undo: Memory 1 Step**

ZEMAX stores a copy of the current lens in memory before and after every edit or optimization. If Undo is selected, then the current lens is swapped with the previous lens. If Redo is then selected, the lenses are swapped again, which results in the edit being restored.

Memory 1 Step Undo is useful for restoring a lens when an accidental edit is made, or for restoring a lens to the prior state after an optimization. However, only a single Undo step is supported. The advantage of this option is speed; the saving of the prior lens in memory is so fast as to be unnoticeable.

### **Undo: Disk Multi Step**

ZEMAX stores a copy of the current lens in a ZMX file on disk after every edit or optimization. These stored lens files are used to implement an infinite multi-step Undo feature, which allows the reversal of any change or series of changes made to the lens. Undo is useful for restoring a lens when an accidental edit is made, or for restoring a lens to the prior state after an optimization, or even several changes.

To reverse the changes made to a lens, simply select Undo from the Editors Menu. Any number of Undo's may be executed, all the way back to the first edit made after loading the lens file. The Redo function reverses the last Undo.

ZEMAX maintains a directory of Undo files, which by default is a subdirectory within the main ZEMAX directory called \UNDO. The Undo files are automatically deleted whenever the file is saved, a new file is opened, or ZEMAX is normally terminated. If ZEMAX abnormally terminates, the operating system fails, the computer power is disrupted, or for any other reason the lens data is lost, ZEMAX may be able to recover the lost data by restoring the last Undo file. When ZEMAX starts, a check is made to see if any of the Undo files exist. Since these files are deleted during a normal termination, the presence of Undo files indicates a previous abnormal termination. ZEMAX will issue a warning message with the option of restoring the last Undo file. If restored, this file should immediately be saved in a new file name, since the old file name is not stored with the lens.

The Undo feature does slow down the operation of the Editors slightly, since every edit is followed by a save operation. The save does not slow down ray tracing or optimization speed, only lens data editing.

This page intentionally left blank.

## Update All

---

This option updates all windows to reflect the most recent lens data. ZEMAX does not automatically change all data presented in graphic and text windows to reflect the latest lens data. The reason is that if ZEMAX constantly recalculated MTF, ray fan, spot diagram, and other data while new data is being typed in the lens data editor, then program response would be very slow. Instead, make all the required changes to the lens, and then select Update All to refresh and recompute all the data windows.

Individual graphic and text windows (not editors) can also be updated by double clicking anywhere within the window.

## Update

---

This option only updates data in the Lens Data and Extra Data editors. Update is used to recompute first-order properties, pupil positions, semi-diameters, index data, and solves. Only data presented in the Lens Data and Extra Data editors are affected. See "Update All" in this chapter for more information.

## General

---

This option invokes the General System Data dialog box, which is used to define commonly used data that pertains to the lens as whole system, rather than data associated with a single surface. See also the section on "Advanced".

### Lens Title

The lens title will appear on graphic and text output. The title is specified by simply typing the title in the required space. Additional text data may be placed on most graphic output, see "Configuring the environment" later in this chapter for details.

### Aperture Type

The system aperture defines the size of the beam through the system on axis. To set the system aperture, you need to define the system aperture type and the system aperture value. Use the cursor keys to select from the drop down list the desired system aperture type. The system aperture type can be any one of the following:

Entrance Pupil Diameter: The diameter of the pupil in lens units as seen from object space

Image Space F/#: The infinite conjugate paraxial F/# in image space

Object Space Numerical Aperture: The numerical aperture ( $n \sin\theta_m$ ) of the marginal ray in object space

Float By Stop Size: Defined by the semi-diameter of the stop surface

Paraxial Working F/#: The defined conjugate paraxial F/# in image space.

Object Cone Angle: The half angle in degrees of the marginal ray in object space, which may exceed 90 degrees.

These terms are further defined in the Chapter "Conventions and Definitions". If you select "Object Space N.A." or "Object cone angle" for a system aperture type, the object thickness must be less than infinity. Only one type of system aperture may be defined; for example, once the Entrance Pupil Diameter is specified, all the other aperture definitions are determined by the lens data prescription.

### Aperture Value

The system aperture value meaning depends upon the system aperture type selected. For example, if "Entrance Pupil Diameter" is selected as the system aperture type, the system aperture value is the entrance pupil diameter in lens units. ZEMAX uses the system aperture type and system aperture value together to determine such fundamental quantities as the entrance pupil size, and clear apertures for all components.

Selecting "Float by Stop Size" for a system aperture type is the only exception to this rule. The semi-diameter of the stop surface (set on the lens data editor) is used to define the system aperture if "Float by Stop Size" is selected for a system aperture type.

### **Lens Units**

There are four choices for lens units: millimeters, centimeters, inches, or meters. These units are used for prescription data, such as radii, thickness, and entrance pupil diameters. Most of the diagnostics (rayfans, spot diagrams) are always in microns, as are the wavelengths.

### **Glass Catalogs**

This control group has an editable field line which lists the names of the currently used glass catalogs (with no extensions). The line by default is "schott misc" which means the lens may use glasses from these two catalogs. Select either the radio buttons or type the name of the catalog to use if you require a different catalog. To use a glass catalog which is not listed by a radio button, simply type the name of the catalog in the edit field. Multiple catalogs may be specified by separating the names with a space. For more information on changing glass catalogs, see the Chapter "Using Glass Catalogs".

### **Notes**

The notes section allows entry of a few lines of text which is stored with the lens file.

### **Advanced**

---

This option invokes the Advanced System Data dialog box, which is used to define less commonly used data that pertains to the lens as whole system, rather than data associated with a single surface. See also the section on "General".

### **Ray Aiming**

The ray aiming selection box has three states: None, Paraxial Reference, and Real Reference. If the ray aiming state is "None", ZEMAX will use the paraxial entrance pupil size and location determined by the aperture settings and calculated at the primary wavelength on axis for launching rays from the object surface. This means ZEMAX will ignore entrance pupil aberration. For slow systems with modest field angles, this is perfectly acceptable. However, certain systems, such as those with low F/#'s or large field angles, may have significant entrance pupil aberration. The two primary effects of pupil aberration are the shift in location of the pupil with field angle, and the anamorphic scaling of the edges of the pupil.

ZEMAX can be instructed to account for the aberration if ray aiming is selected. With ray aiming on, every ray trace is performed iteratively, with the program adjusting the ray aiming so that the ray crosses the correct location on the stop surface.

The correct location on the stop surface is determined by first computing the stop surface radius. The correct stop surface coordinates are then computed using a linear scaling of the pupil coordinates. For example, the marginal ray has a normalized pupil coordinate of  $P_y = 1.0$ . The correct coordinate on the stop surface is then the stop surface radius times  $P_y$ .

The stop surface radius can be computed using either paraxial or real rays. If "Real Reference" is selected then the marginal ray at the primary wavelength is traced from the center of the object through the system to the stop surface. The ray height at the stop surface is then the stop radius. If "Paraxial Reference" is selected, then a paraxial ray is used instead. When the "Real Reference" is used, all real rays are aimed to have the correct location on the stop surface based up on the real stop radius, however, paraxial rays are always referenced to the paraxial stop radius.

When using ray aiming, the stop, rather than the entrance pupil, is the surface that is uniformly illuminated. This can lead to unexpected results. For example, when using Object Space NA as a system aperture type, ZEMAX traces rays with the correct NA to locate and size the paraxial entrance pupil. If ray aiming is then set to "Paraxial Reference", real ray tracing will be referenced to the paraxial stop size. This may yield a numerical aperture which is different from that specified by the system aperture value. This is because the ray angle was adjusted (aimed) to account for any pupil aberration. The solution in this case is to instead use "Real Reference".

In general, "Paraxial Reference" is the preferred setting.



Although ray aiming is more accurate than paraxial pupil aiming, most ray traces will take from two to eight times as long to perform (see the "Tolerance" option described later). Therefore, ray aiming should only be used when required. To determine the amount of entrance pupil aberration in your system, select ray aiming off, and then look at the pupil aberration plot (see the Chapter "Analysis Menu" for a discussion of this feature). Pupil aberration of less than a few percent is generally insignificant. If your system has significant pupil aberration, select ray aiming on and repeat the calculation. The aberration will decrease to zero, or very nearly so.

### **Use Ray Aiming Cache**

If checked, ZEMAX caches ray aiming coordinates so that new ray traces take advantage of previous iterations of the ray tracing algorithm. Using the cache can speed up ray tracing dramatically. However, use of the cache does require that the chief ray can be traced accurately. For some systems, the chief ray cannot be traced, and for these systems, the cache should be turned off.

### **Pupil Shift: X, Y, and Z**

For most systems, simply selecting ray aiming on will eliminate the effects of pupil aberration, as least as far as getting the rays to correctly trace through the system. It does not, of course, actually eliminate the aberration, it merely accounts for it. For some very wide angle or highly tilted or decentered systems, the ray aiming feature will fail if unassisted. The problem is that the paraxial entrance pupil is used as a first guess to trace the ray. If the pupil aberration is severe, it is possible that even this first guess cannot be traced, which prevents the algorithm from taking a second, more refined guess.

The solution is to provide a rough guess as to how much the pupil has shifted with respect to the paraxial pupil. This is called "Pupil shift", and there are three components; x, y, and z. The default value of zero for all three may be modified to assist the algorithm in finding a successful first guess for the ray aiming. Positive values for the z shift indicate that the real pupil is behind (to the right of in the conventional optical coordinate system) the paraxial pupil, negative values indicate the pupil is shifted forward. Most wide angle systems have forward-shifted pupils.

The z pupil shift value provided is scaled linearly with the field angle of the ray being traced, so the pupil shift refers to the offset of the pupil at full field. The x and y pupil shift values account for the change in location of pupils when the object plane is tilted or when the stop is decentered. If "Scale pupil shift factors by field" is selected, the x and y pupil shift values are also scaled with field, otherwise, the shift value is used for all fields without any scaling. All shifts are in lens units.

It is important to understand that the exact values of the pupil shift is unimportant. Once the first guess ray can be traced, the ray aiming algorithm will robustly find the exact pupil location. The pupil shift values are just to get the ray aiming started. Generally, guessing at the pupil shift values is an acceptable way of determining a suitable value.

### **Apodization Type**

By default, the pupil is always illuminated uniformly. However, there are times when the pupil should have a non-uniform illumination. For this purpose, ZEMAX supports pupil apodization, which is a variation of amplitude over the pupil. Three types of pupil apodization are supported: uniform, gaussian, and tangential. Uniform means rays are distributed uniformly over the entrance pupil, simulating uniform illumination.

Gaussian apodization imparts an amplitude variation over the pupil that is gaussian in form. The apodization factor refers to the rate of decrease of the beam amplitude as a function of radial pupil coordinate. The beam amplitude is normalized to unity at the center of the pupil. The amplitude at other points in the entrance pupil is given by

$$A(\rho) = e^{-G\rho^2},$$

where G is the apodization factor and  $\rho$  is the normalized pupil coordinate. If the apodization factor is zero, then the pupil illumination is uniform. If the apodization factor is 1.0, then the beam amplitude has fallen to the 1 over e point at the edge of the entrance pupil (which means the intensity has fallen to the 1 over e squared point, about 13% of the peak). The apodization factor can be any number greater than or equal to 0.0. Values larger than about 4.0 are not recommended. This is because most computations will sample too few rays to produce meaningful results if the beam amplitude falls too quickly off axis.

Tangential apodization is appropriate to simulate the intensity fall off characteristic of a point source illuminating a flat plane (such as the entrance pupil, which is always a plane). For a point source, the intensity on a plane a distance Z from the source is given by

$$I(r) = \frac{Z^2}{Z^2 + r^2},$$

where r is the distance from the vertex on the plane, and the intensity has been normalized to be unity on axis. Substituting normalized pupil coordinates for r and taking the square root yields the amplitude apodization

$$A(\rho) = \frac{1}{\sqrt{1 + \rho^2 \tan^2 \theta}},$$

where  $\tan\theta$  is the tangent of the angle between the z axis at the top of the entrance pupil. For the tangential apodization, the apodization factor is  $\tan\theta$ . For the special case of the apodization factor being zero, ZEMAX uses the entrance pupil position and size to automatically compute  $\tan\theta$  when computing apodization.

The following features take into account apodization factors: diffraction and geometric MTF, PSF, diffraction and geometric encircled energy, Zernike analysis, all layout plots, and all spot diagrams. The default merit functions based upon rectangular arrays also weight rays according to the apodization factor, but the gaussian quadrature algorithms do not.

### **Apodization Factor**

The apodization factor determines how fast the amplitude decays in the pupil. See the previous section on the Apodization Type for details.

### **Reference OPD**

The Optical Path Difference, or OPD, is of value in optical design computations because the OPD represents the phase error of the wavefront forming an image. Any deviations from zero OPD contribute to a degradation of the diffraction image formed by the optical system.

Because the exit pupil is the image of the stop in image space, the exit pupil represents the only location in image space where the beam has a clearly defined edge. The illumination at the exit pupil is generally smoothly varying in amplitude and phase, and there is a clearly defined boundary between regions of zero amplitude and non zero amplitude. In other words, it is reasonable to assume that there are no diffraction effects apparent in the wavefront when viewed in the exit pupil. This is asymptotically true if all apertures in the optical system are large compared to the stop limited beam size incident upon each aperture. Even if the exit pupil is virtual, which is often the case, the exit pupil still defines the only location in image space where the beam is diffraction free. For more information on diffraction image formation and the importance of the exit pupil, see any of the good books referenced in the first chapter.

As the wavefront propagates from the exit pupil towards the image plane, the beam profile becomes complex in amplitude and phase, and the wavefront extends over all space due to the effects of diffraction. For this reason, the phase error as measured in the exit pupil is uniquely and critically important to the description of the wavefront and image quality.

ZEMAX by default uses the exit pupil as a reference for OPD computations. Therefore, when the OPD is computed for a given ray, the ray is traced through the optical system, all the way to the image plane, and then is traced backward to the "reference sphere" which lies in the exit pupil. The OPD as measured back on this surface is the physically significant phase error important to diffraction computations, such as MTF, PSF, and encircled energy. The additional path length due to the tracing of the ray backwards to the exit pupil, subtracted from the radius of the reference sphere, yields a slight adjustment of the OPD called the "correction term". This computation is correct and is the desired method for all cases of practical interest.

However, ZEMAX does permit selection of two other reference methods.

The reference to "Infinity" makes the assumption that the exit pupil is very far away (even if it may not be) and that the OPD correction term is given strictly by the angular error in the ray. There is only one possible reason to use this setting: in the unlikely event that ZEMAX is unable to correctly compute the exit pupil location. This can occur with some very unusual optics which do not form images (real or virtual) of the stop surface. ZEMAX has

special code to handle all known cases where this might occur, and therefore, this setting should not be used unless Focus Software technical support specifically recommends it. There are currently no known cases where this would be the recommended setting.

The reference to "Absolute" means that ZEMAX does not add any correction term at all to the OPD computation, but simply adds up the total optical path length of the ray and subtracts it from the chief ray. This method is never physically significant, and the only intended purpose is for debugging and testing of the OPD algorithms by Focus Software.

In summary, always use the "exit pupil" reference, unless specifically instructed to change the setting by a Focus Software Engineer. Erroneous data may easily result if "exit pupil" is not selected.

### **Paraxial Rays**

Paraxial ray properties are generally not defined for non rotationally symmetric systems. For this reason, ZEMAX by default ignores all tilts and decenters due to coordinate breaks when tracing paraxial rays. By ignoring tilts and decenters, ZEMAX can compute the paraxial properties of an equivalent centered system, which is generally the correct approach even for systems without symmetry.

The default setting "Ignore Coordinate Breaks" is therefore highly recommended. Choosing a different setting may result in the failure of ZEMAX to accurately compute all paraxial data, ray aiming, and OPD computations.

There is only one known case where "Consider Coordinate Breaks" may be required. For ray tracing through steeply tilted gratings, coordinate breaks may be required even for paraxial rays, otherwise, the rays may not be able to satisfy the grating equation. This is because diffraction gratings bend rays by an amount that depends critically on the incident angle.

### **Fast Asphere Trace**

When tracing rays through certain types of aspheric surfaces, iteration is required if no closed form solution exists to the ray-surface intercept equations. If this box is checked (the default condition) ZEMAX makes an initial guess for the ray-surface intercept solution in an attempt to accelerate convergence of the iteration. However, some wildly curved aspheric surfaces will not converge if this "fast guess" is used. For systems which use these surfaces, this box may need to be unchecked, in which case ZEMAX will use a slower but more robust algorithm for finding the solution. Whether the box is checked or unchecked, ZEMAX will find an accurate ray intercept solution, or an error message will be issued.

### **Check GRIN Apertures**

If checked, this setting instructs ZEMAX to check all gradient index ray traces for surface aperture vignetting. Each GRIN trace within the media is checked to see if the ray ever passes outside the front surface aperture boundary, and if it does, then the ray is vignetted. If this setting is not checked, then the ray may travel outside the boundary defined on the front surface, as long as the ray passes the aperture at the surface.

### **Use T, P**

This checkbox should be checked if the temperature and pressure values are different from 20 and 1, respectively. If this box is unchecked, then all temperature and pressure effects are ignored. This speeds up the computation of index data, and so the box should be left unchecked if the nominal temperature and pressure are to be used. See the Chapter "Thermal Analysis". This feature is only available in ZEMAX-EE.

### **Temperature**

The ambient temperature in degrees Celsius. See the Chapter "Thermal Analysis". This feature is only available in ZEMAX-EE.

### **Pressure**

The air pressure in atmospheres. A value of 0.0 implies vacuum, 1.0 implies sea level. See the Chapter "Thermal Analysis". This feature is only available in ZEMAX-EE.

### **Semi Diameter Margin in %**

Normally, the semi-diameter of every surface in "automatic" mode is computed by ZEMAX to be the radial aperture required to pass all rays without clipping. For systems with closely spaced elements in or near edge

contact, this default yields surface clear apertures that provide no clearance left for finishing or mounting. Often, optical surfaces are only well finished within some fraction of the full radial aperture, typically between 90% and 98%, depending upon the part size.

The semi diameter margin control allows specification of an additional amount of radial aperture as a percentage. The default value of zero leaves no margin, while a margin of 5% would add 5% to the semi-diameters of all surfaces under "automatic" control.

This control very much simplifies the design of system with closely spaced elements with steep surfaces and edge contact points. The maximum allowed margin is 50%.

### **Fast Semi-Diameters**

ZEMAX computes "automatic" semi-diameters to estimate the clear aperture required on each surface to pass all rays at all field points and wavelengths. For axial systems, this computation can be done accurately by tracing just two rays for each field and wavelength. The two rays are the top and bottom marginal rays.

For non-axial systems, there is no general way of accurately computing the semi-diameters other than by tracing a large number of rays along the perimeter of the vignetted pupil. Although this is accurate, it can be slow, since ZEMAX needs to update the semi-diameters frequently, especially during optimization. There is a tradeoff between speed and accuracy.

By default, for non-axial systems ZEMAX will trace just two rays in the true tangential plane of the vignetted pupil for each field and wavelength, then use the radial coordinates of each ray at each surface to estimate the semi-diameter required. For some systems, the resulting estimate is not accurate enough. This typically includes systems which have tight edge and clear aperture constraints, or those with component decenters and only a few field points.

If the "Fast Semi-Diameters" option is checked "off", then for these non-axial systems ZEMAX will iteratively trace as many rays as required to determine the semi-diameter to an accuracy of about 0.01% (5 significant figures). The speed of optimization in particular may be slowed substantially by turning Fast Semi-Diameters off, but for systems with very complex merit functions the overhead may be relatively small.

## **Fields**

---

The fields dialog box allows specification of field points. The fields may be specified as angles, object heights (for systems with finite conjugates), or image heights. The buttons can be used to activate or deactivate field positions, as well as to sort the data. For information about field conventions, see the Chapter "Conventions and Definitions".

ZEMAX also provides data fields for defining vignetting factors. The four vignetting factors are VDX, VDY, VCX, and VCY. These vignetting factors should be left at zero if there is no vignetting in the system. The factors are described in the Chapter "Conventions and Definitions" under "Vignetting Factors".

There is also a button labeled "Set Vig" on the fields dialog box. Clicking on this button will recompute the vignetting factors for each field based upon the current lens data. The set vignetting algorithm estimates the vignetting decenter and compression factors so that the four marginal rays in the top, bottom, left, and right edges of the pupil pass within the user-defined semi-diameters of each surface. Only the primary wavelength is used. To clear the vignetting factors back to the default values of zero, click on "Clr Vig".

The algorithm starts by launching a grid of rays through the pupil. At each surface with a user defined semi-diameter, the ray is tested to see if it passes within the specified aperture. All rays which pass all surfaces are then used to compute the unvignetted pupil centroid. Note that only user-defined (not automatic) semi-diameters are used, and surface apertures (such as rectangular or circular apertures) are ignored. The exact edge of the unvignetted pupil is then computed using an iterative method accurate to about 0.001%.

The algorithm may not work in all cases. For systems where the set vignetting algorithm fails, the vignetting factors will need to be adjusted manually. The accuracy of the set vignetting algorithm can be tested by tracing a few marginal rays.

## **Wavelengths**

---

The wavelengths dialog box is used to sets wavelengths, weights, and the primary wavelength. The buttons can be used to activate and deactivate wavelengths, and to sort the data. A list of commonly used wavelengths is also included. To use the entries on the list, select the desired wavelength set, and click on the "Select" button.

For more information, see the Chapter "Conventions and Definitions".

## **Next Configuration**

---

This menu option provides a quick way to change all graphics to reflect the next configuration (or zoom position). All spreadsheet, text, and graphics data will be updated.

## **Last Configuration**

---

This menu option provides a quick way to change all graphics to reflect the last configuration (or zoom position). All spreadsheet, text, and graphics data will be updated.

This page intentionally left blank.

## Introduction

---

This chapter provides detailed descriptions of each of the analysis features ZEMAX supports. Analysis in this context means any graphical or text data computed from data defining the lens. This includes aberrations, MTF, spot diagrams, and many other computations. Program features which modify the lens data or which manipulate other data (such as glass catalog data) are described in the chapter "Tools Menu".

Selecting a menu option will immediately perform the requested calculation. Once the graph or text window is displayed, you may select the Settings menu option to modify the default settings for that window. Once you have made the appropriate changes, click on "OK" and the program will recalculate and redisplay the data presented in the window. If you prefer to change the settings before the graphic or text data is displayed, use the "Show Options First" checkbox on the File:Environment:Graphics dialog box.

For a description of the OK, Cancel, Save, Load, Reset, and Help buttons present on most of the "Settings" dialog windows, see the chapter "User Interface".

Each analysis window has an "Update" menu item. The update function forces ZEMAX to recompute and redisplay the data presented in the window. This is useful if the lens data has changed and the graph now displays obsolete data. Double clicking within the window has the same effect as selecting Update. Clicking with the right mouse button is equivalent to clicking on "Settings".

For more information, see the chapter "User Interface".

## Layout

---

### 2D Layout

#### *Purpose:*

Layout diagram. This is a simple YZ cross section through the lens.

#### *Settings:*

Item	Description
First Surface	The first surface to be drawn.
Last Surface	The last surface to be drawn.
Wavelength	Either any one or all wavelengths may be shown.
Field	Either any one or all field positions may be shown.
Number of Rays	The number of rays specifies the number of tangential rays to be drawn for each defined field. The rays will be evenly distributed along the fan of the pupil, unless apodization has been specified. This parameter may be set to zero.
Scale Factor	If the scale factor is set to zero, then "Fill Frame" will be selected, which will scale the range of surfaces drawn to fill the graphic page. If a numeric value is entered, then the plot will be drawn in "real" scale, times the scale factor. For example, a scale factor of 1.0 will plot the lens actual size on the hardcopy directed to the printer (not the screen). A factor of 0.5 will plot the lens at half scale.
Upper Pupil Limit	The maximum pupil coordinate to draw rays to.
Lower Pupil Limit	The minimum pupil coordinate to draw rays to.
Marginal and Chief Only	Draws only the marginal and chief rays, overriding the other ray settings.
Square Edges	Draws flat faces and edges if selected, otherwise uses the semi-diameter value to draw lens edges.
DXF File	In this text cell enter the name of the DXF format file to use. This option is only used if the button "Export As DXF File" is subsequently pressed. The file will be stored in the default directory for output.

Item	Description
Export As DXF File	If pressed, this button will cause the generation of a DXF formatted file which will contain the same data that is displayed in the graphic window. The file name is given by the "DXF File" option. The DXF file generated is a 2D model of the lens system suitable for importing into CAD programs that can read DXF files. See the discussion section for details.
Color Rays By	Select "Fields" to use color to distinguish between each field position, or "Waves" to distinguish between each wavelength.
Suppress Frame	Suppresses drawing of the frame on the bottom of the screen, which leaves more room for the layout plot itself. No scale bar, address block, or other data will be displayed.

*Discussion:*

This feature is not available if you use coordinate breaks, spider obscurations, obscuration decenters, X-angles, holograms, or other attributes which spoil the rotational symmetry of the lens. Use the 3D layout instead.

The "Export As DXF File" button will generate a 2D DXF file and store it in the file name provided in the "DXF File" data field. The DXF file will consist of arcs and lines. The arcs are used to show the shape of curved lens faces. If only spherical (or plane) lenses are used, then the arc is a perfect representation of the lens. However, the arcs are only approximations to non-spherical surfaces. If the surface sag is described by an asphere, then the arc is correct at only three points: the vertex, and the top and bottom points. ZEMAX uses the exact surface sag at these three points to fit the arc.

If rays miss a surface, then the rays will not be drawn to the surface where the error occurred. If the ray is total internal reflected, then the ray will be drawn up to but not past the surface where the error occurred. Ray failures can be evaluated in detail by using the Ray Trace calculation described later in this chapter.

The first and last surface numbers cannot be saved as the other options can be. This is because the surfaces may not be valid for lens files which are subsequently opened and then displayed.

### **3D Layout**

*Purpose:*

Draws 3D layout plots of the lens system. The algorithm draws a wireframe style representation of the lens.

*Settings:*

Item	Description
First Surface	The first surface to be drawn.
Last Surface	The last surface to be drawn.
Wavelength	Either any one or all wavelengths may be shown.
Field	Either any one or all field positions may be shown.
Number of Rays	The number of rays specifies the number of sagittal and tangential rays to be drawn for each defined field. The rays will be evenly distributed along the fan of the pupil, unless apodization has been specified. This parameter may be set to zero.
Ray Pattern	Choose XY Fan, X Fan, Y Fan, or Ring to indicate what the pattern of rays to be traced should be.
Scale Factor	If the scale factor is set to zero, then "Fill Frame" will be selected, which will scale the range of surfaces drawn to fill the graphic page. If a numeric value is entered, then the plot will be drawn in "real" scale, times the scale factor. For example, a scale factor of 1.0 will plot the lens actual size on the hardcopy directed to the printer (not the screen). A factor of 0.5 will plot the lens at half scale.
Hide Lens Faces	If checked, this option will suppress drawing of the lens faces, and only the lens edges will be drawn. This is useful because some complicated systems look cluttered with the faces drawn.
Hide Lens Edges	If checked, this option will suppress drawing of the outer aperture of the lens. This is useful for giving the 3D layout a 2D "cross section" appearance.



Item	Description
Hide X Bars	If checked, this option will suppress drawing of the X portions of the lens faces. This option is useful when "Hide Lens Edges" is checked and "Hide Lens Faces" is not checked.
Rotation About X	The angle in degrees by which the lens appears to be rotated about the X axis.
Rotation About Y	The angle in degrees by which the lens appears to be rotated about the Y axis.
Rotation About Z	The angle in degrees by which the lens appears to be rotated about the Z axis.
Color Rays By	Select "Fields" to use color to distinguish between each field position, or "Waves" to distinguish between each wavelength.
Suppress Frame	Suppresses drawing of the frame on the bottom of the screen, which leaves more room for the layout plot itself. No scale bar, address block, or other data will be displayed.

*Discussion:*

Pressing the left, right, up, down, Page Up, or Page Down keys will rotate the displayed image for a different perspective.

If rays miss a surface, then the rays will not be drawn to the surface where the error occurred. If the ray is total internal reflected, then the ray will be drawn up to but not past the surface where the error occurred. Ray failures can be evaluated in detail by using the Ray Trace calculation described later in this chapter.

The first and last surface numbers cannot be saved as the other options can be. This is because the surfaces may not be valid for lens files which are subsequently opened and then displayed.

**Solid Model**

*Purpose:*

Draws a hidden-line representation of the lens.

*Settings:*

The options are similar to those available for the 3D layout feature. The "Hide Lens Edges" and "Hide X Bars" checkboxes are not available, and several new controls are added as described below.

Item	Description
DXF File	In this text cell enter the name of the DXF format file to use. This option is only used if the button "Export As DXF File" is subsequently pressed. The file will be stored in the default directory for output.
Export As DXF File	If pressed, this button will cause the generation of a DXF formatted file which will contain the same data that is displayed in the graphic window. The file name is given by the "DXF File" option. The DXF file generated is a 3D faceted model of the lens system suitable for importing into CAD programs that can read DXF files.
Draw Section	Select "Full" to draw each lens element completely. The 3/4, 1/2, and 1/4 options draw just that much of the element, yielding a cut away perspective of the lens interior.
Radial Segs	The number of radial segments used to approximate the lens shapes. Larger numbers require more processing time.
Angular Segs	The number of angular segments used to approximate the lens shapes. Larger numbers require more processing time.

*Discussion:*

The solid model algorithm describes the lens as a collection of polygon facets. The lines and facets which are hidden from view are removed, which gives the lens a solid appearance. This algorithm is slower than the other layout plots, but produces the best looking results. The number of facets used to display the lens elements can be modified using the radial and angular segment options.

The "Export As DXF File" button will generate a 3D DXF file and store it in the file name provided in the "DXF File" data field. The DXF file will consist of small faceted surfaces in a fully 3D orientation. The facets are used to show the shape of curved lens faces. However, the facets are nearly flat segments which only approximate the surface contour. The corners of each facet always lie exactly on the real optical surface, but arbitrary points

within the facet do not follow the contour of the surface. ZEMAX uses the exact surface sag at the corners of each facet to define the facet shape.

Pressing the left, right, up, down, Page Up, or Page Down keys will rotate the displayed image for a different perspective.

If rays miss a surface, then the rays will not be drawn to the surface where the error occurred. If the ray is total internal reflected, then the ray will be drawn up to but not past the surface where the error occurred. Ray failures can be evaluated in detail by using the Ray Trace calculation described later in this chapter.

The first and last surface numbers cannot be saved as the other options can be. This is because the surfaces may not be valid for lens files which are subsequently opened and then displayed.

### **Wireframe**

#### *Purpose:*

Draws a wireframe representation of the lens.

#### *Settings:*

The settings are identical to those available for the Solid Model feature, including support of DXF export.

#### *Discussion:*

The wireframe model is identical to the solid model, except hidden lines are not removed. This representation may cause the screen to become cluttered with lines. The "Hide Lens Faces" option can be used to clean up the display. The advantage to this display method is speed; it is faster than the solid model.

If rays miss a surface, then the rays will not be drawn to the surface where the error occurred. If the ray is total internal reflected, then the ray will be drawn up to but not past the surface where the error occurred. Ray failures can be evaluated in detail by using the Ray Trace calculation described later in this chapter.

The first and last surface numbers cannot be saved as the other options can be. This is because the surfaces may not be valid for lens files which are subsequently opened and then displayed.

### **Shaded Model**

#### *Purpose:*

Draws a shaded solid model representation of the lens using OpenGL graphics.

#### *Settings:*

The options are almost identical to those available for the Solid Model feature, except there are additional controls for setting the lighting level and background color.

#### *Discussion:*

The first and last surface numbers cannot be saved as the other options can be. This is because the surfaces may not be valid for lens files which are subsequently opened and then displayed.

### **Element Drawing**

#### *Purpose:*

This feature creates a mechanical drawing of surface, singlet, or doublet elements suitable for use in optical shop fabrication.

#### *Settings:*

Item	Description
Surface	The first surface of the element to be drawn.
Show As	Select either "Surface", "Singlet" or "Doublet".

Item	Description
Note File Name	The name of the ASCII file which contains the notes to be appended to the notes section of the element drawing. Notes should always start at number 2, since number 1 is reserved for the units specification.
Edit Note File	Clicking on this button will invoke the Windows NOTEPAD.EXE editor, which can then be used to modify the selected note file.
Rad n Tol	The radius (1,2,or 3) tolerance box value. The default value is 1%.
Pow/Irr n	The power/irregularity (1,2, or 3) tolerance box value. The default value is 2.0/0.5.
Clear Ap n	The clear aperture of the lens on surface n. The default value is twice the semi-diameter value.
Thick n Tol	The center thickness tolerance of surface n. The default value is 1%.
Scale Factor	If the scale factor is set to zero, then "Fill Frame" will be selected, which will scale the element to fill the right half of the element drawing. If a numeric value is entered, then the plot will be drawn in "real" scale, times the scale factor. For example, a scale factor of 1.0 will plot the element at actual size on the hardcopy directed to the printer (not the screen). A factor of 0.5 will plot the element at half scale.
Drawing Name	All of these fields are for user defined text. Any text may be entered. No default value.
Approved	
Revision	
Drawn By	
Project	
Note Font Size	Choose Standard, Medium, Small, or Fine. These are in order of decreasing font size. The Note Font Size setting only affects the size of the note file that is annotated on the drawing. Smaller fonts permit larger note files to be displayed.

### *Discussion:*

The most important feature of the element drawing capability is the ability to load different note files and place them on the drawing. The default note file "DEFAULT.NOT" is a generic set of notes which will rarely be useful as is. However, the user can modify the note files (they are ASCII files which any word processor or text editor can modify) and store them under different names. For example, you may want to have a .NOT file for each type of optic you design, and then load the most appropriate note file when the element drawing is generated.

The note files should always start at note number 2. Note number 1 is reserved by ZEMAX for the line "1) All dimensions in millimeters" or whatever the current lens units are. The line breaks and spacings in the note file will be replicated exactly on the element drawing.

It is possible to set different values for the radius tolerance and the power tolerance (in fringes). These specifications are related, but ZEMAX does not force you to make them consistent. For a Newton's rings type optical test against a test plate, the error in radius is related to the number of fringes by

$$\#fringes = \frac{\Delta R}{\lambda} \frac{\rho^2}{R^2},$$

where  $\Delta R$  is the radius error,  $\lambda$  is the test wavelength,  $\rho$  is the radial aperture, and  $R$  is the radius of curvature. This formula is an approximation for shallow curvatures. For more information, see Malacara, Optical Shop Testing, J. Wiley & Sons, Inc.

## **Fans**

### **Ray Aberration**

#### *Purpose:*

Shows ray aberrations as a function of pupil coordinate.

### Settings:

Item	Description
Maximum Scale	Sets the maximum vertical scale for the plots. The maximum scale is in microns for ray fans, waves for OPD plots, or percent for entrance pupil aberration plots. This overrides the automatic selection of scale for the plots. Enter zero for automatic scaling.
Number of Rays	This is the number of rays traced on each side of the origin of the plot.
Wavelength	The wavelength number for which the calculation should be performed.
Field	The field number for which the calculation should be performed.
Tan Fan	Selects which aberration component to plot for the tangential fan. Since tangential fans are functions of the y pupil coordinate, the default is to plot the y component of the aberration.
Sag Fan	Selects which aberration component to plot for the sagittal fan. Since sagittal fans are functions of the x pupil coordinate, the default is to plot the x component of the aberration.
Use Dashes	Selects either color (for color screens and plotters) or dashes (for monochrome screens and plotters).
Check Apertures	Specifies whether or not to check if rays pass all surface apertures. If this is selected, rays which do not pass surface apertures will not be drawn. Surface apertures are different from semi-diameters, see the chapter "Conventions and Definitions".
Vignetted Pupil	If checked, the pupil axis will be scaled to the unvignetted pupil, in which case the data will reflect the vignetting in the system. If unchecked, the pupil axis will be scaled to fit the vignetted pupil.

### Discussion:

The tangential fans show either the x or the y component of the transverse ray aberration as a function of the y pupil coordinate of the ray. The default option is to plot the y component of the aberration. However, since transverse ray aberrations are vectors, this is an incomplete description of the aberration. When ZEMAX plots the y component, the plot is labeled EY, when plotting the x component of the aberration, the plot is labeled EX.

The vertical axis scale is given at the bottom of the graph. The data being plotted is the difference between the ray intercept coordinate and the chief ray intercept coordinate. The tangential fan is the plot of the difference between the ray x or y coordinate and the chief ray x or y coordinate at the primary wavelength, as a function of the y pupil coordinate. The sagittal plot is the difference between the ray x or y coordinate and the chief ray x or y coordinate as a function of the x pupil coordinate. The horizontal scale for each graph is the normalized entrance pupil coordinate, either PX or PY.

If "All" wavelengths are shown, then the plot is referenced to the primary wavelength chief ray. If monochromatic, then the chief ray for the selected wavelength is used as a reference. For this reason, the data for non-primary wavelengths will in general change when switching between monochromatic and polychromatic displays.

Because ray aberrations are vectors, with both an x and a y component, the ray aberration fan is an incomplete description of the aberrations, especially when the image plane is rotated or the system is otherwise non-rotationally symmetric. Also, the fans only indicate aberrations along two "slices" through the pupil, rather than over the entire entrance pupil. The primary purpose of the ray fan plot is to determine what aberrations are present in the system; it is not a complete description of the system performance, especially for systems without rotationally symmetry.

## **Optical Path**

### *Purpose:*

Shows optical path difference as a function of pupil coordinate.

### *Settings:*

The options are identical to those for ray aberration fans, except the only option for "Tan Fan" and "Sag Fan" is OPD, since OPD is a scalar quantity.

*Discussion:*

The vertical axis scale is given at the bottom of the graph. The data being plotted is the optical path difference, or OPD, which is the difference between the optical path length of the ray and the optical path length of the chief ray. Usually, the calculation is referenced back to the difference between the ray path lengths at the system exit pupil. The horizontal scale for each graph is the normalized entrance pupil coordinate.

If "All" wavelengths are shown, then the plot is referenced to the primary wavelength based reference sphere and chief ray. If monochromatic, then the reference sphere and chief ray for the selected wavelength is used as a reference. For this reason, the data for non-primary wavelengths will in general change when switching between monochromatic and polychromatic displays.

**Pupil Aberration**

*Purpose:*

Shows entrance pupil distortion as a function of pupil coordinate.

*Settings:*

The options are identical to those for ray aberration fans., except the only option for "Tan Fan" and "Sag Fan" is pupil aberration, since pupil aberration is a scalar quantity.

*Discussion:*

Entrance pupil aberration is defined as the difference between the real ray intercept on the stop surface and the on axis primary wavelength paraxial ray intercept as a percentage of the paraxial stop radius. If the maximum aberration exceeds a few percent, ray aiming (see the chapter "System Menu") may be required to get the rays in object space to correctly fill the stop surface. If ray aiming is switched on, the entrance pupil aberration will appear to be zero (or a very small residual value), because the distortion is accounted for by the ray tracing algorithm. This is used as a check that ray aiming is working correctly. The definition used here for pupil aberration is not intended to be complete or in agreement with other definitions. The sole purpose of this feature is to provide guidance as to whether or not ray aiming is required.

**Spot Diagrams**

---

**Standard**

*Purpose:*

Show spot diagrams.

*Settings:*

Item	Description
Pupil Pattern	The pupil pattern may be either hexapolar, square, or dithered. These terms refer to the pattern of rays as they appear in a pupil plane. Defocus the lens significantly to see the pupil pattern. Dithered spot diagrams are generated by pseudo-random rays which eliminate the symmetrical artifacts in the spot diagram typical of rectangular or hexapolar patterns. The pupil pattern is distorted to give the correct distribution of rays if pupil apodization is specified. There is no "best" pattern to use, each simply shows a different character of the spot diagram.
Reference Point	The spot diagrams by default are referenced to the real chief ray. The RMS and GEO spot sizes listed at the bottom of the diagram (and defined in the discussion section) are calculated assuming the chief ray is the "zero aberration" point. However, this option allows selection of two other reference points: the centroid or the middle. The centroid is defined by the distribution of rays traced. The middle is defined so that the maximum ray errors are equal in the x and y directions.

Item	Description
Show Scale	Scale bar is the default. Selecting "Airy Disk" will draw a circular ring around each spot showing the size of the Airy disk. The Airy disk radius is 1.22 times the primary wavelength times the working F/# of the system. If the Airy disk is larger than the spot, the Airy disk will set the scale size, otherwise the spot size will set the scale. Selecting "Square" will draw a box, centered on the reference point, whose width is twice the distance from the reference point to the outermost ray. Selecting "Cross" will draw a cross through the reference point location.
Wavelength	The wavelength number for which the calculation should be performed.
Field	The field number for which the calculation should be performed.
Surface Number	Selects the surface at which the spot diagram is to be evaluated. This is useful for evaluating intermediate images or vignetting.
Max Scale	Sets the maximum scale size of the display in microns. A setting of zero will generate an appropriate scale.
Delta Focus	The delta focus option is only used if through-focus spot diagrams are selected. This is the Z-axis spacing between spot diagram planes. Five spot diagrams will be shown for each field angle. The defocus will be -2, -1, 0, 1, and 2 times the delta focus provided. The delta focus units are in microns. For some systems the default delta focus may be too small to cause a change in the spot diagram structure.
Ray Density	The ray density specifies the number of hexapolar rings to be traced if a hexapolar or dithered pupil pattern is selected, or the square root of the number of rays if a rectangular pattern is selected. The more rays traced, the greater the accuracy of the RMS spot size, although the computation time increases. There are 6 rays in the first hexapolar ring, 12 in the second, 18 in the third, and so on.
Use Symbols	If checked, this option will draw different symbols rather than dots for each wavelength. This helps distinguish the various wavelengths.

*Discussion:*

The ray density has a maximum value based upon the number of fields displayed, the number of wavelengths defined, and available memory. Through-focus spot diagrams will trace half of the maximum number of rays possible on standard spot diagrams.

The GEO spot size listed on the plot for each field point is the distance from the reference point (which is either the chief ray at the primary wavelength, the centroid of all the rays traced, or the middle of the spot cluster) to the ray which is farthest away from the reference point. In other words, the GEO spot size is the radius of the circle centered at the reference point which encloses all the rays.

The RMS spot size is the root-mean-square radial size. The distance between each ray and the reference point is squared, and averaged over all the rays, and then the square root is taken. The RMS spot size gives a rough idea of the spread of the rays, since it depends upon every ray. The GEO spot size only gives information about the one ray which is farthest from the reference point.

The Airy disk radius is given by 1.22 times the primary wavelength times the working F/# of the system. This is the radius to the first dark ring of the Airy disk for a circular, uniformly illuminated entrance pupil. The Airy disk may be optionally drawn to give an idea of the scale of the plot. For example, if all the rays are well within the Airy disk, then the system is often said to be "diffraction limited". If the RMS spot size is significantly larger than the Airy disk radius, then the system is not diffraction limited. The threshold for diffraction limited performance depends upon which criterion is used. There is no absolute boundary at which the system becomes diffraction limited. The Airy disk circle shown is not an accurate representation of the diffraction dark ring shape or size if the system does not have uniform illumination or if vignetting is used to eliminate some of the rays.

ZEMAX does not plot vignettted rays on spot diagrams, nor are they used in computing the RMS or GEO spot sizes.

ZEMAX generates grids of rays based upon the wavelength weighting factors and the pupil apodization, if any. The wavelength with the largest weight uses the maximum grid size set by the "Ray Density" option. Wavelengths with lower weights use grids with fewer rays to maintain the correct representation in the diagram. Ray grids are also distorted to maintain the correct ray distribution, if apodization is specified. The RMS spot size stated on the

spot diagram considers the wavelength weighting and apodization factors. However, it is only an estimate of the RMS spot size based on the rays actually traced. It is not a very accurate estimate for some systems.

The image plane intercept coordinates of the reference point are shown underneath each spot diagram. If a surface other than the image plane is specified, then the coordinates are the intercept coordinates of the reference point on that surface. Since the reference point may be selected to be the centroid, this provides a convenient way of determining the centroid coordinates.

### **Through Focus**

#### *Purpose:*

Show spot diagrams as they change through focal plane shifts.

#### *Settings:*

The options are identical to the standard spot diagram.

#### *Discussion:*

The through focus spot diagrams are useful for estimating astigmatism, or for analyzing best focus or depth of focus.

### **Full Field**

#### *Purpose:*

Shows spot diagram with all field points on a common scale.

#### *Settings:*

The options are identical to the standard spot diagram.

#### *Discussion:*

The "Full Field" spot diagram type is similar to the "Standard" type, except all of the spots are plotted with respect to the same reference point, as opposed to a separate reference point for each field position. This provides some idea of how the spot would look relative to the other field points. For example, this can be used to determine if two closely spaced image points can be resolved. The "Full Field" spot diagram type is useless if the spot size is small compared to the total field size, because in this case the spots for each field will appear as simple "dots".

### **Matrix**

#### *Purpose:*

Show spot diagram as a matrix of individual diagrams, with each field along a row and each wavelength down a column.

#### *Settings:*

The options for the matrix spot diagram are similar to those for the standard spot diagram, except for the addition of the following option.

Item	Description
Ignore Lat. Color	If checked, this option will reference each spot diagram to the reference point for each field and wavelength independently. This in effect ignores the effects of lateral color which can displace the reference points for each wavelength.

#### *Discussion:*

The matrix representation is a convenient way of distinguishing the components of wavelength dependent aberrations.

## MTF

### Modulation Transfer Function (MTF)

#### *Purpose:*

Computes the diffraction modulation transfer function data for all field positions. This feature includes Diffraction Modulation Transfer Function (DMTF), Diffraction Real Transfer Function (DRTF), Diffraction Imaginary Transfer Function (DITF), Diffraction Phase Transfer Function (DPTF), and Square-Wave MTF (DSWM).

#### *Settings:*

Item	Description
Sampling	The size of the grid used to sample the OPD in the pupil. The sampling may be 8x8, 16x16, and so on. Although higher sampling yields more accurate data, calculation times increase rapidly. For systems with a few waves of OPD error or less, 32x32 is usually adequate. For systems with less than one wave of OPD, 16x16 is acceptable, and the computations will be roughly four times faster.
Show Diff Limit	Select whether or not the diffraction limited data should be displayed.
Max Frequency	Specify the maximum spatial frequency (in cycles per millimeter) plotted.
Wavelength	The wavelength number to be used in the calculation.
Field	The field number to be used in the calculation.
Type	Select either modulation, real, imaginary, phase, or square wave response.

#### *Discussion:*

The DMTF, DRTF, DITF, DPTF, and DSWM functions present the modulation (the magnitude of the real and imaginary parts), the real part alone, the imaginary part alone, the phase, or the square wave response, respectively. The Square wave MTF is the modulation response for a square wave target of the specific frequency, as opposed to the response to a sine wave target for the other plots. The square wave response is computed from the DMTF data using the following formula:

$$S(v) = \frac{4}{\pi} \left[ \frac{M(v)}{1} - \frac{M(3v)}{3} + \frac{M(5v)}{5} - \frac{M(7v)}{7} + \dots \right]$$

where  $S(v)$  is the square wave response,  $M(v)$  is the sinusoidal modulation response, and  $v$  is the spatial frequency.

The diffraction calculations are accurate as long as the peak-to-valley OPD in the pupil is less than roughly 8 waves for a 32 x 32 sampling density (adjust linearly for other sampling densities). If the OPD exceeds this number, then the wavefront sampling is too coarse and aliasing occurs. Aliasing will result in inaccurate data. ZEMAX will attempt to detect when aliasing occurs, and issue an appropriate error message. However, ZEMAX cannot automatically detect when the sampling is too low in all cases, especially in the presence of very steep slopes on the wavefront phase.

When the OPD in waves is large, such as more than 10 waves, it is probably a good idea to switch to geometric MTF instead of diffraction MTF. For these highly aberrated systems, the geometric MTF is very accurate, especially at low spatial frequencies.

The cutoff frequency at any one wavelength is given by one over the wavelength times the working F/#. ZEMAX computes the working F/# at each wavelength for each field for the sagittal and tangential response separately. This yields accurate MTF data even for systems with anamorphic and chromatic distortion, such as those incorporating cylinders or gratings. Because ZEMAX does not account for vector diffraction, the MTF data may not be accurate for systems faster than about F/1.5 (the deterioration in accuracy is gradual). For these systems, the OPD fan data are more fundamental and therefore more reliable indicators of performance. If the system is not too close to the diffraction limit, the geometric MTF may prove useful.

If shown, the diffraction limit curve is for the aberration free MTF computed on-axis. In the event the on-axis rays cannot be traced, as might be the case for a system designed to work only at off-axis field angles, then the first field position is used instead to compute the "diffraction limited" MTF response.



The spatial frequency scale of the MTF plot is always in cycles per mm in image space, which is the correct term for sinusoidal MTF response. The term line pairs per mm is often used, but strictly speaking line pairs per mm only applies to bar, as opposed to sinusoidal targets. ZEMAX uses these terms interchangeably, as is common in the industry. MTF is always measured in image space, so any magnification of the system needs to be considered when determining spatial frequency response for object space.

### **Through Focus MTF**

#### *Purpose:*

Computes the diffraction modulation transfer function data through focus for all field positions at a specific spatial frequency. This feature includes Diffraction Through Focus MTF (DTFM), Diffraction Through Focus Real TF (DTFR), Diffraction Through Focus Imaginary MTF (DTFI), Diffraction Through Focus Phase MTF (DTFP), and Diffraction Through Focus Square-Wave MTF (DTFS).

#### *Settings:*

Item	Description
Sampling	See the MTF description for details.
Delta Focus	The range of defocus used.
Frequency	The spatial frequency (in cycles per millimeter) for which data is plotted.
# Steps	The number of focal planes at which the data is computed. A smooth curve is drawn through the computed points. More steps yield higher accuracy and longer computation times.
Wavelength	The wavelength number to be used in the calculation.
Field	The field number for which the calculation should be performed.
Type	Select either modulation, real, imaginary, phase, or square wave response.

#### *Discussion:*

See the modulation transfer function for details.

### **Surface MTF**

#### *Purpose:*

Displays the MTF data as a 3D surface, contour, grey scale or false color map. This plot is useful for visualizing the MTF response for object orientations other than purely sagittal or tangential.

#### *Settings:*

Item	Description
Sampling	See the MTF description for details.
Rotation	Rotation specifies how the surface plots are rotated for viewing; either 0, 90, 180, or 270 degrees.
Scale	The scale factor overrides the automatic vertical scaling set by the program on the surface plots. Generally, this value should be set to unity. The scale factor can be greater than unity to vertically stretch the plot, or less than unity to compress it.
Wavelength	The wavelength number to be used in the calculation.
Field	The field number determines for which defined field position the calculation should be performed.
Show As	Choose surface plot, contour map, grey scale, or false color map as the display option.

#### *Discussion:*

The regular MTF plot is just two orthogonal cross sections through the surface MTF plot. This plot is primarily qualitative.

## **Geometric Transfer Function**

### *Purpose:*

Computes the geometric transfer function, which is a close approximation to the diffraction MTF based upon ray aberration data.

### *Settings:*

Item	Description
Sampling	Sampling refers to the number of hexapolar rings of rays traced at each wavelength.
Max Freq	The maximum spatial frequency (in cycles per millimeter) for which data is plotted.
Wavelength	The wavelength number to be used in the calculation.
Field	The field number for which the calculation should be performed.
Mult by Diff Limit	When checked, will scale the geometric MTF by the diffraction limited MTF to yield a more realistic result for systems with small aberrations. Should always be used.

### *Discussion:*

The geometric MTF is a useful approximation to the diffraction MTF if the system is not close to the diffraction limit. The primary advantage to using the geometric MTF is for systems which have too many waves of aberration to permit accurate calculation of the diffraction MTF. The geometric MTF is also very accurate at low spatial frequencies for systems with large aberrations.

## **Through Focus GMTF**

### *Purpose:*

Computes the geometric modulation transfer function data through focus at a specific spatial frequency.

### *Settings:*

Item	Description
Sampling	See the Geometric MTF section for details.
Delta Focus	The range of defocus used.
Frequency	The spatial frequency (in cycles per millimeter) for which data is plotted.
# Steps	The number of focal planes at which the data is computed. A smooth curve is drawn through the computed points. More steps yield higher accuracy and longer computation times.
Wavelength	The wavelength number to be used in the calculation.
Field	The field number for which the calculation should be performed.
Mult by Diff Limit	When checked, will scale the geometric MTF by the diffraction limited MTF to yield a more realistic result for systems with small aberrations. Should always be used.

### *Discussion:*

See the Geometric Transfer Function section for details.

## **PSF**

---

## **FFT Point Spread Function**

### *Purpose:*

Computes the diffraction point spread function using the Fast Fourier Transform (FFT) method.

### *Settings:*

Item	Description
Sampling	See the MTF description for details.

Item	Description
Display	The display size indicates what portion of the computed data will be drawn. The display grid can be any size from 8 x 8 up to twice the sampling grid size. Smaller display sizes will show less data, but at higher magnification for better visibility.
Rotation	Rotation specifies how the surface plots are rotated for viewing; either 0, 90, 180, or 270 degrees.
Wavelength	The wavelength number to be used in the calculation.
Field	The field number for which the calculation should be performed.
Type	Select linear (intensity), logarithmic (intensity), or phase.
Maximum Scale	By default, as the sampling increases, both the spatial extent of the plot and the frequency of sampling in the spatial domain increase. Selecting "maximum scale" overrides this default behavior and places all the increase of sampling into the spatial domain, leaving the spatial period constant. This yields maximum accuracy at the expense of longer computation times and a larger spacing between points. However, systems with larger amounts of aberration can be analyzed if "maximum scale" is selected.
Show As	Choose surface plot, contour map, grey scale, or false color map as the display option.

*Discussion:*

The FFT method of computing the PSF is very fast, however, a few assumptions are made which are not always valid. The slower, but more general Huygens method makes no such assumptions, and is described in the next section.

The FFT PSF computes the intensity of the diffraction image formed by the optical system of a single point source in the field. The Intensity is computed on an imaginary plane which lies perpendicular to the incident chief ray at the reference wavelength. The reference wavelength is the primary wavelength for polychromatic computations, or simply the wavelength being used for monochromatic calculations. Because the imaginary plane lies normal to the chief ray, and not the image surface, the FFT PSF computes overly optimistic (a smaller PSF) results when the chief ray angle of incidence is not zero. This is often the case for systems with tilted image planes, wide angle systems, systems with aberrated exit pupils, or systems far from the telecentric condition.

For systems where the chief ray is nearly normal (less than perhaps 20 degrees) and the exit pupil aberrations are negligible, then the FFT PSF is accurate and generally much faster than the Huygens PSF method. When in doubt, both PSF methods should be employed for comparison.

The FFT PSF algorithm exploits the fact that the diffraction PSF is related to the Fourier transform of the complex amplitude of the wavefront in the exit pupil of the optical system. The amplitude and phase in the exit pupil are computed for a grid of rays, an FFT is performed, and the diffraction image intensity is computed.

There is a tradeoff between the sampling grid size in the pupil, and the sampling period in the diffraction image. For example, to decrease the sampling period in the diffraction image, the sampling in the pupil must decrease. This is done by "stretching" the pupil sampling grid so that it overfills the pupil. This process means fewer points lie within the pupil, which decreases the accuracy of the computation.

For larger sampling grids, ZEMAX stretches the grid on the pupil to yield an increase in the number of points that lie on the pupil, while simultaneously yielding closer sampling in the diffraction image. Each time the grid size is doubled, the pupil sampling increases by the square root of 2, the image plane sampling decreases by a square root of 2, and the total width of the diffraction image grid increases by a factor of the square root of 2. All ratios are approximate, and asymptotically correct for large grids.

The stretching is referenced to a grid size of 8 x 8. The 8 x 8 grid of points is placed over the pupil, and the points that lie within the pupil are actually traced. For this grid size, the distance between points in the diffraction image plane is given by

$$\Delta X = \lambda F \frac{n-2}{2n} ,$$

where F is the working F/# (not the same as the image space F/#),  $\lambda$  is the shortest defined wavelength, and n is the number of points across the grid, in this case 8 (the sampling grid is of size n x n). The -2 factor is due to

the fact the pupil is not centered on the grid (since  $n$  is even), but is offset at  $n/2 + 1$ . The  $2n$  in the denominator is due to the zero-padding described later.

For grids larger than  $8 \times 8$ , the grid is stretched in pupil space by a factor of  $\sqrt{2}$  each time the sampling density doubles. The general formula for the sampling in image space is then

$$\Delta X = \lambda F \frac{n-2}{2n} \left[ \frac{8}{n} \right]^{1/2},$$

and the total width of the image data grid is

$$W = \Delta x (2n - 1).$$

Since the stretching of the pupil grid decreases the number of sample points in the pupil, the effective grid size (the size of the grid that actually represents traced rays) is smaller than the sampling grid. The effective grid size increases as the sampling increases, but not as quickly. The following table summarizes the approximate effective grid size for various sampling density values.

**EFFECTIVE GRID SIZES FOR PSF CALCULATIONS**

Sampling Grid	Approximate Effective Sampling
8 x 8	8 x 8
16 x 16	11 x 11
32 x 32	16 x 16
64 x 64	23 x 23
128 x 128	32 x 32
256 x 256	45 x 45

The sampling is also a function of wavelength. The discussion above is only valid for the shortest wavelength used in the calculation. If the computation is polychromatic, then the longer wavelengths will be scaled to have smaller effective grids. The scale factor used is the ratio of the wavelengths. This should be considered when selecting sampling grids for systems with broad wavelength bands. For polychromatic computations, the data for shorter wavelengths is more accurate than for longer wavelengths.

There are times when the default behavior of stretching the grid to yield a lower effective sampling rate is undesirable. For example, when the system has a large amount of aberration (more than a few waves), the maximum possible sampling in the pupil is desired. Selecting "Maximum Scale" will turn off the grid stretching and yield the maximum pupil sampling as well as the maximum spatial scale, at the expense of a larger distance between points in the image plane.

Once the sampling is specified, ZEMAX doubles the array size in a process called "zero padding". This means for a  $32 \times 32$  sampling, ZEMAX uses the center portion of a  $64 \times 64$  grid. Therefore, the diffraction point spread function will be distributed over a  $64 \times 64$  size grid. The sampling in the image space is always twice the pupil sampling. Zero padding is performed to reduce aliasing.

There is one question that commonly arises when users compare the calculated radius to the first dark Airy ring using the PSF, PSF Cross Section, or Diffraction Encircled Energy features. The Airy disk radius is given by  $1.22\lambda F$ , and for a perfect optical system, the PSF should be zero at this radius. However, for a number of reasons, the lowest intensity point computed by ZEMAX may not be exactly at this radius.

There are two reasons for this apparent discrepancy. First, ZEMAX always uses a finite number of points to sample the pupil. This means the perfect circular exit pupil assumed by the Airy disk formulation can only be approximated. The rectangular grids used to perform the sampling impose a lack of circular symmetry, and this in turn yields small but non-zero values for the intensity in the region of the dark ring.

The second reason is more subtle. Because the intensity is very close to zero in the dark ring, the exact minimum point in the intensity is very sensitive to the sampling errors. In other words, a small shift in radius of a few percent may include very little additional energy, and so the exact minimum point is not very distinct. For these reasons,

the Airy disk radius test is poor means for evaluating the accuracy of the PSF, PSF Cross Section, and Encircled Energy calculations.

### **Huygens Point Spread Function**

#### *Purpose:*

Computes the diffraction point spread function using direct integration of Huygens wavelets method.

#### *Settings:*

Item	Description
Pupil Sampling	Selects the size of the grid of rays to trace to perform the computation. Higher sampling densities yield more accurate results at the expense of longer computation times.
Image Sampling	The size of the grid of points on which to compute the diffraction image intensity. This number, combined with the image delta, determine the size of the area displayed.
Image Delta	The distance in microns between points in the image grid.
Rotation	Rotation specifies how the surface plots are rotated for viewing; either 0, 90, 180, or 270 degrees.
Wavelength	The wavelength number to be used in the calculation.
Field	The field number for which the calculation should be performed.
Type	Select linear (intensity), or logarithmic (intensity).
Scale by Strehl	If checked, the peak intensity displayed will be scaled by the computed Strehl ratio, so that the plot may be directly compared to other PSF plots. Otherwise, the peak is always normalized to unity.
Show As	Choose surface plot, contour map, grey scale, or false color map as the display option.

#### *Discussion:*

One way of considering the effects of diffraction is to imagine each point on a wavefront as a perfect point source with an amplitude and phase. Each of these point sources radiates a spherical "wavelet", sometimes called a "Huygens wavelet" after Huygens, who first proposed the model. The diffraction of the wavefront as it propagates through space is given by the interference, or complex sum, of all the spherical wavelets radiated.

To compute the Huygens PSF, a grid of rays are launched through the optical system, and each ray represents a particular amplitude and phase wavelet. The diffraction intensity at any point on the image surface is simply the complex sum of all these wavelets, squared.

Unlike the FFT PSF, ZEMAX computes the Huygens PSF on an imaginary plane tangent to the image surface at the chief ray intercept. Note the imaginary plane is normal to the normal of the surface, not the chief ray. Therefore, the Huygens PSF accounts for any local tilt in the image surface caused by either the image plane slope, the chief ray incidence angle, or both.

Furthermore, the Huygens method accounts for the evolving shape of the diffraction image as the beam propagates along the image surface. This is an important effect if the image surface is highly tilted with respect to the incoming beam.

Another advantage to the Huygens PSF method is that any grid size, and any grid spacing, may be selected by the user. This allows direct comparison between PSF's from two different lenses, even if the F/#'s or wavelengths are different.

The only disadvantage of the Huygens PSF is speed. Direct integration is not very efficient when compared to the FFT method (see the previous section for details), and the Huygens PSF may take many times as long to compute. The computation time depends approximately upon the pupil grid size squared times the image grid size squared, times the number of wavelengths. ZEMAX automatically accounts for any symmetry the system may have.

### **FFT PSF Cross Section**

#### *Purpose:*

This feature plots cross sections through the diffraction Point Spread Function.

*Settings:*

Item	Description
Sampling	See the MTF description for details.
Row/Col	The row or column to display. For a sampling of 32 x 32, there are 64 rows and 64 columns (see the discussion section of the Point Spread Function feature). Whether a row or a column is used depends upon the "Type" setting.
Wavelength	The wavelength number to be used in the calculation.
Field	The field number for which the calculation should be performed.
Type	Select X or Y cross sections, either linear, logarithmic, or phase. X cross sections are called rows, and Y cross sections are called columns, but this is arbitrary
Maximum Scale	See the Point Spread Function description for details.

*Discussion:*

See the discussion section of the Point Spread Function. Those comments also apply to this feature.

The cross sections are taken directly from the PSF data. Because the PSF is computed directly from the phase in the exit pupil, the orientation of the coordinate system may not be correct in all cases. What constitutes positive x or y may not agree with data presented in image space coordinates such as the spot diagram.

## **Wavefront**

### **Wavefront Map**

*Purpose:*

Displays the wavefront aberration.

*Settings:*

Item	Description
Sampling	See the MTF description for details.
Rotation	Rotation specifies how the surface plots are rotated for viewing; either 0, 90, 180, or 270 degrees.
Scale	The scale factor is used to override the automatic vertical scaling set by the program on the surface plots. The scale factor can be greater than unity to vertically stretch the plot, or less than unity to compress it.
Wavelength	The wavelength number to be used in the calculation.
Field	The field number for which the calculation should be performed.
Reference Primary	To By default, the wavefront aberration is referenced to the reference sphere for the wavelength being used. If this box is checked, then the primary wavelength reference sphere will be used instead. In other words, checking this box will cause the data to exhibit the effects of lateral color.
Use Exit Pupil Shape	By default, the shape of the pupil is distorted to show the approximate shape of the exit pupil as seen from the on axis chief ray image point. If this box is unchecked, then instead the plot will be scaled to circular entrance pupil coordinates, no matter how distorted the exit pupil may actually be.
Show As	Choose surface plot, contour map, grey scale, or false color map as the display option.

*Discussion:*

This feature is very similar to the Interferogram feature described below. The text version of this feature lists the OPD values in eight columns to save space. The OPD values are listed eight per row, plus an eight digit integer code of the form nnnnnnnn, where each n is either 0 (the ray did not pass the system) or 1 (the ray passes all the way to the image surface). The pass codes correspond to the OPD values.

## **Interferogram**

### *Purpose:*

Generates and displays interferograms.

### *Settings:*

Item	Description
Sampling	The size of the grid used to sample the OPD in the pupil. The sampling may be 8x8, 16x16, and so on. Although higher sampling yields more accurate data, calculation times increase.
Fill Factor	At each grid point where the OPD is calculated, ZEMAX will attempt to "fill" neighboring pixels with an estimate of the OPD there, to speed execution time. A fill factor of 1 will result in a three-by-three pixel filling, 2 will result in a five-by-five pixel filling, and so on.
Scale Factor	Determines the number of fringes per wave of OPD. Useful for modeling double pass interferometers (i.e. use a scale factor of two).
Show As	Choose pseudo grey, contour map, grey scale, or false color map as the display option.
Wavelength	The wavelength is the wavelength number to be used in the calculation.
Field	The field number for which the calculation should be performed.
X-Tilt	The number of waves of tilt to add in the x-direction after applying the scale factor.
Y-Tilt	The number of waves of tilt to add in the y-direction after applying the scale factor.

### *Discussion:*

Interferograms may require very long print times, and for high densities, very long calculation times as well. If the fill factor is set too high, the pseudo gray plots may become washed out.

## **RMS**

---

### **RMS vs. Field**

### *Purpose:*

Plots RMS radial, x, and y spot size, RMS wavefront error, or Strehl ratio as a function of field angle; either monochromatic or polychromatic.

### *Settings:*

Item	Description
Ray Density	If the method is Gaussian quadrature, then the ray density specifies the number of radial rays to be traced. The more rays traced, the greater the accuracy, although the computation time increases. The maximum number is 12 which is sufficient for pupil aberrations up to order 24. If the method is rectangular array, then the ray density indicates the grid size. Rays outside the circular entrance pupil are ignored. See the "Discussion" section for details.
Field Density	The field density is the number of points between zero degrees and the maximum field angle specified at which the RMS/Strehl ratio is calculated, intermediate values are interpolated. A maximum of 100 field points is allowed.
Maximum Scale	Sets the maximum vertical scale for the plot. Zero results in automatic scaling.
Method	Selects either Gaussian quadrature or rectangular array. The Gaussian quadrature method is very fast and accurate, but only works if there is no vignetting. If any rays are to be vignetted, then rectangular array is more accurate.
Data	Selects either wavefront error, spot radius, spot x-direction, spot y-direction, or Strehl ratio.

Item	Description
Refer To	Select either chief ray or centroid. For monochromatic calculations, the specified wavelength is used for reference. For polychromatic calculations, the primary wavelength is used for reference. Both reference points subtract out wavefront piston. The centroid reference mode also subtracts out the tilt of the wavefront, which yields smaller RMS values.
Use Dashes	Selects either color (for color screens and plotters) or dashes (for monochrome screens and plotters).
Wavelength	Select "All" to display data for each wavelength and a polychromatic computation, select any one wavelength to plot monochromatic data, or select "Poly Only" which only plots the polychromatic data.
Show Diff Limit	If checked, then a horizontal line indicating the diffraction limited response will be drawn on the plot. For RMS radius, x, or y; the diffraction limit is assumed to be 1.22 times the working F/# times the wavelength (primary wavelength if polychromatic) on axis. The change in the diffraction limit due to changes in F/# with field are ignored; a single value is used across the range of the plot. For Stehl ratio, 0.8 is used, and for RMS wavefront, 0.072 waves is used. These are all approximate indicators for convenience only; the actual meaning of "diffraction limited" may be open to interpretation.

### *Discussion:*

This feature calculates the RMS error or Strehl ratio as a function of field angle for each wavelength and gives a wavelength-weighted polychromatic result as well.

Two different methods of calculation are used; either a Gaussian quadrature method, or a rectangular array of rays. For the Gaussian quadrature method, the rays traced are arranged in a radial pattern with an optimal weighting to estimate the RMS with a minimum number of rays. The method is described in a paper by G. W. Forbes (JOSA A 5 P1943). Although the method is very efficient, the algorithm is not accurate if some of the rays are clipped due to surface apertures. Vignetting implemented with vignetting factors does not cause rays to be clipped, but surface apertures may clip rays.

For wavefront RMS computations, ZEMAX automatically subtracts out the mean OPD, which yields in effect the standard deviation rather than the true RMS. However, ZEMAX uses the term RMS in this case to comply with common conventions in the optics industry.

To compute the RMS wavefront in systems with surface apertures requires the use of the rectangular array method, and a larger number of rays for sufficient accuracy.

### **RMS vs. Focus**

#### *Purpose:*

Plots RMS radial, x, and y spot size, RMS wavefront error, or Strehl ratio as a function of focus change.

#### *Settings:*

Item	Description
Ray Density	If the method is Gaussian quadrature, then the ray density specifies the number of radial rays to be traced. The more rays traced, the greater the accuracy, although the computation time increases. The maximum number is 12 which is sufficient for pupil aberrations up to order 24. If the method is rectangular array, then the ray density indicates the grid size. Rays outside the circular entrance pupil are ignored. See the "Discussion" section for details.
Focus Density	The focus density is the number of points between the minimum and maximum focus shift specified at which the RMS/Strehl ratio is calculated, intermediate values are interpolated. A maximum of 100 points is allowed.
Max Scale	Sets the maximum vertical scale for the plot. Zero results in automatic scaling.
Method	Selects either Gaussian quadrature or rectangular array. The Gaussian quadrature method is very fast and accurate, but only works if there is no vignetting. If any rays are to be vignetted, then rectangular array is more accurate.



Item	Description
Data	Selects either RMS wavefront error, RMS spot size (radial), RMS X-direction, RMS Y-direction, or Strehl ratio.
Use Dashes	Selects either color (for color screens and plotters) or dashes (for monochrome screens and plotters).
Refer To	Select either chief ray or centroid. For monochromatic calculations, the specified wavelength is used for reference. For polychromatic calculations, the primary wavelength is used for reference. Both reference points subtract out wavefront piston. The centroid reference mode also subtracts out the tilt of the wavefront, which yields smaller RMS values.
Wavelength	Select "All" to display data for a polychromatic computation, select any one wavelength to plot monochromatic data.
Min Focus	The minimum value of the defocus to plot. The units are lens units.
Max Focus	The maximum value of the defocus to plot. The units are lens units.
Show Diff Limit	If checked, then a horizontal line indicating the diffraction limited response will be drawn on the plot. For RMS radius, x, or y; the diffraction limit is assumed to be 1.22 times the working F/# times the wavelength (primary wavelength if polychromatic) on axis. The change in the diffraction limit due to changes in F/# with field are ignored; a single value is used across the range of the plot. For Strehl ratio, 0.8 is used, and for RMS wavefront, 0.072 waves is used. These are all approximate indicators for convenience only; the actual meaning of "diffraction limited" may be open to interpretation.

### *Discussion:*

This feature calculates the RMS error or Strehl ratio as a function of a change in focus position for each field position. The method of calculation is identical to that described in the previous section on RMS vs. Field; see that section for a detailed discussion.

ZEMAX simply adds the specified focus shift to the value of the thickness of the surface prior to the image plane. If the optical system has an odd number of mirrors, this surface normally would have a negative thickness, and therefore negative focus values move the image plane farther away from the last component. For systems with an even number of mirrors, then negative focus values move the image plane closer to the last component.

## **Encircled energy**

### **Diffraction**

#### *Purpose:*

Encircled energy diagram. This is the percentage of total energy enclosed as a function of distance from either the chief ray or the image centroid at the image of a point object.

#### *Settings:*

Item	Description
Sampling	See the MTF description for details.
Type	The analysis type option specifies how the encircled energy is calculated; either encircled (radial), X-only, Y-only, or ensquared.
Maximum Distance	This setting overrides the default scaling. The units are microns. To choose the default scaling option, enter zero.
Use Dashes	Selects either color (for color screens and plotters) or dashes (for monochrome screens and plotters).
Ref To	Select either chief ray or centroid as the reference point.
Wavelength	The wavelength number to be used in the calculation.
Field	The field number for which the calculation should be performed.
Show Diff Limit	If checked, the diffraction limited results are computed and displayed. Apertures are accounted for, and the on-axis field is used.
Maximum Scale	See the Point Spread Function description for details.

Item	Description
Use Huygens PSF	If checked, the more accurate but slower Huygens PSF method is used to compute the PSF. This option should always be used if the image surface is tilted, or if the chief ray is not close to normal to the image surface.

*Discussion:*

See the discussion section of the FFT and Huygens Point Spread Function. Those comments also apply to this feature.

The accuracy of the diffraction encircled energy calculation is limited by the magnitude and slope of the OPD error and the sampling density used. If the sampling density is insufficient, ZEMAX will issue an error message indicating that the data is inaccurate. To increase the accuracy, increase the sampling density or decrease the OPD error.

No matter how high the sampling density is, this will always be an approximate calculation. Because the beam size is limited by the optical system (i.e. the stop) the beam has "compact support" in the near field, and therefore cannot have compact support in the far field where the PSF is calculated (in other words, a band-limited signal in one domain cannot also be band-limited in the Fourier domain).

Setting "Maximum Scale" will yield a larger maximum distance scale, at the expense of larger distance between sample points (see the FFT Point Spread Function feature for details). ZEMAX will detect when the accuracy is low, and will issue an error message that will appear on the plot. ZEMAX will automatically switch to the "maximum scale" setting when it is required for accuracy, even if that option is not selected on the options settings box.

If shown, the diffraction limit curve is for the aberration free encircled energy computed on-axis. In the event the on-axis rays cannot be traced, as might be the case for a system designed to work only at off-axis field angles, then the first field position is used instead to compute the "diffraction limited" MTF response.

**Geometric**

*Purpose:*

Computes encircled energy using ray-image plane intercepts.

*Settings:*

Item	Description
Sampling	Specifies the number of hexapolar rings of rays to be traced.
Type	The analysis type option specifies how the encircled energy is calculated; either encircled (radial), X-only, Y-only, or ensquared.
Maximum Distance	This setting overrides the default scaling. The units are microns. To choose the default scaling option, enter zero.
Use Dashes	Selects either color (for color screens and plotters) or dashes (for monochrome screens and plotters).
Ref To	Select either chief ray or centroid as the reference point.
Wavelength	The wavelength number to be used in the calculation.
Field	The field number for which the calculation should be performed.
Mult by Diff Limit	If checked, ZEMAX approximates the diffraction encircled energy by scaling the geometric data by the theoretical diffraction limit curve. The diffraction limit curve ZEMAX uses is based upon the unobscured circular pupil. The only way to compute the obscured pupil diffraction limit function would be to perform the exact diffraction calculation, in which case the diffraction encircled energy feature should be used instead. The diffraction limit approximation is only useful for systems with unobscured pupils and modest field angles, since the approximation ignores the change in F/# with field.

*Discussion:*

The X- and Y-only options will compute the fraction of rays which are contained with plus or minus the specified distance from either the chief ray or the image centroid. If a scale of 10 microns is shown, then the region enclosed

is 20 microns across (and infinite in the other direction). The geometric encircled energy is not a good indicator of performance if the system is close to diffraction limited.

### **Line/Edge Response**

*Purpose:*

Computes the geometric response to a line object and an edge object.

*Settings:*

Item	Description
Sampling	Specifies the number of hexapolar rings of rays to be traced.
Maximum Radius	The maximum radius setting overrides the default scaling. The units are microns. To choose the default scaling option, enter zero.
Wavelength	The wavelength number to be used in the calculation.
Field	The field number to be used in the calculation.

*Discussion:*

The line response function (or line spread function, LSF) is the cross section of the intensity pattern of the image of a line object. The edge response function (ERF) is the cross section of the intensity pattern of the image of an edge (a semi-infinite plane). The tangential and sagittal data refer to the orientation of the line or edge. This is a geometric calculation.

## **Illumination**

---

### **Relative illumination**

*Purpose:*

Computes the relative illumination as a function of radial field coordinate for a uniform Lambertian scene.

*Settings:*

Item	Description
Ray Density	The number of rays along one side of a square array of rays used to integrate the illumination of the exit pupil. A value of 10 will trace about $10 \times 10 \times \pi / 4$ or 78 rays. Higher ray densities yield more accurate results at the expense of longer computation times.
Field Density	The number of points along the radial field coordinate to compute the relative illumination for. Larger field densities yield smoother curves.
Use Polarization	If checked, then the polarization transmission, including thin film, bare surface, and internal transmittance effects will all be accounted for (ZEMAX-EE Only).

*Discussion:*

This feature computes the relative illumination (RI) as a function of radial y field coordinate. RI is defined as the intensity of illumination on a small patch of image surface normalized to the illumination at zero field. The computation considers apodization, vignetting, apertures, aberrations of both the image and pupils, variations in F/#, chromatic aberrations, image surface shape, angle of incidence, and optionally, polarization effects assuming unpolarized light. The method is based upon one described in "Relative illumination calculations" by M. Rimmer, SPIE Vol. 655, pp. 99 (1986). The published method was extended to include apodization, transmission, and polarization effects. The method assumes a uniform, Lambertian object. The relative illumination is computed by numerical integration of the effective area of the exit pupil as seen from the image point(s). The integration is carried out in direction cosine space using a uniform grid in the image cosine space.

Note that the RI computation will not in general yield a cosine-fourth law curve, because the cosine-fourth law is in fact a rough approximation based upon a thin, aberration free lens with the stop at the lens illuminating a plane image surface. For general systems including telecentric systems, systems with pupil or image aberrations, or vignetting, the RI can be computed using an integration of the solid angle or effective area of the exit pupil as seen from the image location.

## Image Analysis

### *Purpose:*

The image analysis feature has many applications. It can be used to model extended sources, analyze useful resolution, display distortion, represent the appearance of imaged objects, provide intuition as to image rotation, compute coupling efficiency from extended or point objects into optical fibers, display beam "foot prints", and show surface plots of illumination on any surface, to name just a few. The image analysis feature uses special IMA files to describe the object to be imaged. The IMA file is described in the discussion section which follows.

See the chapter "ZEMAX Support Files" for a list of sample IMA files and some suggested applications.

### *Settings:*

Item	Description
Field Size	This value defines the full width of the square image file in field coordinates, which may be either lens units or degrees, depending upon the current field definition (heights or angles, respectively).
Image Size	This value sets the size of the scale bar which is superimposed on the image diagram. It has no effect on the actual size of the image. The image size is set by the object scale and the magnification and aberrations of the system. The default may not be acceptable to see the desired portion of the image.
Parity	The "Even" setting leaves the object as it would appear when viewed looking down the negative Z axis in object space. The parity can be set to "Odd" which reverses the object from top to bottom.
Rotation	The rotation can be set to any angle in degrees. The algorithm actually rotates the object before tracing the rays, so this feature can be used to switch from tangential to sagittal orientation of bar targets, for example.
# Rays x 1000	This setting determines approximately how many rays will be traced. The number of rays traced is approximately 1000 times the specified value. The reason the number of rays is only approximate is because the distribution of rays over the pixels in the image must be uniform. For example, if there are 300 pixels in an image file, then at least 300 rays will be traced, even if a value of 1 is selected. The distribution of rays at each wavelength is in proportion to the wavelength weights.
Show	Select either "Image Diagram", "Surface Plot", or "3D Histogram". The image diagram shows a diagram similar to a spot diagram and displays the percent efficiency (see the discussion for a description of this calculation). The surface plot and 3D histogram options display a 3-dimensional image of the intensity represented by the rays landing in each pixel of the detector.
Source	The source may be uniform or lambertian. The uniform setting weights all rays equally. Lambertian weights all rays by the cosine of the angle the ray makes with the axis of the object surface.
Wavelength	The wavelength number to be used in the calculation.
Field	The image file may be centered on any defined field position. This permits a small target such as a bar chart to be moved to any location in the field of view. The resulting image is then centered on the chief ray coordinate of this field position.
Image File Name	The name of the .IMA image file. This file must reside in the default directory for ZEMAX. See the discussion section for a full description of the IMA file format.
Surface	The surface number at which to evaluate the rays. The default is the image surface. Other surfaces may be selected, for example, to visualize the beam footprint on an optical surface.
Numerical AP	Rays approaching the image surface with a numerical aperture (defined as $n\sin U$ ) greater than the value specified are ignored. The default value of zero indicates all rays are accepted.
X-pixels	The number of pixels across the X direction on the detector. This value is only used in the surface plot and 3D histogram output options.

Item	Description
Y-pixels	The number of pixels across the Y direction on the detector. This value is only used in the surface plot and 3D histogram output options.
X-pixel size	The size in lens units of each pixel measured in the X direction.
Y-pixel size	The size in lens units of each pixel measured in the Y direction.
Use Symbols	If checked, this option will draw different symbols rather than dots for each wavelength. This helps distinguish the various wavelengths.
Fast Draw	If checked, this option will draw the "3D Histogram" and "Surface Plot" output options (see the "Show" option) as wireframe, rather than hidden line plots. This is dramatically faster, at the expense of the clarity of the plot. This option has no effect if "Image Diagram" is selected.
Edit IMA File	Pressing this button will invoke the Windows Notepad editor which allows modification of the currently selected IMA file.

*Discussion:*

ZEMAX supports two different IMA file formats, one ASCII and one binary. Whichever file format is used, the file must end in the extension IMA. ZEMAX will distinguish between the two types of file formats automatically.

The ASCII image file is a text file which ends in the extension .IMA. At the top of the file is one number which indicates the size of the file in pixels. The remaining rows and columns contain the pixel data, with one character to each pixel. All IMA files must be square, with n x n pixels defined. For example, a 7 x 7 representation of the letter "F" could be described by the following IMA file:

```
7
0111110
0100000
0100000
0111100
0100000
0100000
0100000
```

Note that the single entry "7" starts the file, and it is followed by a carriage return. Then there are 7 rows of 7 columns each, and each row is terminated with a carriage return. The columns are not separated by a space or any other character. The image file must be square. ZEMAX will attempt to allocate enough memory to hold the image file and will report an error if there is not enough memory.

The "intensity" at each pixel can be any digit between 0 and 9. The number of rays each pixel will generate is proportional to this value. Pixels with a value of 0 do not radiate any rays.

The binary IMA file format is more complicated than the ASCII format, and binary IMA files cannot be edited with a text editor. However, the binary IMA files are dramatically more powerful. Each pixel in the binary IMA file is represented by an unsigned byte, which means there are 256 "gray-scale" levels of intensity. Furthermore, each wavelength can be assigned a separate pixel map. Therefore, very realistic photograph like extended sources can be modeled.

The binary IMA file format requires 3 16-bit header values. The first 16-bit value is a signed integer that must be equal to zero. The second 16-bit signed integer is the width of the pixel map in pixels, which can be any number from 1 to 2000. The third 16-bit signed integer is the number of pixel maps, which correspond to the number of colors (or wavelengths) represented in the file.

For example, a 3-color binary pixel map of a 50 by 50 image would have 6 bytes of header (0, 50, and 3), followed by 2500 bytes for color 1, then 2500 bytes for color 2, then 2500 bytes for color 3, for a total of 7506 bytes. The data for each color is stored by columns for each row (the column index changes faster than the row index).

The rays generated by each pixel are chosen randomly from coordinates within the pixel cell. The entrance pupil coordinates are also chosen randomly for each ray. The distribution of rays is uniform over the pixel and over the circular paraxial entrance pupil (if ray aiming is used, then there may be some pupil distortion). For the ASCII IMA files, The number of rays generated by each pixel is equal to the pixel intensity times the number of wavelengths times the ray density. The wavelength used for each ray is selected randomly in proportion to the wavelength

weights provided on the wavelength data screen. For the binary IMA files, the number of rays generated from each pixel is proportional to the ray density times the fractional intensity relative to 256.

The field size determines the physical size of the image file as seen by the optical system. For example, if a 30 x 30 pixel size image file is used, and the field size is 2.0 mm (this assumes the fields are defined in terms of object or image heights), then each pixel represents a 66.67 micron region. If the same image file is later used with a system with a 40 degree full field of view, then the field size can be set to 40 to cover the entire field. Each pixel will now represent a 1.33 degree square. By separating the form of the object from the scale, the same image file can be used for many applications. For example, the sample image file "letterf.ima" contains a 7x7 grid of pixels defining the capital letter F. The object scale can be set to 1 mm, then .1 mm, then .01 mm to get a feel for how small a character F the optical system can resolve, without the need to change the IMA file.

Note that if fields are defined by image height, then field size determines the size of the object in image space, not object space. The field size is always in whatever units the fields are defined in, and so for image height the field size determines image height. The size of the object is then determined by the field size divided by the magnification of the lens.

The choice of field position also permits great flexibility in analyzing image quality. For example, the letter F image file can be tested at several field points to see if the resolution is strongly affected by field aberrations. The object scale is set to the height of the letter, but the image will be centered about the chief ray intercept of the selected field point.

The source is by default a uniform radiator of rays. Uniform here means uniform in the entrance pupil. All rays generated fall within the entrance pupil, and they are all weighted equally. Since ray wavelengths are selected randomly in proportion to the wavelength weights, no explicit wavelength weighting is required. The uniform setting is usually preferred for large object distance systems with small fields of view. The source may also be defined to be Lambertian, which weights all rays by a cosine factor.

By default, all rays which remain unvignetted are displayed. The define detect option allows the numerical aperture to be set, which is used to exclude all rays with a greater numerical aperture than the specified value. This feature can be used to estimate the efficiency of rays coupling into an optical fiber, for example. The size and number of pixels on the detector may also be defined. These values are only used by the "Surface Plot" and "3D Histogram" options. These features simply count the number of rays which land in each pixel, and presents the resulting image as a 3D intensity plot. The surface plot and 3D histograms may require considerable memory and time to compute and display.

A common application of this feature is the calculation of coupling efficiency into a fiber optic. The percent efficiency is defined by

$$\%E = \frac{\sum W_i}{\sum W_j}$$

where the sum i is over all rays which were unvignetted, and the sum over j is over all rays which were launched. The efficiency calculation does not include consideration of reflection, transmission, or scattering losses anywhere in the optical system. It does consider vignetting, the distribution of the source, wavelength weights, and the detector numerical aperture. To restrict the acceptance diameter of the fiber as well as the numerical aperture, place a circular aperture at the image surface with the appropriate maximum radial aperture.

Another common application of this feature is to select a grid shaped object (such as the GRID.IMA sample file) and then use the resulting image to visualize distortion. This works in systems where object height is the selected field type because distortion refers to the deviation from constant magnification over a flat object plane. However, if angles are used to define the field points, then the image analysis feature will generate correct but possibly misleading results. The reason is that the image analysis feature will divide the extended source image file into areas of equal angle, not height. For example, a 10 pixel wide image file used when the field width is 10 mm will have 1 pixel per square mm. The same image file used when the field width is 10 degrees will have 1 pixel per square degree. The "object" shape in these two cases are very different.

See the chapter "ZEMAX Support Files" for a list of sample IMA files and some suggested applications.

When viewing the surface plot or 3D histogram options, pressing the left, right, up, down, Page Up, or Page Down keys will rotate the displayed image for a different perspective.

Selecting the "Text" option on the image analysis window menu bar will generate and display an ASCII file listing the ray data. If the "Show" option is set to "Image Diagram", the file will have 9 columns. The first column is the sequential ray number. The second and third columns are the x and y field coordinates (either degrees or object height). The fourth and fifth columns are the normalized pupil coordinates, Px and Py. The sixth column is the integer wavelength number. The seventh column is the weight of the ray, which depends upon the source properties. The eighth and ninth columns are the image coordinates in lens units, relative to the reference ray.

If the "Show" option is set to either "Surface Plot" or "3D Histogram", then the text display will list the weighted ray count in each pixel. Use the "Escape" key to terminate a length image analysis computation.

### **Vignetting Plot**

*Purpose:*

Calculates fractional vignetting as a function of field angle.

*Settings:*

Item	Description
Ray Density	The ray density specifies the number of rays to be traced. The more rays traced, the greater the accuracy, although the computation time increases. For a ray density of n, ZEMAX traces a grid of $(2n+1) \times (2n + 1)$ rays at each field point.
Field Density	The field density is the number of points between zero degrees and the maximum field angle specified at which vignetting is calculated, intermediate values are interpolated. A maximum of 100 field points is allowed.

*Discussion:*

Fractional vignetting is the percentage of rays incident upon the entrance pupil which pass all obscurations and apertures in the system and survive to the image plane, normalized to relative pupil area (if there are vignetting factors). The graphic generated by this function shows fractional vignetting as a function of field position. If too few rays are used, the results may be inaccurate. This is especially true in systems with many apertures and large field angles.

Only the primary wavelength is used in this calculation. This is a geometric calculation. Only positive y field positions are used, therefore this feature is only appropriate for rotationally symmetric lenses and fields. Vignetting factors are considered in this calculation in determining the relative pupil area. Rays which cause errors such as missing a surface or those which are TIR are considered vignetted.

See also the relative illumination feature.

### **Illumination XY scan**

*Purpose:*

Computes relative illumination for an extended source along a line across the image surface.

*Settings:*

Item	Description
Sampling	Selects the grid size used to sum the illumination on the image surface. The sampling determines how many pixels are used to collect the ray data.
Rays x 1000	Determines the approximate total number of rays to be traced in determining the illumination due to the extended source.
IMA File	The name of the IMA file used to define the shape of the extended source. See the Image Analysis feature for details.
Src. Size	The source size in field units of the extended source.
Rotation	The angle to rotate the extended source in object space about the normal to the center of the extended source.
Show As	Select either X or Y direction scan.

Item	Description
Smoothing	Smoothing helps remove jaggie lines due to low ray sampling by applying a smoothing operator to average the data in adjacent pixels.
Wavelength	The wavelength number to be used in the calculation.
Field	The field number indicates which field position is used as the center reference point for the extended source.
Surface	The scan may be computed at any surface, however, the relative illumination computation is only accurate at the image surface.
Det. Size	The total width of the detector in lens units. The detector size is divided into pixels according to the "Sampling" setting.
Use Relative Illumination	If checked, then the RI computation described in the feature "Relative illumination" is used to weight the rays from various points in the field of view to accurately account for the effects of exit pupil radiance and solid angle. The computation is generally more accurate, but slower, if this feature is used.
Use Polarization	If checked, then polarization effects such as surface and bulk transmission are accounted for in the illumination computation. ZEMAX-EE Only.

*Discussion:*

The illumination XY scan is similar to the relative illumination (RI) feature, with the added capability to estimate the RI for non-uniform extended sources. For uniform extended Lambertian sources, the RI feature is faster and more accurate. However, for systems with complex source properties, the illumination XY scan can estimate illumination by Monte Carlo ray tracing combined with the conventional RI computation.

The extended sources are defined in the same way as described in the Image Analysis feature.

**Illumination 2D surface**

*Purpose:*

Computes relative illumination for an extended source over a 2D surface.

*Settings:*

The options are identical to the XY scan, except that a 2D surface is drawn in either an isometric surface or contour plot, or as a grey scale or false color map.

*Discussion:*

See the illumination XY scan feature.

**Miscellaneous**

**Field Curvature/Distortion**

*Purpose:*

Displays the field curvature and distortion plots.

*Settings:*

Item	Description
Max Curvature	Defines the maximum scale for the field curvature plot in lens units. Enter zero for automatic.
Max Distortion	Defines the maximum scale for the distortion plot in percent. Enter zero for automatic.
Wavelength	The wavelength number to be used in the calculation.
Use Dashes	Selects either color (for color screens and plotters) or dashes (for monochrome screens and plotters).
Ignore Vignetting Factors	See the discussion section.



Item	Description
Distortion	Select Standard, F-Theta, or Calibrated. See the discussion section for details.
Do X-Scan	If checked, then the data is computed based upon a scan along the positive X field, otherwise, the scan is along the positive Y field.

*Discussion:*

The field curvature plot shows the distance from the currently defined focal (image) plane to the paraxial focal plane as a function of field coordinate. The tangential data are the distances measured along the Z-axis from the currently defined focal plane to the paraxial focal plane measured in the tangential (YZ) plane. The sagittal data are the distances measured in the plane orthogonal to the tangential plane. The base of the plot is on axis, and the top of the plot represents the maximum field (angle or height). There are no units on the vertical scale because the plot is always normalized to the maximum radial field.

The field curvature for the tangential and sagittal rays is defined as the distance from the defined image plane to the paraxial focus for that ray. In non-rotational systems, the real ray and the chief ray may never intersect, and so the value presented is at the point of closest approach.

The field scan is along the positive Y field by default. If "Do X-Scan" is selected, then the maximum field is along the positive X field, in which case the tangential curve is for the XZ plane, and the sagittal curve is in the YZ plane.

Novice users often ask why the field curvature plot does not always start at zero for zero field. The reason is that the plot shows the distance from the currently defined image plane to the paraxial focal plane, and the currently defined image plane need not be coincident with the paraxial focal plane. If there is any defocus, then the two planes are offset, and so is the field curvature data.

"Standard" distortion is defined as the real chief ray height, minus the paraxial chief ray height, divided by the paraxial chief ray height, times 100. All heights are taken to be the image plane radial coordinate, at whatever image plane is defined (the data is not referred to the paraxial image plane). The paraxial ray height is computed by tracing a real ray from a very small field height, and then scaling the results as required. This generalization permits the computation of reasonable distortion values even for systems not well described by paraxial ray tracing.

"F-Theta" distortion does not use the paraxial chief ray height, but instead uses the height given by the focal length of the lens multiplied by the angle the chief ray makes in object space. This so called "F-Theta" height is only meaningful in systems with the object at infinity, when field heights are measured in angles. Generally, F-Theta is used only in scanning systems where the image height needs to be linear with scan angle.

"Calibrated" distortion is similar to "F-Theta" distortion, except that the "best-fit" focal length is used rather than the system focal length. Calibrated distortion measures the deviation from linearity between the image height and the field angle, without the restriction that the linearity be defined by the F-Theta condition. A focal length is chosen that best fits the data, rather than the system focal length, although in general the best fit focal length is close to the system focal length. The calibrated focal length used is given on the text listing for this feature.

For non-rotationally symmetric systems, or systems with curved image planes, distortion is poorly defined and the data presented is probably meaningless. The reason is that no single number adequately describes distortion at a single field point if the system is not rotationally symmetric. Instead, use the "grid distortion" plot.

Strictly speaking, the field curvature and distortion plots are only valid for rotationally symmetric systems with plane image surfaces. However, ZEMAX uses a generalization of the field curvature and distortion concepts to give reasonable results for some, but not all, non-rotationally symmetric systems. Caution should be used when interpreting these graphs for non-rotationally symmetric systems.

By default, ZEMAX ignores vignetting factors when computing the field curvature and distortion plots. Vignetting factors can change the chief ray location on the stop surface, such that the chief ray no longer goes through the center of the stop.

**Grid Distortion**

*Purpose:*

Displays a grid of chief ray intercept points to indicate distortion.

*Settings:*

Item	Description
Display	Select either "Cross" to mark each chief ray intercept with a cross, or "Vector" to plot a vector from the ideal image point to the actual chief ray image point.
Grid Size	The size of the grid.
Wavelength	The wavelength number to be used in the calculation.
Field	The reference field position. See the "Discussion".
Scale	If the scale is different than 1.0, then the "x" points on the distortion grid will be exaggerated by the selected scale factor.
Aspect	If unity, then a square field will be selected. The output image may not be square if the system is not symmetric, but the object field will be square. If the aspect is greater than 1, then the "Y" field will be compressed by the aspect ratio given. If the aspect is less than 1, then the "X" field will be compressed by the reciprocal of the aspect ratio. The aspect ratio is therefore the "X" field size divided by the "Y" field size. The aspect ratio only affects the input field; the image aspect ratio is determined by the optical system imaging properties.
Symmetric Magnification	If checked, then the X magnification is required to be identical to the Y magnification. This causes distortion to be referenced to a symmetric predicted grid rather than an anamorphic predicted grid.

*Discussion:*

This feature displays or computes the coordinates of a grid of chief rays. In a system without distortion, the chief ray coordinates on the image plane follow a linear relationship with the field coordinate:

$$\begin{bmatrix} x_p \\ y_p \end{bmatrix} = \begin{bmatrix} AB \\ CD \end{bmatrix} \begin{bmatrix} f_x \\ f_y \end{bmatrix},$$

where  $x_p$  and  $y_p$  are the predicted image coordinates relative to a reference image point and  $f_x$  and  $f_y$  are the linear coordinates on the object surface relative to a reference point. For optical systems using angles as a field definition,  $f_x$  and  $f_y$  are the tangents of the field angles (the field coordinates must be linear, therefore, tangents of angles rather than angles are used). To compute the ABCD matrix, ZEMAX traces rays over a very small region centered upon the reference field position. Usually, this is the center of the field of view. ZEMAX allows selection of which field position to use for reference.

ZEMAX sets the corner of the field grid in object space to be at the maximum radial field distance. Because the object height is linear with field angle and not angle directly, the full width of the field when angles are used to define the field is given by

$$\theta_{wide} = 2 \tan^{-1} \left[ \frac{\sqrt{2}}{2} \tan \theta_r \right],$$

where  $\theta_r$  is the maximum radial field angle at the corner of the field.

The ray coordinates in image space for the very small field of view are used to determine the ABCD matrix components. The use of an ABCD matrix allows for coordinate rotations. If the image plane is rotated, such that a y object coordinate images to both an x and a y image coordinate, the ABCD matrix will automatically account for the rotation. The grid distortion plot shows the linear grid, and then marks the actual chief ray intercept for a ray with the same linear field coordinates with an "X" for each point on the grid.

The text listing available tabulates the predicted image coordinate, the actual image coordinate, and the "percent distortion" defined by

$$P = 100\% \frac{R_{real} - R_{predicted}}{R_{predicted}},$$

where R is the radial coordinate on the image plane relative to the reference field position image location. This definition may not be applicable in all cases, and the results should be used with caution.

## **Longitudinal Aberration**

### *Purpose:*

Displays the longitudinal aberration as a function of pupil height at each wavelength.

### *Settings:*

Item	Description
Scale	Defines the maximum scale for the plot in lens units. Enter zero for automatic.
Wavelength	The wavelength number to be used in the calculation.
Use Dashes	Selects either color (for color screens and plotters) or dashes (for monochrome screens and plotters).

### *Discussion:*

This feature computes the distance from the image surface to where a zonal marginal ray "focuses", or crosses the optical axis. The computation is performed only for the on axis field point, and only for zonal marginal tangential rays as a function of pupil zone. The base of the plot is on axis, and the top of the plot represents the maximum entrance pupil radius. There are no units on the vertical scale because the plot is always normalized to the maximum entrance pupil radius. The horizontal scale is in lens units, and represents the distance from the image surface to the point where the ray crosses the optical axis.

Because longitudinal aberration is defined in terms of the distance to the ray-axis crossing point, this feature may produce meaningless data for non-rotationally symmetric systems. Great care should be exercised in interpreting this plot for non-rotationally symmetric systems.

## **Lateral Color**

### *Purpose:*

Displays the lateral color as a function of field height.

### *Settings:*

Item	Description
Scale	Defines the maximum scale for the plot in lens units. Enter zero for automatic.
Use Real Rays	Either real or paraxial rays may be used. The default is paraxial rays. To use real rays, check this box.

### *Discussion:*

This feature computes the lateral color, which is the distance on the image surface from the shortest wavelength chief ray intercept to the longest wavelength chief ray intercept. The base of the plot is on axis, and the top of the plot represents the maximum field radius. Only positive field angles or heights in the Y direction are used. The vertical scale is always normalized to the maximum field angle or height. The horizontal scale is in lens units. Either real or paraxial rays may be used. This feature may produce meaningless data for non-rotationally symmetric systems. Great care should be exercised in interpreting this plot for non-rotationally symmetric systems.

## **Y-Ybar Diagram**

### *Purpose:*

Y-Ybar diagram.

### *Settings:*

Item	Description
Start Surface	The first surface for which data will be plotted.
Stop Surface	The last surface for which data will be plotted.
Wavelength	The wavelength number to be used in the calculation.

Item	Description
Max Scale	Sets the maximum scale for the plot. The plot is always shown in a square box; the default scale is the maximum transverse ray coordinate. Enter zero for automatic.

*Discussion:*

The Y-Ybar diagram is a plot of marginal ray height as a function of chief ray height for a paraxial skew ray at every surface in the lens.

**Chromatic Focal Shift**

*Purpose:*

Chromatic focal shift plot.

*Settings:*

Item	Description
Maximum Shift	The maximum extent in lens units for the horizontal axis. The vertical axis scale is set by the range of wavelengths defined. Enter zero for automatic.
Pupil Zone	The radial zone in the pupil used to compute the back focus. The default value of zero means a paraxial ray will be used. Values between 0 and 1 mean real marginal rays are used in the appropriate zone in the entrance pupil. A 1 is at the edge of the pupil, or full aperture.

*Discussion:*

This is a plot of the shift in back focal length with respect to the primary wavelength. At each plotted wavelength, the shift in the image plane required to reach paraxial focus for that color marginal ray is computed. This plot may not be meaningful for non-paraxial systems.

The maximum shift setting overrides the default scaling. Units are in lens units. The entire plot is always referenced to the primary wavelength paraxial focus. The diffraction limited depth of focus listed is given by the formula  $4\lambda F^2$ , where F is the working F/# and  $\lambda$  is the primary wavelength.

**Dispersion Diagram**

*Purpose:*

Plots index of refraction as a function of wavelength for any material in the glass catalog.

*Settings:*

Item	Description
Min Wavelength	Defines the left X- axis of the dispersion plot.
Max Wavelength	Defines the right X- axis of the dispersion plot.
Minimum Index	Defines the bottom Y- axis of the dispersion plot. Enter zero for automatic scaling.
Maximum Index	Defines the top Y- axis of the dispersion plot. Enter zero for automatic scaling.
Glass	The name of the material.
Use Temp, Pres	If checked, then the change in index due to temperature and pressure effects will be considered.

*Discussion:*

This is useful for checking if constants of dispersion or other formula data were entered correctly.

**Internal Transmittance vs. Wavelength**

*Purpose:*

Plots the internal transmittance for any thickness as a function of wavelength for any material in the glass catalog.

*Settings:*

Item	Description
Min Wavelength	Defines the left X- axis of the plot.
Max Wavelength	Defines the right X- axis of the plot.
Minimum Transmission	Defines the bottom Y- axis of the plot.
Maximum Transmission	Defines the top Y- axis of the plot. Enter zero for automatic scaling.
Glass	The name of the material.
Thickness	The thickness of the glass in millimeters.

*Discussion:*

This is useful for checking the transmittance of a particular glass. See the Chapter "Polarization Analysis".

**Glass Map**

*Purpose:*

Plots names of glasses on the glass map according to the index of refraction and Abbe V-number. The index and Abbe number are taken directly from the entries in the glass catalog, and are not computed based upon the wavelength data or dispersion coefficients. All currently loaded glass catalogs are searched for glasses within the boundary values specified in the following table.

*Settings:*

Item	Description
Min Abbe	Defines the left X- axis of the plot.
Max Abbe	Defines the right X- axis of the plot.
Min Index	Defines the bottom Y- axis of the plot. Enter zero for automatic scaling.
Max Index	Defines the top Y- axis of the plot. Enter zero for automatic scaling.

*Discussion:*

This is useful for locating a glass with particular refractive and dispersive properties. By convention, the glass map is shown with the Abbe number decreasing while going from left to right, which is why the default min and max Abbe numbers seem to be reversed.

**Test Plate Lists**

*Purpose:*

Displays in a text window a list of test plates from a specific vendor.

*Settings:*

Item	Description
File Name	The name of the test plate file. The file must be in the default directory for ZEMAX.

---

**Calculations**

**Ray Trace**

*Purpose:*

Paraxial and real trace of a single ray.

### Settings:

Item	Description
Hx	Normalized x-field coordinate. The value should be between -1 and 1.
Hy	Normalized y-field coordinate. The value should be between -1 and 1.
Px	Normalized x-pupil coordinate. The value should be between -1 and 1.
Py	Normalized y-pupil coordinate. The value should be between -1 and 1.
Type	Select "Dir Cosines" to display the ray direction cosines at each surface, "Tangent Ang" to display the tangent of the angle the ray makes at each surface, or "Ym,Um,Yc,Uc" to display the paraxial marginal and chief ray intercept/tangent angle display. The tangent angle is simply the ratio of the x (or y) direction cosine to the z direction cosine.
Wavelength	The wavelength number of the ray to trace.

### Discussion:

The Hx, Hy, Px, Py data is ignored if the "Ym, Um, Yc, Uc" option is selected. For a description of normalized coordinates, see the chapter "Conventions and Definitions".

For the other options, this feature allows the user to specify the normalized object coordinates, the normalized pupil coordinates, and wavelength number and then view the real and paraxial ray coordinates at every surface.

The first set of data presented is for a real ray. The values presented are the coordinates (in the surface local coordinate system) of the ray intercept point. The direction cosines (or tangent angles) are the values of the ray in the surface local coordinate system after refraction. The direction cosine value is the cosine of the angle the ray makes with respect to the specified axis (the x-direction cosine is the cosine of the angle the ray makes with respect to the local x-axis, etc.) The second set of data is just like the first set, except it is for a paraxial ray.

## **Gaussian Beam**

### Purpose:

Computes Gaussian beam parameters.

### Settings:

Item	Description
Wavelength	The wavelength number to use for the calculation.
M Factor	The M quality factor used to simulate mixed mode beams. See the Discussion.

### Discussion:

ZEMAX assumes that the entrance pupil diameter is twice the radial beam size at the entrance pupil position. The second assumption ZEMAX makes is that the beam phase radius of curvature at the entrance pupil is equal to the object distance. These two conditions (along with the wavelength) uniquely determine the object space beam waist size and location.

It is important to understand that ZEMAX does not use the object height to determine the object space beam waist, and that ZEMAX does not necessarily place the beam waist at the object plane. For example, if the object distance is infinite, the phase radius of curvature at the entrance pupil is infinite, and therefore the waist is located at the entrance pupil. To initiate the calculation, ZEMAX computes the object space beam waist using the formula

$$\omega_0 = \omega \left[ 1 + \left( \frac{\pi \omega^2}{\lambda R} \right)^2 \right]^{-1/2},$$

where R is the distance from the object surface to the entrance pupil and  $\omega$  is one-half the entrance pupil diameter. The distance from the waist to the object surface (note this is not always zero) can then be computed using

$$z = R - R \left[ 1 + \left[ \frac{\lambda R}{\pi \omega^2} \right]^2 \right]^{-1} .$$

Once the initial parameters are established, ZEMAX traces the Gaussian beam through the system and computes the radial beam size, the narrowest radial waist, the surface coordinate relative to the beam waist, the semi divergence angle, and the Rayleigh range for every surface in the system. ZEMAX calls these parameters the Size, Waist, Waist Z, Divergence, and Rayleigh, respectively, on the text listing that is generated.

The dimensions for all parameters are lens units except for the semi divergence angle, which is in units of radians. The Gaussian beam parameters can be computed using the following standard formulas. The Rayleigh range is defined as

$$z_r = \frac{\pi \omega_0^2}{\lambda} .$$

The phase radius of curvature is given by

$$R(z) = z + \frac{z_r^2}{z} ,$$

where  $z_r$  is the Rayleigh range and  $z$  is the distance from the beam waist. The radial beam size at any  $z$  is computed from

$$\omega(z) = \omega_0 \left[ 1 + \left( \frac{z}{z_r} \right)^2 \right]^{1/2} ,$$

where  $\omega_0$  is the beam waist. The divergence angle of the beam is given by

$$\theta = \tan^{-1} \frac{\omega_0}{z_r} .$$

All of the preceding results are correct for an ideal TEM00 fundamental mode Gaussian beam. For mixed-mode beams, a simple extension to the fundamental Gaussian beam model has been developed by Siegman. The method uses a term called the beam quality factor, usually denoted by  $M$ . The factor  $M$  can be thought of as "times diffraction limited" number, and is always greater than unity. The  $M$  factor scales what is called the "embedded" Gaussian mode, which is a TEM00 mode.

If the  $M$  factor is set to unity, the default value, ZEMAX simply computes the TEM00 data described above. If  $M$  is greater than unity, then ZEMAX computes both the TEM00 mode data and the embedded Gaussian beam parameters as well as the scaled mixed mode data.

The "quality factor"  $M$ , sometimes given as  $M^2$ , is a measure of the ratio of the beam size to the size of the "embedded" mode. The embedded Gaussian mode is computed by ZEMAX just like the TEM00 mode, except the beam size on the entrance pupil is assumed to be given by  $W' = W/M$ . For  $M = 2$ , the embedded TEM00 mode has a beam size half the pupil radius. The actual, mixed mode beam has a size which is given by  $W = M w$ , where  $w$  is the embedded Gaussian beam size.

The mixed mode waist, beam size, and divergence angle are related to the embedded Gaussian mode quantities by a simple factor of  $M$ .

ZEMAX computes the Gaussian beam parameters for both X and Y orientations.

For more information on Gaussian beam propagation, see "Lasers", A. E. Siegman, University Science Books (1986), and "Gaussian beam ray-equivalent modeling and optical design", R. Herloski, S. Marshall, and R. Antos, Applied Optics Vol. 22, No. 8 pp. 1168 (1983). For information on the  $M$  factor, see A. E. Siegman, "New developments in laser resonators", Proc. SPIE Vol. 1224.

Because the Gaussian beam parameters are based upon paraxial ray data the results cannot be trusted for systems which have large aberrations, or those poorly described by paraxial optics, such as non-rotationally symmetric systems. This feature ignores all apertures.

## Seidel Coefficients

### *Purpose:*

Displays Seidel (unconverted, transverse, and longitudinal), and Wavefront aberration coefficients.

### *Settings:*

Item	Description
Wavelength	The wavelength number to use for the calculation.

### *Discussion:*

ZEMAX will compute the unconverted Seidel, transverse, longitudinal, and some wavefront coefficients. The Seidel coefficients are listed surface by surface, as well as a sum for the entire system. The coefficients listed are for spherical aberration (SPHA, S1), coma (COMA, S2), astigmatism (ASTI, S3), field curvature (FCUR, S4), distortion (DIST, S5), longitudinal color (CLA, CL), and transverse color (CTR, CT). The units are always the same as the system lens units, except of course for the coefficients measured in waves.

These calculations are only valid and accurate for systems consisting entirely of standard surfaces. Any systems which contain coordinate breaks, gratings, paraxial, or other non-standard surfaces are not adequately described by the paraxial rays which are used to compute the coefficients.

The "Seidel/Wavefront" calculations are only valid and accurate for systems consisting entirely of standard surfaces.

Transverse aberration coefficients are also listed by surface with a system sum. The coefficients given are transverse spherical (TSPH), transverse sagittal coma (TSCO), transverse tangential coma (TTCO), transverse sagittal field curvature (TSFC), transverse tangential field curvature (TTFC), transverse distortion (TDIS), and transverse lateral color (TLAC). The transverse aberrations are in the system lens units. The transverse aberration coefficients may be very large in optical spaces where the light is nearly collimated, and have little meaning in these optical spaces.

Longitudinal aberration coefficients are computed for longitudinal spherical aberration (LSPH), longitudinal astigmatism (LAST), longitudinal field curvature Petzval (LFCP), longitudinal field curvature sagittal (LFCS), longitudinal field curvature tangential (LFCT), and longitudinal axial color (LAXC). The longitudinal aberrations are in the system lens units. The longitudinal aberration coefficients may be very large in optical spaces where the light is nearly collimated, and have little meaning in these optical spaces.

The wavefront coefficients given are spherical aberration (W040), coma (W131), astigmatism (W222), field curvature Petzval (W220P), distortion (W311), axial color defocus term (W020), lateral color tilt term (W111), field curvature sagittal (W220S), field curvature medial (W220M), and field curvature tangential (W220T). All the wavefront coefficients are in units of wavelengths at the edge of the exit pupil. The various aberration coefficients are interrelated according to the following table. The symbols  $n$  and  $u$  refer to the index of refraction and the paraxial marginal ray angle in the object space side of each surface. The primes above the  $n$  and  $u$  symbols indicate that these are the values on the image space side of the surface. For a discussion of the meaning and derivation of the Seidel aberration coefficients, see Welford, *Aberrations of Optical Systems*, Smith, *Modern Lens Design*, or O' Shea, *Elements of Modern Optical Design*. Complete references can be found in the chapter "Introduction".

### INTERRELATIONSHIP OF ABERRATION COEFFICIENTS

Name	Seidel	Wave	Description	Transverse	Longitudinal
Spherical	$S_1$	$\frac{S_1}{8}$	-	$-\frac{S_1}{2n'u}$	$\frac{S_1}{2n'u^2}$



Name	Seidel	Wave	Description	Transverse	Longitudinal
Coma	$S_2$	$\frac{S_2}{2}$	Sagittal	$\frac{-S_2}{2n'u}$	-
			Tangential	$\frac{-3S_2}{2n'u}$	-
Astigmatism	$S_3$	$\frac{S_3}{2}$	From tangential to sagittal foci	-	$\frac{S_3}{n'u^2}$
Field Curvature	$S_4$	$\frac{S_4}{4}$	Gaussian to Petzval	-	$\frac{S_4}{2n'u^2}$
			Gaussian to sagittal	$\frac{-(S_3 + S_4)}{2n'u}$	$\frac{S_3 + S_4}{2n'u^2}$
			Gaussian to tangential	$\frac{-(3S_3 + S_4)}{2n'u}$	$\frac{3S_3 + S_4}{2n'u^2}$
Distortion	$S_5$	$\frac{S_5}{2}$	-	$\frac{-S_5}{2n'u}$	-
Axial Color	$C_L$	$\frac{C_L}{2}$	-	-	$\frac{C_L}{n'u^2}$
Lateral Color	$C_T$	$C_T$	-	$\frac{C_T}{n'u}$	-

The Petzval radius of curvature shown on the Seidel screen is in system lens units, as is the Lagrange invariant.

### **Zernike Coefficients**

*Purpose:*

Displays Zernike coefficients.

*Settings:*

Item	Description
Sampling	Specify the density in the pupil to use for coefficient fitting. Larger grid sizes are more accurate, although the computation time increases.
Max Term	Specify the maximum Zernike coefficient to compute. Any value up to 36 may be specified.
Wavelength	The wavelength number to use for the calculation.
Field	The field number to use for the calculation.

*Discussion:*

The Zernike screen displays the individual coefficients as well as the peak-to-valley, RMS, variance, Strehl ratio, residual RMS fit error, and maximum fit error.

The RMS of the wavefront error,  $\sigma$ , when referenced to the mean, is defined as

$$\sigma^2 = \overline{W^2} - \overline{W}^2$$

where  $W$  is the wavefront error,  $\overline{W^2}$  is the mean square error, and  $\overline{W}$  is the average wavefront error. The RMS can actually be computed several different ways. If the mean wavefront term is ignored, then the RMS "referenced to zero" results. This computation is rarely used. If the mean wavefront term is used, then the RMS is "referenced to the mean". Typically, the RMS is further referenced to the diffraction image centroid, which is equivalent to

subtracting out not only the mean (which is piston) but the average tilt in x and y as well. This is justified because tilt simply shifts the location of the diffraction image centroid, but otherwise has no effect on image quality. Most of the time, the RMS is taken to mean the RMS referenced to the diffraction image centroid, which is always the lowest of the three numbers.

The Strehl ratio is computed using the RMS referenced to the centroid by the following approximation:

$$S = e^{-(2\pi\sigma)^2}$$

The approximation is only valid for Strehl ratios higher than about 0.10. The Zernike polynomials are enumerated according to the following table.

### ENUMERATION OF ZERNIKE POLYNOMIALS

Term	Z( $\rho, \phi$ )
0	1
1	$\rho \cos \phi$
2	$\rho \sin \phi$
3	$2\rho^2 - 1$
4	$\rho^2 \cos 2\phi$
5	$\rho^2 \sin 2\phi$
6	$(3\rho^2 - 2)\rho \cos \phi$
7	$(3\rho^2 - 2)\rho \sin \phi$
8	$6\rho^4 - 6\rho^2 + 1$
9	$\rho^3 \cos 3\phi$
10	$\rho^3 \sin 3\phi$
11	$(4\rho^2 - 3)\rho^2 \cos 2\phi$
12	$(4\rho^2 - 3)\rho^2 \sin 2\phi$
13	$(10\rho^4 - 12\rho^2 + 3)\rho \cos \phi$
14	$(10\rho^4 - 12\rho^2 + 3)\rho \sin \phi$
15	$20\rho^6 - 30\rho^4 + 12\rho^2 - 1$
16	$\rho^4 \cos 4\phi$
17	$\rho^4 \sin 4\phi$
18	$(5\rho^2 - 4)\rho^3 \cos 3\phi$
19	$(5\rho^2 - 4)\rho^3 \sin 3\phi$
20	$(15\rho^4 - 20\rho^2 + 6)\rho^2 \cos 2\phi$
21	$(15\rho^4 - 20\rho^2 + 6)\rho^2 \sin 2\phi$
22	$(35\rho^6 - 60\rho^4 + 30\rho^2 - 4)\rho \cos \phi$

Term	Z( $\rho, \varphi$ )
23	$(35\rho^6 - 60\rho^4 + 30\rho^2 - 4)\rho \sin\varphi$
24	$70\rho^8 - 140\rho^6 + 90\rho^4 - 20\rho^2 + 1$
25	$\rho^5 \cos 5\varphi$
26	$\rho^5 \sin 5\varphi$
27	$(6\rho^2 - 5)\rho^4 \cos 4\varphi$
28	$(6\rho^2 - 5)\rho^4 \sin 4\varphi$
29	$(21\rho^4 - 30\rho^2 + 10)\rho^3 \cos 3\varphi$
30	$(21\rho^4 - 30\rho^2 + 10)\rho^3 \sin 3\varphi$
31	$(56\rho^6 - 105\rho^4 + 60\rho^2 - 10)\rho^2 \cos 2\varphi$
32	$(56\rho^6 - 105\rho^4 + 60\rho^2 - 10)\rho^2 \sin 2\varphi$
33	$(126\rho^8 - 280\rho^6 + 210\rho^4 - 60\rho^2 + 5)\rho \cos\varphi$
34	$(126\rho^8 - 280\rho^6 + 210\rho^4 - 60\rho^2 + 5)\rho \sin\varphi$
35	$252\rho^{10} - 630\rho^8 + 560\rho^6 - 210\rho^4 + 30\rho^2 - 1$
36	$924\rho^{12} - 2772\rho^{10} + 3150\rho^8 - 1680\rho^6 + 420\rho^4 - 42\rho^2 + 1$

### **YNI Contributions**

#### *Purpose:*

This feature lists for each surface the paraxial YNI value, which is proportional to the Narcissus contribution of that surface.

#### *Settings:*

Item	Description
Wavelength	The wavelength number to use for the calculation.

#### *Discussion:*

See Applied Optics, Vol. 21, #18, p3393 for a discussion.

### **Sag Table**

#### *Purpose:*

This feature lists for the selected surface the surface sag (z-coordinate) at various distances from the vertex. The best-fit spherical radius is computed, and the sag of the best-fit sphere and the difference is also tabulated. Only the Y-coordinate of the surface is considered, therefore the data may not be useful for non-rotationally symmetric surfaces.

#### *Settings:*

Item	Description
Surface	The surface number to use for the calculation.
Step Size	The distance between steps measured from the vertex at which the sag is computed. The default value of zero will automatically choose a reasonable step size.

Item	Description
Negative Remove	See "Discussion".

*Discussion:*

This feature can be used to determine the maximum aspheric deviation of a surface. The sag table is also useful for lens fabrication.

This feature lists out five columns:

Y-Coord: The y coordinate of the point on the surface being computed.

Sag: The sag of the surface at the y coordinate.

BFS Sag: The sag of the best fit sphere.

Deviation: The difference between the BFS Sag and the Sag, i.e. (BFS Sag) - (Sag).

Remove: The amount of material to remove assuming that the surface was first generated to the best fit sphere. By default, this number is given to be positive. Checking the "Negative Remove" box will yield all negative values for the remove column. Note that the remove column values are identical to the deviation values, but a constant offset value has been added.

The remove values can be selected to be either all positive or all negative. Which is correct depends upon whether the surface is convex or concave. ZEMAX cannot always make this determination correctly, and so it is left as a user selectable option. The general rules of thumb are:

If the BFS radius is positive, and the surface convex, remove should be positive.

If the BFS radius is positive, and the surface concave, remove should be negative.

If the BFS radius is negative, and the surface convex, remove should be negative.

If the BFS radius is negative, and the surface concave, remove should be positive.

It is always advisable to check the sign convention carefully before any fabrication decisions are made.

## **Gradient Index**

---

### ***Gradium™ Profile***

*Purpose:*

Plots the axial index profile of a Gradium surface.

*Settings:*

Item	Description
Profile	The name of the profile. See the chapter on surface types for details.
Surface	The surface number to plot.
Wavelength	The wavelength number to use for the calculation.

*Discussion:*

If the surface is selected to "None", then the reference wavelength for the profile glass family is used, no matter what wavelength is selected. If a Gradium surface is selected, then any defined wavelength, or the reference wavelength may be selected. Also, if a surface number is selected, then the starting and ending points for the glass blank will be indicated using an "X" on the plot. The starting and ending positions include consideration of the sag of the surface at the defined semi-diameter.

## **Polarization**

---

For more information on polarization calculations, see the chapter "Polarization Analysis". All of the features available in the polarization menu are only available in ZEMAX-EE.

## **Polarization Ray Trace**

### *Purpose:*

The polarization ray trace feature generates a text window which displays all the polarization data for a single ray.

### *Settings:*

Item	Description
Ex	Electric field amplitude in X direction.
Ey	Electric field amplitude in Y direction.
X-Phase	Phase of the X component of the electric field in degrees.
Y-Phase	Phase of the Y component of the electric field in degrees.
Hx	Normalized x-field coordinate. The value should be between -1 and 1.
Hy	Normalized y-field coordinate. The value should be between -1 and 1.
Px	Normalized x-pupil coordinate. The value should be between -1 and 1.
Py	Normalized y-pupil coordinate. The value should be between -1 and 1.
Wavelength	The wavelength number of the ray to trace.

### *Discussion:*

This feature tabulates all of the data computed by ZEMAX to perform polarization ray tracing. See the chapter "Polarization Analysis".

## **Polarization Pupil Map**

### *Purpose:*

Generates a graph of the polarization ellipse as a function of pupil position. This aids in visualizing the change in polarization over the pupil.

### *Settings:*

Item	Description
Ex	Electric field amplitude in X direction.
Ey	Electric field amplitude in Y direction.
X-Phase	Phase of the X component of the electric field in degrees.
Y-Phase	Phase of the Y component of the electric field in degrees.
Wavelength	The wavelength number of the rays to trace.
Field	The field position number of the rays to trace.

### *Discussion:*

The polarization ellipse is a representation of the figure traced out by the electric field vector as the wave propagates during one cycle. The magnitude of the ellipse is determined by the transmission of the ray, which is generally a function of pupil position. See the chapter "Polarization Analysis".

## **Transmission**

### *Purpose:*

Computes the integrated transmission through the optical system considering polarization effects.

### *Settings:*

Item	Description
Ex	Electric field amplitude in X direction.
Ey	Electric field amplitude in Y direction.
X-Phase	Phase of the X component of the electric field in degrees.

Item	Description
Y-Phase	Phase of the Y component of the electric field in degrees.
Unpolarized	If checked, the electric field definition is ignored and the unpolarized computation is performed.

*Discussion:*

This feature tabulates for each field position and wavelength the integrated transmission of the optical system for the specified polarization. The transmission is computed as a fraction of 100%, with 100% being that transmitted if there were no absorption, reflection, or vignetting losses. The transmission calculation accounts for vignetting factors, vignetting due to apertures or obscurations, ray clipping due to ray trace errors, surface Fresnel or coating losses, and bulk internal transmittance due to absorption. See also the relative illumination feature.

**Phase Aberration**

*Purpose:*

Computes the polarization induced phase aberration of an optical system.

*Settings:*

Item	Description
Ex	Electric field amplitude in X direction.
Ey	Electric field amplitude in Y direction.
X-Phase	Phase of the X component of the electric field in degrees.
Y-Phase	Phase of the Y component of the electric field in degrees.
Wavelength	The wavelength number of the rays to trace.
Field	The field position number of the rays to trace.

*Discussion:*

This feature computes for the specified field position and wavelength the phase aberration in the image space X and Y orientations. Polarization phase aberrations are induced by the effects of refraction through dielectric media, and by reflection from metallic or dielectric mirrors.

Like the usual OPD plots, the polarization phase aberration is referenced to the chief ray. However, there are cases where the chief ray phase cannot be determined in both orientations. For example, in an axial symmetric system, if the incident polarization is linear in the Y direction, there is zero intensity in the X direction for the chief ray, and therefore the X phase is indeterminate. For other rays in the pupil, there is generally a slight rotation of the polarization, and therefore the resulting electric field in the X direction yields a valid phase angle. To avoid this discontinuity in the phase, ZEMAX averages two rays on either side of the chief ray to interpolate the chief ray phase. The problem may still appear under certain circumstances, even with this averaging technique. In all cases, the phase data is still valid, because the phase aberration has no effect on image quality if the intensity is zero.

**Transmission Fan**

*Purpose:*

Generates a graph of the transmitted intensity for each field and wavelength as a function of either tangential or sagittal pupil fans.

*Settings:*

Item	Description
Ex	Electric field amplitude in X direction.
Ey	Electric field amplitude in Y direction.
X-Phase	Phase of the X component of the electric field in degrees.
Y-Phase	Phase of the Y component of the electric field in degrees.

Item	Description
Wavelength	The wavelength number of the rays to trace.
Field	The field position number of the rays to trace.
Unpolarized	If checked, the electric field definition is ignored and the unpolarized computation is performed.

*Discussion:*

The polarization transmission fan is useful for determining the transmission variation over the pupil as a function of field and wavelength. See the chapter "Polarization Analysis".

## Coatings

---

For more information on coating calculations, see the chapter "Polarization Analysis". All of the features available in the coatings menu are only available in ZEMAX-EE.

### **Reflection vs. Angle**

*Purpose:*

Computes the S, P, and average polarization intensity coefficients for reflection for the specified surface as a function of incident angle.

*Settings:*

Item	Description
Min Angle	The minimum angle of incidence to plot. This defines the left edge of the plot.
Max Angle	The maximum angle of incidence to plot. This defines the right edge of the plot.
Min Y	The minimum y value to plot. This defines the bottom edge of the plot.
Max Y	The maximum y value to plot. This defines the top edge of the plot.
Surface	The surface number which defines the interface to use.
Wavelength	The wavelength number to use.

*Discussion:*

The incident angle is measured in the medium prior to the specified surface. See the chapter "Polarization Analysis" for details on the computation.

### **Transmission vs. Angle**

*Purpose:*

Computes the S, P and average polarization intensity coefficients for transmission for the specified surface as a function of incident angle.

*Settings:*

See "Reflectivity vs. Angle" above.

*Discussion:*

The incident angle is measured in the medium prior to the specified surface. See the chapter "Polarization Analysis" for details on the computation.

### **Absorption vs. Angle**

*Purpose:*

Computes the S, P, and average polarization intensity coefficients for absorption for the specified surface as a function of incident angle.

*Settings:*

See "Reflectivity vs. Angle" above.

*Discussion:*

The incident angle is measured in the medium prior to the specified surface. See the chapter "Polarization Analysis" for details on the computation.

**Diattenuation vs. Angle**

*Purpose:*

Computes the R (reflected) and T (transmitted) diattenuation for the specified surface as a function of incident angle.

*Settings:*

See "Reflectivity vs. Angle" above.

*Discussion:*

The incident angle is measured in the medium prior to the specified surface. See the chapter "Polarization Analysis" for details on the computation.

**Phase vs. Angle**

*Purpose:*

Computes the S and P polarization phase for reflection (if the surface is a mirror) or for transmission (if the surface is not a mirror) for the specified surface as a function of incident angle.

*Settings:*

See "Reflectivity vs. Angle" above.

*Discussion:*

The incident angle is measured in the medium prior to the specified surface. See the chapter "Polarization Analysis" for details on the computation.

**Retardance vs. Angle**

*Purpose:*

Computes the retardance for the specified surface as a function of incident angle.

*Settings:*

See "Reflectivity vs. Angle" above.

*Discussion:*

The incident angle is measured in the medium prior to the specified surface. See the chapter "Polarization Analysis" for details on the computation.

**Reflection vs. Wavelength**

*Purpose:*

Computes the S, P, and average polarization intensity coefficients for reflection for the specified surface as a function of incident wavelength.

*Settings:*

Item	Description
Min Wave	The minimum wavelength to plot. This defines the left edge of the plot.



Item	Description
Max Wave	The maximum wavelength to plot. This defines the right edge of the plot.
Min Y	The minimum y value to plot. This defines the bottom edge of the plot.
Max Y	The maximum y value to plot. This defines the top edge of the plot.
Surface	The surface number which defines the interface to use.
Angle	The angle of incidence, in degrees, to use.

*Discussion:*

See the chapter "Polarization Analysis".

### **Transmission vs. Wavelength**

*Purpose:*

Computes the S, P, and average polarization intensity coefficients for transmission for the specified surface as a function of incident wavelength.

*Settings:*

See "Reflectivity vs. Wavelength" above.

*Discussion:*

See the chapter "Polarization Analysis".

### **Absorption vs. Wavelength**

*Purpose:*

Computes the S, P, and average polarization intensity coefficients for absorption for the specified surface as a function of incident wavelength.

*Settings:*

See "Reflectivity vs. Wavelength" above.

*Discussion:*

See the chapter "Polarization Analysis".

### **Diattenuation vs. Wavelength**

*Purpose:*

Computes the R (reflected) and T (transmitted) diattenuation for the specified surface as a function of incident wavelength.

*Settings:*

See "Reflectivity vs. Wavelength" above.

*Discussion:*

See the chapter "Polarization Analysis".

### **Phase vs. Wavelength**

*Purpose:*

Computes the S and P polarization phase for reflection (if the surface is a mirror) or for transmission (if the surface is not a mirror) for the specified surface as a function of incident wavelength.

*Settings:*

See "Reflectivity vs. Wavelength" above.

*Discussion:*

See the chapter "Polarization Analysis".

**Retardance vs. Wavelength**

*Purpose:*

Computes the retardance for the specified surface as a function of incident wavelength.

*Settings:*

See "Reflectivity vs. Wavelength" above.

*Discussion:*

See the chapter "Polarization Analysis".

This page intentionally left blank.

## Optimization

---

*Purpose:*

Optimization is used to improve or modify a design to meet specific conditions.

*Discussion:*

See the chapter "Optimization" for a full discussion of building merit functions, setting variables, and using the optimization feature.

## Global Search

---

This feature is only available in the XE and EE editions of ZEMAX.

*Purpose:*

This feature initiates a search for the global optimum, which is the best possible design for a given merit function and set of variables.

*Discussion:*

See the chapter "Global Optimization".

## Hammer Optimization

---

This feature is only available in the XE and EE editions of ZEMAX.

*Purpose:*

This feature automates the repetitive optimization of a design to escape local minima in the merit function.

*Discussion:*

See the chapter "Global Optimization".

## Tolerancing

---

*Purpose:*

Tolerancing.

*Discussion:*

See the chapter "Tolerancing" for a complete discussion.

## Glass Catalogs

---

*Purpose:*

Provides access to the glass catalogs.

*Discussion:*

See the chapter "Using Glass Catalogs" for an extended discussion of this feature.

## Lens Catalogs

---

### *Purpose:*

Used to search for a particular lens from or browse through the stock lens catalogs.

### *Settings:*

Item	Description
Vendor	Lists the available stock lens vendors. Each vendor name is the name of the file which contains the stock lenses available from that vendor. The vendor files must be placed in the default directory for stock lenses, which is set on the environments dialog box.
Use Focal Length	If checked, then the specified focal length range is used as part of the search criteria, otherwise, any value will be accepted.
Focal Length Min/Max	Used to define the range of acceptable focal lengths in millimeters.
Use Diameter	If checked, then the specified entrance pupil diameter range is used as part of the search criteria, otherwise, any value will be accepted.
Diameter Min/Max	Used to define the range of acceptable entrance pupil diameters in millimeters.
Equi-, Bi-, Plano-, Meniscus	If any of these options are checked, then the search will be limited to lenses meeting at least one of the selected criteria. For example, if both Equi- and Bi- are selected, then both equiconvex/equiconcave and biconvex/biconcave lenses will be included.
Spherical, GRIN, Aspheric, Toroidal	If any of these options are checked, then the search will be limited to lenses meeting at least one of the selected criteria. The "Spherical" category is also used as "other" for lenses which are not gradient index, aspheric, or toroidal.
Max elements	Selects the maximum number of elements allowed in lenses that meet the search criteria. Choose "Any #" to include all lenses in the search.
Search Results	Lists all lens files in the specified directory which match the current search criteria.
Search	Causes a new search to be conducted for lenses in the specified vendor file based upon the current search criteria.
Load	Loads the currently selected lens file into the Lens Data Editor, and updates all windows to display data for the newly loaded lens.
Append	Appends the currently selected lens file to the existing lens in the Lens Data Editor. A prompt will be issued for the append file options, such as at which surface the lens should be appended.
Data	Displays a prescription report of the currently selected lens file.
Layout	Displays a layout plot (or 3D Layout for systems without axial symmetry) of the currently selected lens file.
Exit	Closes the dialog box.

### *Discussion:*

The "Search Results" table will list the part name, focal length, and diameter of all the lenses from the selected vendor which meet the specified search criteria. After the diameter, a 3 part code is listed, such as (P,S, 1). The first entry is the shape code, which is E, B, P, M, or "?", for Equi-, Bi-, Plano-, Meniscus, or other. Other is used if the lens is multi-element. The second entry is S, G, A, or T, for Spherical, GRIN, Aspheric, or Toroidal. The third entry is the number of elements.

Upon installation, ZEMAX creates a subdirectory within the ZEMAX directory called STOCKCAT. Within STOCKCAT, many individual files are stored with the extension .ZMF. Each of these ZMF files contains a large number of individual ZMX lens files, each of which contains lens files which represent stock lenses available from various vendors. For example, if the ZEMAX directory name is C:\ZEMAX, then the stock lens files for Melles Griot are stored in the file C:\ZEMAX\STOCKCAT\MGRIOT.ZMF.

The Lens Search tool provides the capability to search through the available lenses to find an appropriate choice. Once selected, the lens file may be loaded or appended to an existing design. See the Chapter "Support Files" for a complete list of the available stock lens vendor catalogs.

## Performance Test

---

*Purpose:*

System ray-trace performance test.

*Discussion:*

Performance test runs a check on the number of ray-surfaces per second your hardware/lens combination is capable of. The number is calculated by tracing a large number of random skew rays through the current optical system, and then dividing the number of surfaces traced by the elapsed time in seconds. The speed will vary tremendously depending upon the system processor, coprocessor, clock speed, and lens complexity.

## Reverse Elements

---

*Purpose:*

Reverses a lens element or group.

*Settings:*

Item	Description
First Surface	This is the first surface of the lens group to be reversed.
Last Surface	The last surface of the lens group to be reversed.

*Discussion:*

The feature may not work correctly if mirrors, coordinate breaks, or other non-standard surfaces are included in the range of surfaces.

## Scale Lens

---

*Purpose:*

Scale will scale the entire lens by the specified factor. This is useful for scaling an existing design to a new focal length, for example. Wavelengths are not scaled. The scale lens feature may also be used to change the units from mm to inches, or other combinations of unit types.

*Settings:*

Item	Description
Scale by factor	If selected, then a scale factor may be entered directly.
Scale by units	If selected, then the lens will be converted by the selected units.

## Make Focal

---

*Purpose:*

Make focal length is similar to scale lens, except the desired focal length is typed in directly. The entire lens is then scaled to make the focal length the specified value.

## Ghost Focus Generator

---

*Purpose:*

Ghost focus analysis.

*Settings:*

Item	Description
Bounces	Select either single-bounce or double-bounce analysis.

Item	Description
First Surface	The first surface to consider reflections from.
Last Surface	The last surface to consider reflections from.
Save Files	If checked, the files used to compute the ghost ray trace are saved to disk.
Image Plane Only	If checked, only data for the image surface will be shown when computing double bounce ghosts.

*Discussion:*

This feature may not work correctly for systems which contain coordinate breaks, non-standard surfaces, or multi-configuration data.

This feature generates lens files which are derived from the current lens prescription data. The generated files are set up to reflect light at a given surface, rather than refract it. The portion of the optical system prior to the new reflective surface is then duplicated so the rays can be traced back through. The purpose of this analysis is to check whether or not rays reflected from any optical surface can form "ghost" images on other components or near the focal plane. These effects are significant in high power laser systems, where focused reflections can damage optics. Ghost images also reduce contrast. Both single and double reflections are supported.

For each ghost system, the marginal ray height, paraxial ray F/#, and RMS spot size on axis are listed. Glass surfaces which may have an internal focus are also indicated. For the image surface, when doing double-bounce ghost analysis, the distance from the image surface to the ghost and the EFL of the ghost system are also provided.

The lens files generated are stored in the file GHfffsss.ZMX, and these lens files can be opened for more analysis. The fff is the number of the first ghost surface, and sss is the number of the second ghost surface. For example GH007002.ZMX is a ghost focus file of the double bounce off surface 7 then 2. For single bounce ghosts, only the fff number is non-zero.

This routine does not work correctly for systems that have non-standard surfaces or coordinate breaks, or if the lens is under multi-configuration control. This latter problem is circumvented by activating the configuration of interest, and then deleting all multi-configuration data before executing this feature. Not all ghost reflection configurations can be traced; there are occasionally problems with total internal reflection or rays completely missing surfaces. The GHfffsss.ZMX file may need to be opened and modified to analyze it in detail.

If the first reflected surface is before the stop, then the entrance pupil position is incorrectly calculated as well. This problem is easily overcome (for the purposes of analyzing the ghosts only) by using the following procedure before you begin the ghost focus generation:

- 1) Record the entrance pupil position and diameter.
- 2) Create a dummy surface at surface one.
- 3) Give this new dummy surface a thickness equal to the negative of the entrance pupil position value recorded.
- 4) Make the dummy surface the stop, and use an entrance pupil diameter equal to the entrance pupil diameter recorded.
- 5) For finite conjugates only, increase the object surface thickness by an amount equal to the thickness of the dummy surface.

These steps will give you a real entrance pupil in object space, and the rays will be correctly traced when reflected off surfaces before the stop. Ghost focus analysis can be very complex for moderately complicated systems, and care is required in interpreting the results of this analysis.

## Test Plate Fitting

This feature is only available in the XE and EE editions of ZEMAX.

*Purpose:*

Provides automatic fitting of radii to a vendor test plate list.

*Settings:*

Item	Description
File Name	Selects different test plate lists

*Discussion:*

This feature automates the task of matching radii of lens elements to the existing tooling of a particular vendor. To match a particular radius, mark it as variable on the lens data editor. As many radii as desired can be matched at one time. Now execute the test plate fitting tool. Select which test plate file to be used (these are described in the chapter "ZEMAX Support Files").

Now press OK and the test plate fitting will begin. ZEMAX starts by searching the test plate list for the closest match (in fringes) between all of the radii and all of the test plates. The test plates must be of the correct shape (convex or concave or both, as required) and must have sufficient diameter to test the clear aperture of the lens surface (as determined by the semi-diameter value on the main spreadsheet). The test plate has sufficient diameter if the diameter is at least 3/4 of the clear aperture of the lens surface.

The radius of the test plate that best fits one of the radii is then substituted in for the actual radius. The variability of the chosen radius is removed, and the lens is reoptimized. For this reason it is important to allow compensators such as spacings to be variable as well as the radii to be fit. The reoptimization will adjust all remaining variables, including the remaining radii. After optimization, if there are any variable radii remaining, the procedure repeats. Note that the radii are fit in the order in which they best match test plates, not in the order they appear in the lens data editor.

During the fitting process, the number of radii remaining and the increase in the merit function will be presented. After all of the radii have been fit, a report will be presented on the screen. The report will provide the vendor identification information, and a list of the radii that were changed and which vendor test plate ID numbers were selected.

There is no way to know if the test plates selected are optimal. If there is a large number of test plates in the test plate list, and the values are reasonably continuous without large gaps, the fitting will usually be quite good. If the merit function increases unacceptably during the fitting process, either a different vendor test plate list should be used, the design needs to be modified, or some of the lenses may have to have custom test plates made. Usually, the last radii to be fit are the likely candidates for custom test plates. The report shows the order in which the radii were fit, with the best fit presented first, and the worst fit presented last.

There are occasions where the algorithm is unable to find any suitable test plate match for a given radius. This can occur when no test plate with a large enough diameter is sufficiently close to the desired radii. If this occurs, a message stating "NO MATCH FOUND!" will be printed in the report file, and that radii will subsequently be ignored. Usually this means test plates for this radii will have to be custom made.

All test plate data is provided by the respective vendors, and no warranty is provided as to the accuracy or completeness of the data. For details on the latest vendor test plate lists, contact the vendor. New test plates may be added to ZEMAX; vendors should contact Focus Software directly for details.

## **ZPL Macros**

---

*Purpose:*

Runs ZEMAX Programming Language macros.

*Discussion:*

See the chapter "ZEMAX Programming Language".



## Remove All Variables

---

### *Purpose:*

This tool provides a quick way to remove all of the variables defined in the current lens.

### *Discussion:*

Variables are removed by setting each of the values to "fixed" at the current value.

## Edit Coating File

---

### *Purpose:*

Invokes the Windows NOTEPAD editor to edit the file COATING.DAT. This file contains the material and coating definitions.

### *Discussion:*

See the chapter "Polarization Analysis". If the file COATING.DAT is edited, the lens file must be saved and then opened again in order to update ZEMAX to use the new catalog data.

## Quick Focus

---

### *Purpose:*

Quickly focuses an optical system by adjusting the back focal distance.

### *Settings:*

Item	Description
Spot Size Radial	Focus to best RMS spot size image plane.
Spot Size X Only	Focus to best RMS x direction spot size image plane.
Spot Size Y Only	Focus to best RMS y direction spot size image plane.
Wavefront Error	Focus to best RMS wavefront error image plane.
Use Centroid	Reference all calculations to the image centroid rather than the chief ray. This option is slightly slower, but more appropriate for systems dominated by coma.

### *Discussion:*

This feature adjusts the thickness of the surface prior to the image plane. The thickness chosen is that which minimizes the RMS aberration. There are several different RMS computations that may be performed as described in the "options" table above. The "best focus" position depends upon which criterion is selected. The RMS is always computed as a polychromatic average over the field, using the defined fields, wavelengths, and weights.

## Add Fold Mirror

---

### *Purpose:*

Inserts a fold mirror, including the required coordinate breaks, to bend a beam.

### *Settings:*

Item	Description
Fold surface	Selects which surface will become the fold mirror. The selected surface should be a dummy surface already located at the desired fold position.
Reflect angle	The angle between the incident and reflected beam.
Tilt type	Select either tilt about the local x or y axis.

*Discussion:*

This feature inserts two dummy surfaces, one before and one after the selected fold surface. The fold surface then becomes a mirror, and the two newly inserted adjacent surfaces are set to be coordinate breaks with the appropriate tilt angles. The second tilt angle is set as a pickup from the first tilt angle. Finally, all subsequent surface thicknesses and curvatures are changed sign to account for the new mirror.

This feature may not provide useful or even correct alterations if the selected fold surface is not a plane, standard type dummy surface in air. The dummy surface should be located at the desired fold mirror position before using this feature. For example, to insert a fold mirror midway between two lenses 100 mm apart, a dummy surface should be inserted between the two lenses, with the thickness before and after the dummy being set to 50 mm. The dummy surface then should be used as the fold surface.

This feature may also not work for lenses with multi-configuration data that alters the thicknesses or glass types of any surfaces from the fold surface onward.

This page intentionally left blank.

## Introduction

---

This chapter provides detailed descriptions of each of the report features ZEMAX supports. The contents of any window may be printed by selecting Print from the window menu. The printer used is the current printer selected by the "printers" tool in the Windows control panel icon.

The Settings menu option allows the default parameters for the calculation to be changed. Selecting this option will cause a feature dialog box to appear on the screen. The dialog box typically has five buttons: OK, Cancel, Save, Load, and Reset. OK causes the window to recalculate and redisplay the data with the currently selected options. Cancel reverts the options to those selected before the dialog box appeared, and does not update the data in the window. Save stores the currently selected options as defaults for future use, and then causes the window to recalculate and redisplay the data. Load loads the most recently saved default options, but does not exit the dialog box. Reset returns the options to the "factory default" values, but does not exit the dialog box.

A report window can be updated by double clicking with the left mouse button within the window. The settings dialog box is invoked by clicking with the right mouse button.

## System Data

---

*Purpose:*

Display system data.

*Settings:*

None.

*Discussion:*

This feature generates a text window which lists many system-related parameters, such as pupil positions and sizes, magnification, F/#, etc.

## Surface Data

---

*Purpose:*

Display surface data.

*Settings:*

Item	Description
Surface	The surface number to display
the data for.	

*Discussion:*

This feature generates a text window which displays surface-specific data. The data includes surface and element powers and focal lengths, edge thickness, index of refraction, and other data for the surface.

If the glass type of the surface is a "model" glass, then ZEMAX will list out the index of refraction at each defined wavelength calculated from the model glass parameters. Also listed is the name of the "Best Fit Glass", which is the name of the glass in the currently loaded catalog(s) which has the closest index of refraction to the model glass in an RMS sense. Specifically, ZEMAX computes the index error as the sum of the squares of the difference between the model index and the actual glass index using the dispersion formulas. The sum is over the defined wavelengths. The index error is computed for every glass in the current catalogs, and the glass with the lowest RMS index deviation is designated as the best fit glass.

Note that the best fit glass may have a different V number than the model glass, however, this is due to the approximations made in the model glass dispersion. Since index of refraction is the physically significant parameter, only the index is used in making the glass selection.

When changing from model glass to a real "fixed" glass, the same algorithm is used to make the selection.

## Prescription Data

---

### *Purpose:*

This function generates a list of all surface data, and summarizes the lens system. This is the feature to use for printing the contents of the Lens Data Editor.

### *Settings:*

Item	Description
General Data	Includes F/#, pupil positions, magnification, etc.
Surface Data	Surface type, radii, thickness, glass, semi-diameter, conic.
Surf. Data Detail	Parameter data.
Edge Thicknesses	The X and Y edge thickness of each surface.
Multi-configuration data	A table of multi-configuration operand data.
Solves & Variables	Solve types and data, variables.
Index Data	Index at each wavelength/surface.
Global Vertex	Global coordinates of each vertex and direction cosines. See discussion.
Element Volume	Volume in cc, density, and mass for spherical optics.
F/ Numbers	Lists working F/# for each field position and wavelength combination.
Cardinal Points	Lists the locations of principal, nodal, focal, and anti-principal points.

### *Discussion:*

This is a catch-all text generating feature. The file created contains many details of the lens, such as specification data, indices of refraction, global coordinates, element volumes, and more. It is suitable for describing the lens prescription.

The global coordinates are referenced to the object surface coordinate system if the object surface thickness is not infinity. If the object surface thickness is infinity, then the global vertex coordinates are referenced to the coordinate system of the first surface. If this first surface is a coordinate break, then the coordinates are referenced to the coordinate system defined after the coordinate break decenters and tilts.

## Merit Function Listing

---

### *Purpose:*

This function generates a text listing of the merit function which can be saved or printed.

### *Settings:*

None.

## Tolerance Listing

---

### *Purpose:*

This function generates a text listing of the tolerances which can be saved or printed.

### *Settings:*

None.

## Coating Listing

---

*Purpose:*

This function generates a text listing of the definitions of materials and coatings contained in the file COATING.DAT.

*Settings:*

None.

This page intentionally left blank.

## Introduction

---

ZEMAX models many types of optical components. These include conventional spherical glass surfaces, plus aspheres, toroids, cylinders, and others, ZEMAX can also model components such as diffraction gratings, "thin" lenses, binary optics, Fresnel lenses, holograms, and others.

Because of the large number of surface types ZEMAX supports, a conventional spreadsheet arrangement for the user interface would be difficult to use. For example, there is no need to have a column for "diffraction order" for a surface that has no diffractive capability. To make the user interface as uncluttered as possible, ZEMAX uses different surface types to indicate what kinds of data are needed to define that type of surface.

## Parameter data

---

A "standard" surface can be a plane, spherical, or conic aspheric surface which is followed by a homogeneous material (such as air, mirror, or glass). The only parameters required are a radius (which may be infinity to yield a plane), a thickness, a conic constant (the default zero value indicates a sphere), and the name of the glass type.

Other surfaces types use these same basic data, as well as other values. For example, the "even asphere" surface uses all the "standard" column data plus eight additional values which describe coefficients on a polynomial. These eight values are called parameters. The eight parameters are called parameter 1, parameter 2, and so.

The most important property of the parameter values to understand is that their meaning changes depending upon the surface type selected. For example, the "even asphere" surface type uses parameter 1 to specify the coefficient on the parabolic aspheric term. However, the "paraxial" surface uses parameter 1 to specify the focal length of the surface. Both surface types use parameter 1, but for different purposes, since both surface types are never used on the same surface at the same time.

This sharing of data storage simplifies the ZEMAX interface, as well as reduces the total memory required to run the program. But does this sharing make ZEMAX more difficult to use, since you have to remember which parameter does what? The answer is no, because ZEMAX keeps track of what each parameter does on each surface for you.

After you change a surface from "standard" to some other surface type, ZEMAX will automatically change the column headings on the parameter columns to reflect what each parameter does on that surface. All you have to do is type in the appropriate data in the correct cell. As you move the cursor from cell to cell, the column headings will always show you what that cell is used for. If the current surface does not use the parameter column, the column heading will display "Unused" and the column number. For more information about the Lens Data Editor, see the chapter "Editors Menu".

## Extra data

---

ZEMAX-EE supports surfaces which cannot be described by only eight parameter values. For example, the "binary optic 1" surface type requires eight parameters as well as up to 67 additional numbers. This would make for a very large spreadsheet, and so a separate editor is used for the extra data values. However, the concept is the same. The extra data values are shared by each surface type, and change meaning depending upon the surface type selected. The column headings in the Extra Data Editor also change as the cursor is moved from cell to cell. For more information about the Extra Data Editor, see the chapter "Editors".

## Summary of surface types

---

ZEMAX models planes, spheres, and conics; all of these surface types are grouped under the category of standard surface. By double clicking on the "Surface Type" column other surface models can be selected. A pop-up screen will appear which lists all of the available surface types which can be selected. ZEMAX supports many different types of surfaces in addition to the standard surface. The types are summarized in the following table. The columns marked SE, XE, and EE indicate availability of the surface type in each of the three editions of ZEMAX.



## SUMMARY OF SURFACE TYPES

Surface Model	Description	SE	XE	EE
Standard	Includes planes, spheres, and conics	Yes	Yes	Yes
Even Asphere	Standard surface plus polynomial asphere terms	Yes	Yes	Yes
Odd Asphere	Standard surface plus polynomial asphere terms	Yes	Yes	Yes
Paraxial	Thin lens surface, has ideal behavior	Yes	Yes	Yes
Paraxial XY	Thin lens with separate specification in X,Y	Yes	Yes	Yes
Toroidal	Conic and aspheric toroids and cylinders	Yes	Yes	Yes
Biconic	A conic asphere in X and Y independently	Yes	Yes	Yes
Toroidal Grating	Ruled grating on a conic toroid	Yes	Yes	Yes
Cubic Spline	Rotationally symmetric fit to 8 points	Yes	Yes	Yes
Hologram 1	Two-point optically fabricated hologram	Yes	Yes	Yes
Hologram 2	Two-point optically fabricated hologram	Yes	Yes	Yes
Coordinate Break	Permits rotation and decentration	Yes	Yes	Yes
Polynomial	Eight term polynomial expansion in x and y	Yes	Yes	Yes
Fresnel	Plane surface with refractive power	Yes	Yes	Yes
ABCD	Uses ABCD matrix to model a "black box"	Yes	Yes	Yes
Alternate	Standard surface with alternate solution selected	Yes	Yes	Yes
Diffraction Grating	Ruled grating on standard surface	Yes	Yes	Yes
Conjugate	Defines surface with perfect imaging at two points	Yes	Yes	Yes
Tilted	Defines a tilted surface without changing coordinate systems	Yes	Yes	Yes
Irregular	A standard surface with decenter, tilt and other deformations	Yes	Yes	Yes
Gradient 1	Radial gradient index material surface	No	Yes	Yes
Gradient 2	Radial gradient index material surface	No	Yes	Yes
Gradient 3	Radial and axial gradient index material surface	No	Yes	Yes
Gradient 4	X, Y, and Z gradient index material surface	No	Yes	Yes
Gradient 5	Radial and axial gradient index material surface with dispersion model	No	Yes	Yes
Gradient 6	Radial gradient index material surface with Gradient Lens Corp. dispersion model	No	Yes	Yes
Gradient 7	Spherical gradient profile	No	Yes	Yes
GRADIUM™	Axial gradient index material surface with dispersion model	No	Yes	Yes
Gradient 9	Radial gradient index material surface with NSG SELFOC lens dispersion model.	No	Yes	Yes
Gradient 10	Y gradient index material surface with dispersion model.	No	Yes	Yes
Zernike	Uses 36 Zernike polynomials to define sag	No	No	Yes
Zernike Phase	Uses 36 Zernike polynomials to define phase	No	No	Yes
Extended Polynomial	Uses 65 term polynomial expansion to define sag	No	No	Yes
Binary Optic 1	Uses 65 term polynomial to define phase	No	No	Yes
Binary Optic 2	Uses radial polynomial to define phase	No	No	Yes
Extended Cubic Spline	Rotationally symmetric fit of up to 98 points	No	No	Yes
Extended Asphere	Uses radial polynomial to define sag	No	No	Yes
Extended Odd Asphere	Uses odd terms of radial powers	No	No	Yes
VLS Grating	Variable line space grating surface	No	No	Yes

Surface Model	Description	SE	XE	EE
Elliptical Grating	Elliptical grating with aspheric terms	No	No	Yes
Superconic	Superconic asphere with fast convergence	No	No	Yes
Extended Fresnel	Polynomial Fresnel on a polynomial surface	No	No	Yes
Grid Sag	A surface shape described by a grid of points	No	No	Yes
Grid Phase	A phase surface described by a grid of points	No	No	Yes
Generalized Fresnel	XY polynomial Fresnel on an aspheric substrate	No	No	Yes
Periodic	Cosine shaped surface	No	No	Yes
Toroidal Hologram	Toroidal substrate with two point optically fabricated hologram.	No	No	Yes
Jones Matrix	General Jones Matrix for modifying polarization state	No	No	Yes
Atmospheric Refraction	Refraction caused by looking through Earth's atmosphere	No	No	Yes
Zone Plate	Fresnel Zone Plate model using annular rings of varying depth	No	No	Yes
User Defined	A general surface which uses an arbitrary user defined function to describe the refractive, reflective, diffractive, or gradient properties of the surface.	No	No	Yes

## The standard surface

The most commonly used optical surface is a spherical surface. The sphere is centered on the current optical axis, with the vertex located at the current axis position. ZEMAX treats planes as a special case of the sphere (a sphere with infinite radius of curvature) and conics as a special case as well. The "sag" or z-coordinate of the standard surface is given by

$$z = \frac{c r^2}{1 + \sqrt{1 - (1 + k)c^2 r^2}}$$

where  $c$  is the curvature (the reciprocal of the radius),  $r$  is the radial coordinate in lens units and  $k$  is the conic constant. The conic constant is less than -1 for hyperbolas, -1 for parabolas, between -1 and 0 for ellipses, 0 for spheres, and greater than 0 for oblate ellipsoids. For more information on conic constants, see any of the reference books given in the chapter "Introduction". The standard surface does not use any of the parameter values.

There are a few handy formulas for converting the semi major and semi minor axis lengths of an elliptical surface to a radius and conic description. If "a" is the semi major axis length, and "b" is the semi minor axis length, then

$$\frac{1}{c} = R = \pm \frac{b^2}{a}$$

$$k = -\varepsilon^2 = -\left[ \frac{a^2 - b^2}{a^2} \right]$$

## The even asphere surface

Rotationally symmetric polynomial aspheric surfaces are described by a polynomial expansion of the deviation from a spherical (or aspheric described by a conic) surface. The even asphere surface model uses only the even powers of the radial coordinate to describe the asphericity. The model uses the base radius of curvature and the conic constant. The surface sag is given by

$$z = \frac{c r^2}{1 + \sqrt{1 - (1 + k)c^2 r^2}} + \alpha_1 r^2 + \alpha_2 r^4 + \alpha_3 r^6 + \alpha_4 r^8 + \alpha_5 r^{10} + \alpha_6 r^{12} + \alpha_7 r^{14} + \alpha_8 r^{16}$$

Note that the eight coefficients have units. The coefficients are the only numbers supplied to the spreadsheet, ZEMAX calculates  $r$  as required to trace rays through the surface. The eight coefficients are entered in the corresponding parameter columns, as shown in the following table. The even asphere model is commonly used to describe corrector plates for Schmidt telescopes.

#### PARAMETER DEFINITIONS FOR EVEN ASPHERE SURFACES

Parameter 1	Parameter 2	Parameter 3	Parameter 4	Parameter 5	Parameter 6	Parameter 7	Parameter 8
$\alpha_1$	$\alpha_2$	$\alpha_3$	$\alpha_4$	$\alpha_5$	$\alpha_6$	$\alpha_7$	$\alpha_8$

### The odd asphere surface

The odd asphere model deviation is similar to the even asphere, except both even and odd powers of  $\rho$  are used. The name is slightly misleading, but somehow appropriate for the unusual shapes generated by this model. The sag is given by

$$z = \frac{c r^2}{1 + \sqrt{1 - (1 + k)c^2 r^2}} + \beta_1 r^1 + \beta_2 r^2 + \beta_3 r^3 + \beta_4 r^4 + \beta_5 r^5 + \beta_6 r^6 + \beta_7 r^7 + \beta_8 r^8 .$$

Note that the eight coefficients have units. The coefficients are the only numbers supplied to the spreadsheet, ZEMAX calculates  $r$  as required to trace rays through the surface. The eight coefficients are entered in the corresponding parameter columns, as shown in the following table. The odd aspheric model can be used to generate cone-shaped surfaces called axicons. For a discussion of modeling axicons with the odd asphere surface, see the chapter "Advanced Topics".

#### PARAMETER DEFINITIONS FOR ODD ASPHERE SURFACES

Parameter 1	Parameter 2	Parameter 3	Parameter 4	Parameter 5	Parameter 6	Parameter 7	Parameter 8
$\beta_1$	$\beta_2$	$\beta_3$	$\beta_4$	$\beta_5$	$\beta_6$	$\beta_7$	$\beta_8$

### The paraxial surface

The paraxial surface acts as an ideal thin lens. The paraxial surface is particularly useful for analyzing and optimizing systems whose output is collimated light. Afocal systems can be modeled by placing a paraxial surface before the image plane, and then setting the thickness of the paraxial surface (the distance to the image plane) to be the same as the focal length. If the focal length is chosen to be one meter, then all aberration data presented in microns (such as ray fans) can be interpreted to be in units of microradians.

One parameter is required to model the paraxial surface: the focal length. The focal length is that which would be measured in air (unity index) although the paraxial model will support imaging into a non-unity index medium. The paraxial surface shape is plane.

#### PARAMETER DEFINITIONS FOR PARAXIAL SURFACES

Parameter 1	Parameter 2-8
Focal length	Not used

### The paraxial XY surface

The paraxial XY surface type is similar to the paraxial surface except the optical power can be specified in the X and Y directions separately. This surface can therefore be used as a paraxial cylindrical or toroidal lens. Two parameters are required to define the paraxial XY surface: the X power, and the Y power. The paraxial XY surface shape is plane.

## PARAMETER DEFINITIONS FOR PARAXIAL XY SURFACES

Parameter 1	Parameter 2	Parameter 3-8
X Power	Y Power	Not used

### The toroidal surface

Toroidal surfaces are formed by defining a curve in the Y-Z plane, and then rotating this curve about an axis parallel to the Y axis and intersecting the Z axis. Toroids are defined using a base radius of curvature in the Y-Z plane, as well as a conic constant and polynomial aspheric coefficients. The curve in the Y-Z plane is defined by:

$$z = \frac{c y^2}{1 + \sqrt{1 - (1 + k) c^2 y^2}} + \alpha_1 y^2 + \alpha_2 y^4 + \alpha_3 y^6 + \alpha_4 y^8 + \alpha_5 y^{10} + \alpha_6 y^{12} + \alpha_7 y^{14} .$$

This curve is similar to the even aspheric surface sag formula, except the sixteenth order term has been omitted, and the coordinate argument is y, not r. This curve is then rotated about an axis a distance R from the vertex. This distance R is referred to as the radius of revolution, and may be positive or negative.

The Y-Z radius of curvature is specified in the same column on the spreadsheet editor as the radius for standard surfaces. The radius of revolution is set on parameter column number 1. To model a cylinder lens which is flat in the X direction, either enter a large radius of revolution, or simply enter zero, which ZEMAX interprets as infinite radius.

Note if the Y-Z radius is set to infinity, a surface with power in x but not in y can be described, therefore, the cylinder may be oriented in either direction. The other parameter columns are used for the optional aspheric coefficients, as specified in the following table. If aspheric coefficients are required in the X direction, then rotate the toroid with a pair of coordinate break surfaces and rotate about Z. If different aspheric surfaces are required in both the X and Y directions, see the "biconic", "polynomial" and "extended polynomial" surfaces described elsewhere in this chapter.

## PARAMETER DEFINITIONS FOR TOROIDAL SURFACES

Parameter 1	Parameter 2	Parameter 3	Parameter 4	Parameter 5	Parameter 6	Parameter 7	Parameter 8
Radius of Revolution	$\alpha_1$	$\alpha_2$	$\alpha_3$	$\alpha_4$	$\alpha_5$	$\alpha_6$	$\alpha_7$

### The biconic surface

The biconic surface is similar to a toroidal surface, except the conic constant and base radius may be different in the X and Y directions. The biconic surface allows specification of Rx, Ry, Kx, and Ky directly. The sag of a biconic is given by:

$$z = \frac{c_x x^2 + c_y y^2}{1 + \sqrt{1 - (1 + k_x) c_x^2 x^2 - (1 + k_y) c_y^2 y^2}} ,$$

where

$$c_x = \frac{1}{R_x} , c_y = \frac{1}{R_y} .$$

The radius in the x direction is set in the parameter 1 column. If set to zero, the x radius is interpreted to be infinity.

## PARAMETER DEFINITIONS FOR BICONIC SURFACES

Parameter 1	Parameter 2	Parameter 3-8
$R_x$	$k_x$	Not used

### The toroidal grating surface

Toroidal grating surfaces are similar to regular toroidal surfaces, except no aspheric sag terms are supported, and a diffraction grating may be placed on the toroidal surface. Toroidal gratings are described by defining a curve in the Y-Z plane, and then rotating this curve about an axis parallel to the Y axis and intersecting the Z axis. Toroidal gratings are defined using a base radius of curvature in the Y-Z plane, as well as a conic constant. The curve in the Y-Z plane is defined by:

$$z = \frac{c y^2}{1 + \sqrt{1 - (1 + k)c^2 y^2}}$$

This curve is similar to the standard surface sag formula, except the coordinate argument is  $y$ , not  $r$ . This curve is then rotated about an axis a distance  $R$  from the vertex. This distance  $R$  is referred to as the radius of revolution, and may be positive or negative.

The Y-Z radius of curvature is specified in the same column on the spreadsheet editor as the radius for standard surfaces. The radius of revolution is set on parameter column number 1. To model a cylinder lens which is flat in the X direction, either enter a large radius of revolution, or simply enter zero, which ZEMAX interprets as infinite radius.

Note if the Y-Z radius is set to infinity, a surface with power in  $x$  but not in  $y$  can be described, therefore, the cylinder may be oriented in either direction.

The diffraction grating is defined in terms of the number of lines per micron and the diffraction order. These values are specified in parameter columns 2 and 3, respectively. The grating lines are parallel to the local  $x$  axis, and are evenly spaced when projected onto a plane.

## PARAMETER DEFINITIONS FOR TOROIDAL GRATING SURFACES

Parameter 1	Parameter 2	Parameter 3	Parameter 4-8
Radius of Revolution	Grating Lines per micron	Diffraction order	Not used

### The cubic spline surface

The cubic spline surface is described by eight sag values which are the distances between the vertex tangent plane and the surface. The eight values represent the sag at one-eighth, two-eighths, and so on to eight-eighths of the semi-diameter of that surface. Cubic spline surfaces are rotationally symmetric, and are normal to the local axis at the vertex (there is no cusp). All eight points must be defined. A subset cannot be used, although the semi-diameter may be defined to exceed the useful aperture of the surface. This is often required because of the steep curvatures occasionally introduced by spline fitting. Spline surfaces are used to describe unusual correctors, headlamps, and other non-standard optical surfaces.

If eight points provides an overly coarse sampling, see also the "Extended cubic spline" surface section later in this chapter. Spline surfaces can cause rough or ragged ray tracing results. A more general and much smoother solution is to use the grid sag surface, which is not restricted to rotationally symmetry. See the discussion later in this chapter.

## PARAMETER DEFINITIONS FOR CUBIC SPLINE SURFACES

Parameter 1	Parameter 2	Parameter 3	Parameter 4	Parameter 5	Parameter 6	Parameter 7	Parameter 8
Sag at 1/8	Sag at 2/8	Sag at 3/8	Sag at 4/8	Sag at 5/8	Sag at 6/8	Sag at 7/8	Sag at 8/8

### The hologram 1 surface

The hologram 1 surface can be used to model optically constructed holograms. The hologram surface can be plane, spherical, or conical, and the medium behind the hologram can be air or glass. The glass can also be "MIRROR" which indicates the hologram is constructed and used in reflection. The hologram itself is described by the x, y, and z coordinates of two different construction points, a construction wavelength, and the diffraction order. The hologram deviates ray paths according to the equation

$$\hat{n} \times (r'_o - r'_r) = \frac{m\lambda'}{\lambda} \hat{n} \times (r_o - r_r),$$

where  $\hat{n}$  is the unit vector normal to the surface of the hologram at ray intersection point,  $r_o$  is the unit vector along the first construction beam,  $r_r$  is the unit vector along the second construction beam,  $r'_r$  is the unit vector along the incident readout beam,  $r'_o$  is the refracted ray,  $\lambda$  and  $\lambda'$  are the construction and playback wavelengths, and  $m$  is the diffraction order. A value of  $m = 0$  means the ray is undeviated, while other integral values of  $m$  refer to higher diffracted orders. The notation used here is from the book "Aberrations of Optical Systems" by Welford, Adam Hilger (1986). Modeling holograms requires an understanding of their behavior which is beyond the scope of this manual, and the user is advised to see the discussion in Welford, or some other reference, before using this feature.

Most holograms are constructed and used in transmission or reflection. There are rare occasions where the hologram is constructed in transmission, and then the substrate is aluminized and used in reflection. This special case can be simulated with the hologram surface by specifying a negative construction wavelength. Although ray tracing will be correct for this special case, OPD tracing will not work.

ZEMAX only models holograms to the extent of deviating ray paths. Other properties, such as efficiency and relative transmission are not supported. For more information on ray tracing with holograms, see the Welford reference.

The two construction beams are defined in terms of their source points. The x, y, and z coordinates of the source points are measured relative to the hologram vertex coordinate are defined in current system units (millimeters, centimeters, etc.) ZEMAX computes the unit vector at the ray-surface intersection point using the local coordinate data and the construction point data for the two construction beams. The construction wavelength always has units of microns.

The hologram 1 surface assumes that both construction beams diverge from the specified construction points. Because of the reciprocity of the construction beams, this is identical to the case where both construction beams converge toward the construction points. Some hologram fabrication methods require one beam to be converging while the other beam is diverging. See "The hologram 2 surface" for information on modeling this latter type of hologram.

## PARAMETER DEFINITIONS FOR HOLOGRAM SURFACES

Parameter 1	Parameter 2	Parameter 3	Parameter 4	Parameter 5	Parameter 6	Parameter 7	Parameter 8
$X_1$	$Y_1$	$Z_1$	$X_2$	$Y_2$	$Z_2$	$\lambda$	M

### The hologram 2 surface

The hologram 2 surface is very similar to the hologram 1 surface. The key difference is that the hologram 1 surface assumes both construction beams either diverge from or converge to the construction points, whereas the hologram 2 surface assumes one construction beam converges to one construction point, and the other construction beam diverges from the other construction point. Which beam is first or second does not matter due to reciprocity. The parameter data is the same for both hologram 1 and hologram 2 surfaces.

## The coordinate break surface

The coordinate break surface is used to define a new coordinate system in terms of the current system. It is always considered a "dummy" surface for ray tracing purposes. There are six parameters used to describe the new coordinate system: x-decenter, y-decenter, tilt x, tilt y, tilt z, and a flag to indicate the order of tilting and decentration. Coordinate breaks are always relative to the current, not the global coordinate system. This very general and powerful surface has many desirable properties, but it takes a little practice to use. For a tutorial on using coordinate breaks, see the chapter "Tutorial".

The only way to tilt or decenter a surface or group of surfaces is through the use of the coordinate break surface. There is one extremely important consideration to keep in mind when using a coordinate break: The order of the decenters and tilts matters!

If the "order" flag is set to zero, ZEMAX first decenters in x, and then in y (because these coordinates are orthogonal which is first does not matter). Then ZEMAX tilts about the current local x axis. Note that rotation about the x axis will change the orientation of the y and z axes. The rotation about the new y axis is then performed, which changes the orientation of the x and z axes. Finally, the rotation about the resulting z axis is performed.

If the "order" flag is any other value (such as unity), then the tilts are done first, in the order z, y, and x, and then the decenters are done. This "order" flag is extremely useful because a single coordinate break can undo an earlier coordinate break, even for compound tilts and decenters.

The coordinate break acts like a plane surface oriented in the coordinate system after the decenters and tilts have been applied. Refraction occurs at the surface boundary if the index of refraction changes at the boundary. Coordinate breaks can therefore have a glass type, like any other surface. However, the surface is never drawn, and it should not be used to define the boundary between two media. The glass type should be the same as the adjacent surfaces, unless one of the surfaces is a mirror. Coordinate breaks themselves can never be mirrors, nor can the object surface be a coordinate break.

The coordinate rotations are described by a series of three rotation matrices (if the "order" flag is zero):

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} \cos(\theta_z) & -\sin(\theta_z) & 0 \\ \sin(\theta_z) & \cos(\theta_z) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(\theta_y) & 0 & \sin(\theta_y) \\ 0 & 1 & 0 \\ -\sin(\theta_y) & 0 & \cos(\theta_y) \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta_x) & -\sin(\theta_x) \\ 0 & \sin(\theta_x) & \cos(\theta_x) \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

### PARAMETER DEFINITIONS FOR COORDINATE BREAK SURFACES

Parameter 1	Parameter 2	Parameter 3	Parameter 4	Parameter 5	Parameter 6	Parameter 7-8
Decenter X	Decenter Y	Tilt About X	Tilt About Y	Tilt About Z	Order	Not used

## The polynomial surface

The non-rotationally symmetric polynomial aspheric surface is called simply "Polynomial". The base radius of curvature and the conic constant are not used by this surface model. The sag of the polynomial surface is given by

$$z = \gamma_1 x^2 + \gamma_2 x^4 + \gamma_3 x^6 + \gamma_4 x^8 + \gamma_5 y^2 + \gamma_6 y^4 + \gamma_7 y^6 + \gamma_8 y^8.$$

This surface model can be used to describe "potato chip" surfaces and some anamorphic aspheres. A more general polynomial surface is also available, see the section "The extended polynomial surface" later in this chapter.

## PARAMETER DEFINITIONS FOR POLYNOMIAL SURFACES

Parameter 1	Parameter 2	Parameter 3	Parameter 4	Parameter 5	Parameter 6	Parameter 7	Parameter 8
$\gamma_1$	$\gamma_2$	$\gamma_3$	$\gamma_4$	$\gamma_5$	$\gamma_6$	$\gamma_7$	$\gamma_8$

### The Fresnel surface

The Fresnel surface model is used to simulate flat surfaces which have been etched to have a spherical (or optionally aspheric) profile on a small scale (for Fresnel lenses which are not flat see the extended Fresnel surface type description). The surface intercept is determined by computing the intersection of the incoming ray with a plane. Once the plane intercept points are found, the surface is then treated as spherical (or aspherical) for the purposes of refraction into the next medium. This is only an approximation to a real Fresnel lens, however. The real Fresnel lens has grooves which may alter the exact intercept point. The model used here is adequate for Fresnel lens which have fine grooves (the groove depth is very shallow compared to the aperture). Extreme Fresnel lenses, such as those used in lighthouses, are not modeled well. The radius of curvature and conic constant, if any, are specified in the same manner as a standard surface. The parameter values are exactly the same as for the even asphere model; polynomial aspherics are supported to 16th order.

## PARAMETER DEFINITIONS FOR FRESNEL SURFACES

Parameter 1	Parameter 2	Parameter 3	Parameter 4	Parameter 5	Parameter 6	Parameter 7	Parameter 8
$\alpha_1$	$\alpha_2$	$\alpha_3$	$\alpha_4$	$\alpha_5$	$\alpha_6$	$\alpha_7$	$\alpha_8$

### The ABCD surface

The ABCD surface provides a powerful method for modeling "black box" optical systems. If you have a lens (or a complete optical system) which is only a subsection of what you wish to model, and you do not have prescription data for the individual components, you can still model the behavior to first order. You will need to know the principal plane locations and the first order properties such as focal length and effective propagation length.

The ABCD surface accepts eight parameters:  $A_x$ ,  $B_x$ ,  $C_x$ ,  $D_x$ ,  $A_y$ ,  $B_y$ ,  $C_y$ , and  $D_y$ . These are used to form two two-by-two matrices (one for the x-direction and one for the y) which are used to alter the ray as it crosses the surface. The exit ray is related to the incident ray by

$$\begin{bmatrix} x' \\ \omega'_x \end{bmatrix} = \begin{bmatrix} A_x & B_x \\ C_x & D_x \end{bmatrix} \begin{bmatrix} x \\ \omega_x \end{bmatrix},$$

with a similar expression for the y components. For details on using matrix optics, see Hecht, Optics. A great variety of optical components can be modeled with this method, including thin and thick lenses, cylinders, and even gradient index media.

However, because there is no reliable way to compute the phase through an ABCD surface, any calculation that requires OPD data, such as OPD fans, MTF, and Zernike coefficients, will not be supported if an ABCD surface is present in the lens description.

## PARAMETER DEFINITIONS FOR ABCD SURFACES

Parameter 1	Parameter 2	Parameter 3	Parameter 4	Parameter 5	Parameter 6	Parameter 7	Parameter 8
$A_x$	$B_x$	$C_x$	$D_x$	$A_y$	$B_y$	$C_y$	$D_y$

### The alternate surface

There are two solutions to the ray-surface intercept equations used in tracing a ray to the next spherical optical surface. ZEMAX selects the correct solution in the vast majority of cases. However, in certain systems, so-called "strange" rays are actually intended to intersect the next surface at the other, "alternate" solution. Strange rays are most likely to occur after a grazing incidence reflection in which the ray is still traveling in the same direction (the Z component of the ray vector does not change sign). The alternate surface model is identical to the standard surface model, except the alternate solution is used. ZEMAX may not be able to correctly compute the optical path difference when the alternate surface is being used.



## The diffraction grating surface

The diffraction grating surface can be used to model straight-line ruled gratings. ZEMAX models the diffraction grating as a special case of the hologram model where the two construction beams are from infinitely distant sources. The grating lines are parallel to the local x-axis. Other orientations can be simulated by using a coordinate break surface before and after the grating surface. For a plane grating, rays traced to the grating are refracted according to the equation

$$n_2 \sin \theta_2 - n_1 \sin \theta_1 = \frac{M\lambda}{d} = M\lambda T,$$

where  $d$  is the grating spacing (always in microns),  $\theta_2$  is the refracted angle,  $\theta_1$  is the incident angle,  $M$  is the diffraction order,  $\lambda$  is the wavelength (always in microns),  $n_1$  and  $n_2$  are the indices of refraction before and after the grating, and  $T$  is the grating period in lines per micron. Note that the sign convention for  $M$  is completely arbitrary. ZEMAX uses the definition for  $T$  (lines per micron) rather than  $d$  (microns per line). The grating surface can be plane, spherical, or conical, and the medium before the grating, as well as the grating itself, can be air, glass, "MIRROR" or any other valid glass type. The grating is described by the y-spacing of the grating lines measured in lines per micron (independent of the system units) and the diffraction order. ZEMAX only models gratings to the extent of deviating ray paths. Other properties, such as efficiency, and relative transmission are not supported. If the grating spacing is too small (or if  $T$  is too large) to satisfy the grating relation, then a "Ray missed surface" error will be reported.

### PARAMETER DEFINITIONS FOR DIFFRACTION GRATING SURFACES

Parameter 1	Parameter 2	Parameter 3-8
Grating lines per micron	Diffraction order	Not used

## The conjugate surface

The conjugate surface is defined by two user-specified points. ZEMAX always uses the surface vertex as the reference point; the two points required to define the conjugate surface are specified relative to this vertex. The conjugate surface will always perfectly image one point to the other point, assuming the surface is a mirror. Although the conjugate surface can have any material type, it is useful to think of it as being defined by its reflective properties.

If the z-coordinates of the two points are either both positive or both negative, then the image formed from one of the points to the other is real. In this case, the distance from one of the points to an arbitrary point on the surface, plus the distance from the arbitrary point on the surface to the second point, is constant for all points on the surface. One additional constraint is needed to make the surface unique: the surface must pass through the vertex of the local coordinate system. If the surface is reflective, then one point is the conjugate of the other, hence the name.

The surface generated by these two points satisfies the following expression if both  $z_1$  and  $z_2$  have the same sign:

$$\sqrt{(x-x_1)^2+(y-y_1)^2+(z-z_1)^2} + \sqrt{(x-x_2)^2+(y-y_2)^2+(z-z_2)^2} = \sqrt{x_1^2+y_1^2+z_1^2} + \sqrt{x_2^2+y_2^2+z_2^2}.$$

Note that the surface must intersect the point (0,0,0). Several types of surfaces can be formed with this model. For example, a sphere can be formed by setting the x and y values to zero, and the two z values each to the radius of the sphere. An elliptical surface of arbitrary orientation can be formed by specifying non-zero values for either the x or y values.

If  $z_1$  and  $z_2$  have opposite signs, then the image formed from one of the points to the other is virtual. In this case, the distance from one of the points to an arbitrary point on the surface, minus the distance from the arbitrary point on the surface to the second point, is constant for all points on the surface. Like the real imaging case, the surface must pass through the vertex of the local coordinate system.

The surface generated by these two points satisfies the following expression if  $z_1$  and  $z_2$  have opposite signs:

$$\sqrt{(x-x_1)^2+(y-y_1)^2+(z-z_1)^2} - \sqrt{(x-x_2)^2+(y-y_2)^2+(z-z_2)^2} = \sqrt{x_1^2+y_1^2+z_1^2} - \sqrt{x_2^2+y_2^2+z_2^2}.$$

Note that the surface must intersect the point (0,0,0). Several types of surfaces can be formed with this model. For example, a hyperbola can be formed by setting the x and y values to zero, and the two z values to opposite values. If the z values are equal but opposite, then a plane will be generated.

The coordinates of the two construction points are specified in the parameter columns, as shown in the following table. Neither the z1 nor the z2 value can be zero.

#### PARAMETER DEFINITIONS FOR CONJUGATE SURFACES

Parameter 1	Parameter 2	Parameter 3	Parameter 4	Parameter 5	Parameter 6	Parameter 7-8
x1	y1	z1	x2	y2	z2	Not used

### The tilted surface

The tilted surface is simply a plane that makes an angle with respect to the x and y axes. The surface is easily defined in terms of the tangent angle between the plane and the X and Y axes:

$$z = x \tan\theta_x + y \tan\theta_y .$$

The tilted surface uses the first two parameters to define the tangents of the x and y angles. This surface is very useful for implementing tilted object and image surfaces, as well as tilted faces on prisms. It should not be used for implementing fold mirrors; use the coordinate break surface instead.

#### PARAMETER DEFINITIONS FOR TILTED SURFACES

Parameter 1	Parameter 2	Parameter 3-8
$\tan\theta_x$	$\tan\theta_y$	Not used

### The irregular surface

The irregular surface is a standard surface shape (plane, spherical, or conic) that has additional aspheric deviations in terms of decenter, tilt, spherical aberration, astigmatism, and coma. This surface type is primarily used by the tolerancing algorithm to model small irregularities in a standard shape surface. The surface sag is given by:

$$z = \frac{c^2}{1 + \sqrt{1 - (1 + k)c^2 r^2}} + Z_x \rho_x + Z_y \rho_y + Z_s \rho^4 + Z_a \rho_y^2 + Z_c \rho^2 \rho_y ,$$

where

$$\rho_x = \frac{x}{r_{\max}} , \rho_y = \frac{y}{r_{\max}} , \rho = \sqrt{\rho_x^2 + \rho_y^2} ,$$

and rmax is the maximum radial aperture of the lens, defined by the semi-diameter value for the surface. The coefficients Zx, Zy, Zs, Za, and Zc represent the amount of x tilt, y tilt, spherical aberration, astigmatism, and coma, respectively, in lens units at the maximum radial aperture.

The x and y coordinates of the previous equations are in the decentered coordinate system defined by

$$x' = x - dx , y' = y - dy ,$$

where the primes have been omitted for clarity.

Note that the tilt terms do not actually tilt the base sphere or conic surface. The tilt terms are in addition to the base standard surface shape. This formulation is perfectly acceptable for small amounts of tilt or aberration, which is typical when performing a tolerance analysis. See the Chapter "Tolerancing" for more details.

The irregular surface uses the first seven parameters to define the decenter and Z coefficients. All the coefficients are measured in lens units.

## PARAMETER DEFINITIONS FOR IRREGULAR SURFACES

Parameter 1	Parameter 2	Parameter 3	Parameter 4	Parameter 5	Parameter 6	Parameter 7	Parameter 8
dx	dy	Zx	Zy	Zs	Za	Zc	Not used

### The gradient 1 surface

This feature is only available in the XE and EE editions of ZEMAX.

Media whose index of refraction is described by

$$n = n_0 + n_{r2} r^2,$$

where  $r^2 = x^2 + y^2$ , can be modeled using the gradient 1 surface. Three parameters are required: the maximum step size,  $\Delta t$ , the base index,  $n_0$ , and the quadratic index,  $n_{r2}$ . Note that  $n_{r2}$  has units.

#### **Discussion on maximum step size for GRIN surfaces**

The maximum step size  $\Delta t$  determines the tradeoff between ray tracing speed and accuracy. The exact value required depends upon the numerical aperture of the system and the magnitude of the coefficients. To determine a suitable step size, start at a large value (on the order of the thickness of the surface), and then perform a spot diagram. Note the RMS spot size. Now decrease the step size by a factor of two. If the RMS spot size changes by less than a few percent, the new step size is probably small enough. Otherwise, decrease the step again. For the final design phase, you may want to decrease the step size again. Using too small a step size needlessly slows the ray tracing speed without enhancement of accuracy. OPD tracing generally converges more slowly than ray tracing, so you may want to repeat the above procedure while inspecting the OPD fan. Occasionally check to insure that the step size is appropriate as the design progresses.

#### **Restrictions on surfaces following GRIN surfaces**

The ray trace through a gradient index media requires iteration to determine the point of intersection of the ray with the surface following the gradient index surface. Because of this, not all surface types are allowed to follow gradient index surface types. If the type of the surface after a gradient index surface is not supported, a message will be presented indicating the error. Support for additional surface types to follow a GRIN surface may be added upon request; contact FSI for details.

## PARAMETER DEFINITIONS FOR GRADIENT 1 SURFACES

Parameter 1	Parameter 2	Parameter 3	Parameter 4-8
$\Delta t$	$n_0$	$n_{r2}$	Not used

### The gradient 2 surface

This feature is only available in the XE and EE editions of ZEMAX.

Media whose index of refraction is described by

$$n^2(x,y,z) = n_0 + n_{r2} r^2 + n_{r4} r^4 + n_{r6} r^6 + n_{r8} r^8 + n_{r10} r^{10} + n_{r12} r^{12},$$

where  $r^2 = x^2 + y^2$ , can be modeled using the gradient 2 surface. Eight parameters are required: the maximum step size,  $\Delta t$ , the base index squared,  $n_0$ , and the remaining six coefficients of the preceding equation. Note that some of the coefficients have units. Most radial GRIN medium can be described by this formulation by redefining the coefficients provided by the manufacturer.

The maximum step size  $\Delta t$  determines the tradeoff between ray tracing speed and accuracy. For details see "Discussion on maximum step size for GRIN surfaces" in the section "The gradient 1 surface". Also see the discussion "Restrictions on surfaces following GRIN surfaces".

## PARAMETER DEFINITIONS FOR GRADIENT 2 SURFACES

Parameter 1	Parameter 2	Parameter 3	Parameter 4	Parameter 5	Parameter 6	Parameter 7	Parameter 8
$\Delta t$	$n_0$	$n_{r2}$	$n_{r4}$	$n_{r6}$	$n_{r8}$	$n_{r10}$	$n_{r12}$

### The gradient 3 surface

This feature is only available in the XE and EE editions of ZEMAX.

Media whose index of refraction is described by

$$n = n_0 + n_{r2} r^2 + n_{r4} r^4 + n_{r6} r^6 + n_{z1} z + n_{z2} z^2 + n_{z3} z^3,$$

where  $r^2 = x^2 + y^2$ , can be modeled using the gradient 3 surface. Eight parameters are required: the maximum step size,  $\Delta t$ , the base index,  $n_0$ , and the remaining six coefficients of the preceding equation. Note that some of the coefficients have units.

The maximum step size  $\Delta t$  determines the tradeoff between ray tracing speed and accuracy. For details see "Discussion on maximum step size for GRIN surfaces" in the section "The gradient 1 surface". Also see the discussion "Restrictions on surfaces following GRIN surfaces".

## PARAMETER DEFINITIONS FOR GRADIENT 3 SURFACES

Parameter 1	Parameter 2	Parameter 3	Parameter 4	Parameter 5	Parameter 6	Parameter 7	Parameter 8
$\Delta t$	$n_0$	$n_{r2}$	$n_{r4}$	$n_{r6}$	$n_{z1}$	$n_{z2}$	$n_{z3}$

### The gradient 4 surface

This feature is only available in the XE and EE editions of ZEMAX.

Media whose index of refraction is described by

$$n = n_0 + n_{x1} x + n_{x2} x^2 + n_{y1} y + n_{y2} y^2 + n_{z1} z + n_{z2} z^2,$$

can be modeled using the gradient 4 surface model. There are eight parameters required: the maximum step size,  $\Delta t$ , the base index,  $n_0$ , and the remaining six coefficients of the preceding equation. Note that some of the coefficients have units. This particular GRIN model is useful for gradient index optics with a cylindrical power profile. It is also useful for modeling thermal gradients in optical elements, if you have sufficient data to compute the coefficients.

The linear transverse terms,  $n_{x1}$  and  $n_{y1}$ , are ignored when performing paraxial ray tracing.

The maximum step size  $\Delta t$  determines the tradeoff between ray tracing speed and accuracy. For details see "Discussion on maximum step size for GRIN surfaces" in the section "The gradient 1 surface". Also see the discussion "Restrictions on surfaces following GRIN surfaces".

## PARAMETER DEFINITIONS FOR GRADIENT 4 SURFACES

Parameter 1	Parameter 2	Parameter 3	Parameter 4	Parameter 5	Parameter 6	Parameter 7	Parameter 8
$\Delta t$	$n_0$	$n_{x1}$	$n_{x2}$	$n_{y1}$	$n_{y2}$	$n_{z1}$	$n_{z2}$

### The gradient 5 surface

This feature is only available in the XE and EE editions of ZEMAX.

The gradient 5 surface has the gradient form:

$$n = n_0 + n_{r2} r^2 + n_{r4} r^4 + n_{z1} z + n_{z2} z^2 + n_{z3} z^3 + n_{z4} z^4,$$

where  $r^2 = x^2 + y^2$ . Eight parameters are required: the maximum step size,  $\Delta t$ , the base index,  $n_0$ , and the remaining six coefficients of the preceding equation. Note that some of the coefficients have units.

The important feature of the gradient 5 model is that the model allows specification of dispersion properties of the media. The dispersion data is user defined and is stored in an ASCII file called GRIN.DAT. The GRIN.DAT file contains blocks of 18 lines each. The first line in the file is the name of the material, which can be any name (without special characters such as spaces or quotes) less than 10 characters long. The second line in the block is the "reference" wavelength in microns. The next 16 lines contain the coefficients  $\alpha_1, \beta_1, \gamma_1, \delta_1, \alpha_2, \beta_2, \dots, \delta_4$ . There are no blank lines allowed between blocks. The supplied file GRIN.DAT contains coefficients to describe the lead silicate gradient material LPTPBD offered by LightPath Technologies at a reference wavelength of .58756 microns.

The material name is entered in the glass column of the gradient 5 surface type. If the glass column is blank, then dispersion effects are ignored.

To perform ray tracing, ZEMAX first computes the index at the reference wavelength using the preceding formula for  $n_{ref}$ . The index at any other wavelength is then computed using the following method based upon the Buchdahl formula:

$$n(\lambda) = n_{ref} + v_1 \omega + v_2 \omega^2 + v_3 \omega^3 + v_4 \omega^4,$$

$$\omega = \frac{\lambda - \lambda_{ref}}{1 + C[\lambda - \lambda_{ref}]},$$

$$C = \frac{1.0}{\lambda_{ref} - 0.187},$$

$$v_i = \alpha_i [n_{ref}]^3 + \beta_i [n_{ref}]^2 + \gamma_i [n_{ref}] + \delta_i,$$

where  $i$  varies from 1 to 4. The sixteen coefficients and the reference wavelength define the dispersion of the material, whereas the gradient coefficients specified by parameters 2-8 (see the following table) define the index gradient profile.

The maximum step size  $\Delta t$  determines the tradeoff between ray tracing speed and accuracy. For details see "Discussion on maximum step size for GRIN surfaces" in the section "The gradient 1 surface". Also see the discussion "Restrictions on surfaces following GRIN surfaces".

#### PARAMETER DEFINITIONS FOR GRADIENT 5 SURFACES

Parameter 1	Parameter 2	Parameter 3	Parameter 4	Parameter 5	Parameter 6	Parameter 7	Parameter 8
$\Delta t$	$n_0$	$n_{r2}$	$n_{r4}$	$n_{z1}$	$n_{z2}$	$n_{z3}$	$n_{z4}$

### The gradient 6 surface

This feature is only available in the XE and EE editions of ZEMAX.

The gradient 6 surface has the following gradient profile:

$$n = n_{00} + n_{10} r^2 + n_{20} r^4.$$

The difference between the gradient 6 and gradient 1 surface models is that the gradient 6 surface uses dispersion formulas to compute  $n_{00}$ ,  $n_{10}$ , and  $n_{20}$  automatically rather than using entries from the lens data editor.

The  $n_{00}$  value is computed from the following formula:

$$n_{00} = A + B\lambda^2 + \frac{C}{\lambda^2} + \frac{D}{\lambda^4},$$

with an identical expression (but different values for A, B, C, and D) for  $n_{10}$  and  $n_{20}$ . This expression uses nanometers rather than microns for  $\lambda$ .

The dispersion data is user defined and is stored in an ASCII file called GLC.DAT. The GLC.DAT file contains blocks of 13 lines each. The first line in the file is the name of the material, which can be any name (without special characters such as spaces or quotes) less than 10 characters long. The next 12 lines are the values for A, B, C, and D for  $n_{00}$ ,  $n_{10}$ , and then  $n_{20}$ , respectively. There are no blank lines allowed between blocks. ZEMAX can read in data for up to 25 different materials in the GLC.DAT file.

The dispersion data in the supplied GLC.DAT file was provided by the Gradient Lens Corporation (GLC) in Rochester, NY, (716) 235-2620. Contact GLC for detailed information on material properties. Not all of the materials offered by GLC are included in the GLC.DAT file. The following materials are included: ARS10, ARS20, ARS27, and ARS31.

To use the gradient 6 surface materials, simply change the surface type to gradient 6, and then enter the appropriate material name in the glass column of the Lens Data Editor.

The maximum step size  $\Delta t$  determines the tradeoff between ray tracing speed and accuracy. For details see "Discussion on maximum step size for GRIN surfaces" in the section "The gradient 1 surface". Also see the discussion "Restrictions on surfaces following GRIN surfaces".

#### PARAMETER DEFINITIONS FOR GRADIENT 6 SURFACES

Parameter 1	Parameter 2-8
$\Delta t$	Not used

### The gradient 7 surface

This feature is only available in the XE and EE editions of ZEMAX.

The gradient 7 surface has a gradient profile with spherical symmetry. The index of refraction is described by

$$n = n_0 + \alpha (r - R) + \beta (r - R)^2,$$

where

$$r = \frac{R}{|R|} \sqrt{x^2 + y^2 + (R - z)^2}.$$

The coordinates x, y, and z are the usual coordinates measured with respect to the vertex tangent plane, and R is the radius of the isoindex contours measured at the vertex. The isoindex contours are spherical shells centered about the point z = R. The starting index,  $n_0$ , is measured at the vertex of the surface, not at the center of the isoindex contours.

There are five parameters required: the maximum step size,  $\Delta t$ , the base index,  $n_0$ , R,  $\alpha$ , and  $\beta$ . Note that  $\alpha$  and  $\beta$  have units.

The isoindex contour radius R may be specified independently from the front or rear radius of curvature of the lens. However, if R is set to be zero, then ZEMAX assumes the isoindex radius and the front radius of curvature are equal.

The maximum step size  $\Delta t$  determines the tradeoff between ray tracing speed and accuracy. For details see "Discussion on maximum step size for GRIN surfaces" in the section "The gradient 1 surface". Also see the discussion "Restrictions on surfaces following GRIN surfaces".

## PARAMETER DEFINITIONS FOR GRADIENT 7 SURFACES

Parameter 1	Parameter 2	Parameter 3	Parameter 4	Parameter 5	Parameter 6-8
$\Delta t$	$n_0$	$R$	$\alpha$	$\beta$	Not used

### The GRADIUM™ surface

This feature is only available in the XE and EE editions of ZEMAX.

This surface type models lenses made from stock gradient index blanks available from LightPath Technologies, Inc. The blanks have an axial gradient index profile, which describes a reference index of refraction as a function of axial position within the glass blank. All that is required to define the lens is the starting position within the blank, the name of the stock blank profile, and of course the radii and thickness. The GRADIUM surface has a gradient index profile which is described by a polynomial of the form:

$$n = \sum_{i=0}^{11} n_i \left( \frac{z + \Delta z}{z_{\max}} \right)^i.$$

The  $z$  coordinate is the distance from the front vertex of the surface,  $z_{\max}$  is the maximum  $z$  coordinate of the blank, and the value  $\Delta z$  is the "offset" distance along the profile. Unlike most other gradient glass models in ZEMAX, the GRADIUM surface uses only fixed, predefined axial profile coefficients. The only design parameter required is the offset value,  $\Delta z$ . The available profiles are defined in the ASCII file PROFILE.DAT. For a list of the available profiles, see the Analysis, Gradient index, GRADIUM Profile feature described in the Chapter "Analysis Menu". The file format is a series of blocks of 13 lines of data defined as follows:

```
PROFILE_NAME GLASS_FAMILY MAX_Z
n0
n1
...
n11
```

Each block of data starts with the name of the profile, which can be any valid ASCII name less than 10 characters. On the same line, the glass family name follows, which must be a gradient index material as defined in the GRIN.DAT file, which is described in the section on the gradient 5 surface type. The glass family name defines the reference wavelength which the profile describes. The last entry on the line is the maximum  $z$  coordinate of the blank. The 12 polynomial coefficients follow, from  $n_0$  to  $n_{11}$ .

When ZEMAX performs ray tracing, the local  $z$  coordinate on the surface (which may be negative) is computed. The offset value is then added to determine where in the profile the coordinate lies.

Normally, the resulting value must always be positive and less than or equal to the maximum  $z$ , otherwise an error is generated (See the discussion of "Capping" below). The reference index is then evaluated. Once the reference index is computed, the index at the wavelength being traced is computed using the technique described in the section on the gradient 5 surface.

The reference wavelength index at the front vertex is displayed as parameter 3. This value may be changed in the Lens Data Editor. When a new value is entered, ZEMAX computes the appropriate  $\Delta z$  to yield the specified reference index. However, the  $\Delta z$  is the value that matters. The reference index is just displayed for convenience, and should not be made variable, or a multi-configuration operand. Note the reference index is the vertex index at the reference wavelength, which is the wavelength defined in the glass family definition file GRIN.DAT. It is not necessarily the primary wavelength.

The GRADIUM surface model also supports 4 additional parameter terms that are intended for use in tolerancing: Decenter X, Decenter Y, Tilt X, and Tilt Y. These four terms model an axial gradient that is not perfectly centered and not perfectly parallel to the local  $Z$  axis. The tolerance terms modify the axial profile by redefining the axial coordinate  $z$  as follows:

$$z' = t_x (x - d_x) + t_y (y - d_y) + t_z z ,$$

where

$$t_z = \left[ 1.0 - t_x^2 - t_y^2 \right]^{\frac{1}{2}},$$

and  $t_x$ ,  $t_y$ ,  $t_z$  are the coefficients of the unit vector which points along the axial gradient axis and  $d_x$  and  $d_y$  are the decenters in lens units of the start of the profile. If  $t_x$  and  $t_y$  are both zero, then the  $d_x$  and  $d_y$  values do not matter (since the gradient is only along the  $z$  axis) and the  $t_z$  value is unity. The terms  $t_x$  and  $t_y$  determine the slope of the profile axis in  $x$  and  $y$ , which is intended to model the tolerance of axial alignment between the gradient axis and the mechanical axis of the lens. This expression is a linear approximation which is only valid for asymptotically small  $t_x$  and  $t_y$ .

The tolerance terms,  $d_x$ ,  $d_y$ ,  $t_x$ , and  $t_y$ , are ignored when performing paraxial ray tracing.

Usually, only the defined range of the profile is used. However, in some cases the profile may be extended in one or both directions to add additional glass to the ends of the profile, which allows the use of GRADIUM in thicker lenses. This technique is called "capping". By default, ZEMAX turns capping off, so that any ray trace which requires glass beyond the profile limits is flagged as an error. This automatically enforces boundary constraints during optimization. To remove this restriction, the capping flag can be set to be 1, 2, or 3. The default value of zero indicates the blank is bounded to the profile length at both ends. If the capping flag is 1, then only the left edge is bounded (the right edge is allowed to pass beyond the profile limit). If the capping flag is 2, then only the right edge is bounded. And if the capping is 3, then neither the left nor right edges of the blank are bounded, and both the thickness and the offset may take on any values.

The maximum step size  $\Delta t$  determines the tradeoff between ray tracing speed and accuracy. For details see "Discussion on maximum step size for GRIN surfaces" in the section "The gradient 1 surface". Also see the discussion "Restrictions on surfaces following GRIN surfaces".

#### PARAMETER DEFINITIONS FOR GRADIUM SURFACES

Parameter 1	Parameter 2	Parameter 3	Parameter 4	Parameter 5	Parameter 6	Parameter 7	Parameter 8
$\Delta t$	$\Delta z$	$n_{ref}$	$d_x$	$d_y$	$t_x$	$t_y$	Capping

### The gradient 9 surface

This feature is only available in the XE and EE editions of ZEMAX.

The gradient 9 surface may be used to model SELFOC® materials available from NSG America, Inc. The "sag" or  $z$ -coordinate of the GRIN 9 surface is the same as the standard surface plus a "tilt" term in both  $x$  and  $y$  directions:

$$z = \frac{c r^2}{1 + \sqrt{1 - (1 + k)c^2 r^2}} + \alpha x + \beta y,$$

where  $c$  is the curvature (the reciprocal of the radius),  $r$  is the radial coordinate in lens units,  $k$  is the conic constant, and  $\alpha$  and  $\beta$  are the tangents of the tilt angles in  $x$  and  $y$ . Note this is not the same surface shape as a tilted standard surface, but it is a close approximation as long as the curvature is small, or if the tilt angles are small. The gradient 9 surface has the following gradient profile:

$$n = n_0 \left[ 1.0 - \frac{A}{2} r^2 \right].$$

Both  $A$  and  $n_0$  are functions of wavelength:

$$A(\lambda) = \left[ K_0 + \frac{K_1}{\lambda^2} + \frac{K_2}{\lambda^4} \right]^2,$$

$$n_0 = B + \frac{C}{\lambda^2},$$



where the wavelength is measured in microns. The dispersion data is user defined and is stored in an ASCII file called SELFOC.DAT. The SELFOC.DAT file contains blocks of 6 lines each. The first line in the file is the name of the material, which can be any name (without special characters such as spaces or quotes) less than 10 characters long. The next 5 lines are the values for B, C, K0, K1, and K2. There are no blank lines allowed between blocks. ZEMAX can read in data for up to 25 different materials in the SELFOC.DAT file.

The dispersion data in the supplied SELFOC.DAT file was provided by NSG America, Inc., in Somerset, NJ, (908) 469-9650. Contact NSG for detailed information on material properties. Not all of the materials offered by NSG are included in the SELFOC.DAT file. The included materials are: SLS-1.0, SLS-2.0, SLW-1.0, SLW-1.8, SLW-2.0, SLW-3.0, SLW-4.0, and SLH-1.8.

To use the gradient 9 surface materials, simply change the surface type to gradient 9, and then enter the appropriate material name in the glass column of the Lens Data Editor.

The maximum step size  $\Delta t$  determines the tradeoff between ray tracing speed and accuracy. For details see "Discussion on maximum step size for GRIN surfaces" in the section "The gradient 1 surface". Also see the discussion "Restrictions on surfaces following GRIN surfaces".

#### PARAMETER DEFINITIONS FOR GRADIENT 9 SURFACES

Parameter 1	Parameter 2	Parameter 3	Parameter 4-8
$\Delta t$	$\alpha$	$\beta$	Not used

### The gradient 10 surface

This feature is only available in the XE and EE editions of ZEMAX.

The gradient 10 surface models glass with the following gradient profile:

$$n = n_0 + n_{y1}|y| + n_{y2}|y^2| + n_{y3}|y^3| + n_{y4}|y^4| + n_{y5}|y^5| + n_{y6}|y^6| ,$$

where the || symbols indicate the absolute value. This form of gradient has a discontinuity at the plane  $y = 0$ , and the gradient is bisymmetric about the  $y = 0$  plane. If the glass type is left blank, then there is no dispersion. If a glass type is entered in the glass column, then it must be one of the GRIN 5 materials defined in the section on the GRIN 5 surface type. The above formula then defines the index profile at the reference wavelength for the material, and the index at other wavelengths is computed according to the Buchdahl dispersion model defined for the GRIN 5 surface.

The linear transverse term,  $n_{y1}$ , is ignored when performing paraxial ray tracing.

The maximum step size  $\Delta t$  determines the tradeoff between ray tracing speed and accuracy. For details see "Discussion on maximum step size for GRIN surfaces" in the section "The gradient 1 surface". Also see the discussion "Restrictions on surfaces following GRIN surfaces".

#### PARAMETER DEFINITIONS FOR GRADIENT 10 SURFACES

Parameter 1	Parameter 2	Parameter 3	Parameter 4	Parameter 5	Parameter 6	Parameter 7	Parameter 8
$\Delta t$	$n_0$	$n_{y1}$	$n_{y2}$	$n_{y3}$	$n_{y4}$	$n_{y5}$	$n_{y6}$

### The Zernike surface

This feature is only available in the EE edition of ZEMAX.

The Zernike surface is defined by the even aspheric surface (which supports planes, spheres, conics, and polynomial aspheres) plus additional aspheric terms defined by the Zernike coefficients. The surface sag is of the form:

$$z = \frac{c r^2}{1 + \sqrt{1 - (1 + k)c^2 r^2}} + \sum_{i=1}^8 \alpha_i r^{2i} + \sum_{i=0}^N A_i Z_i(\rho, \varphi)$$

where  $N$  is the number of Zernike coefficients in the series,  $A_i$  is the coefficient on the  $i^{\text{th}}$  Zernike polynomial,  $r$  is the radial ray coordinate in lens units,  $\rho$  is the normalized radial ray coordinate, and  $\varphi$  is the angular ray coordinate. The Zernike polynomials are defined in the table given in the Zernike coefficients section of the chapter "Analysis Menu". ZEMAX-EE currently supports terms up to order 36. The coefficients  $A_i$  all have units which are the same as the lens units, such as millimeters or inches. The coefficients  $\alpha_i$  have units, and are as defined in the section on the "Even Asphere" surface type.

Note that the Zernike surface describes surface deformations, not wavefront error directly! If you have Zernike coefficient data in terms of waves of OPD, as might have been measured with an interferometer, use the Zernike phase (as opposed to the Zernike) surface instead.

### EXTRA DATA DEFINITIONS FOR ZERNIKE SURFACES

Extra Data Number	Description
1	Number of terms.
2	Maximum radial aperture. Coefficients are normalized by this value.
3 - 39	Coefficients on Zernike polynomials 0 - 36, respectively, in lens units.
40 and above	Not used.

The "Number of terms" is used to specify the maximum Zernike term to be used in calculating the surface sag. This number is provided simply to speed the ray tracing calculation, as terms beyond this number are ignored. For example, if the maximum term number you wish to use is Zernike #8, then specify "9" in the number of terms column (the first Zernike term is number 0, therefore there are 9 terms).

The maximum radial aperture is the normalization radius. Zernike polynomials are orthonormal over the unit circle, and so the maximum radial aperture should be set to the radius over which the coefficient data was normalized. Zernike polynomials tend to diverge quite rapidly beyond the normalization radius, and so care should be taken that rays do not strike the surface beyond this radius. Although the ray tracing algorithm may work, the data may be inaccurate.

This surface does not use parameter data. For details on optimization using the Zernike surface type, see the chapter "Optimization" under "Optimizing extra data values". For details on editing the extra data coefficients, see the chapter "Editors Menu".

### The Zernike phase surface

This feature is only available in the EE edition of ZEMAX.

The Zernike phase surface is defined by the standard surface (which supports planes, spheres, and conics) plus additional phase terms defined by the Zernike coefficients. The surface sag is identical to the standard surface formula. The additional phase terms deviate and add optical path to the rays as they cross the surface. This surface model is well suited to modeling system aberrations for which measured interferometer data is available. The Zernike phase surface can also be used to model some holograms and binary optics surfaces. The phase added to the rays is given by

$$\Phi = \sum_{i=0}^N A_i Z_i(\rho, \varphi)$$

where  $N$  is the number of Zernike coefficients in the series,  $A_i$  is the coefficient on the  $i^{\text{th}}$  Zernike polynomial,  $\rho$  is the normalized radial ray coordinate, and  $\varphi$  is the angular ray coordinate. The Zernike polynomials are defined in the table given in the Zernike coefficients section of the chapter "Analysis Menu". ZEMAX supports up to 37 terms, including the zero term. The coefficients  $A_i$  all have units of waves. One wave is  $2\pi$  radians. Note that the phase

of the surface is independent of the wavelength; ZEMAX accounts for the deviation in the ray path based upon the wavelength being traced.

Note that the Zernike phase surface describes phase variations (or wavefront error), not surface deformations directly! If you have Zernike coefficient data in terms of surface deformations, as might have been measured with a profilometer, use the Zernike (as opposed to Zernike phase) surface instead.

### EXTRA DATA DEFINITIONS FOR ZERNIKE PHASE SURFACES

Extra Data Number	Description
1	Number of terms.
2	Maximum radial aperture. Coefficients are normalized by this value.
3 - 39	Coefficients on Zernike polynomials 0 - 36, respectively, in units of primary waves.
40 and above	Not used.

The "Number of terms" is used to specify the maximum Zernike term to be used in calculating the phase. This number is provided simply to speed the ray tracing calculation, as terms beyond this number are ignored. For example, if the maximum term number you wish to use is Zernike #8, then specify "9" in the maximum term number column (the first Zernike term is number 0, therefore there are 9 terms).

The maximum radial aperture is the normalization radius. Zernike polynomials are orthonormal over the unit circle, and so the maximum radial aperture should be set to the radius over which the coefficient data was normalized. Zernike polynomials tend to diverge quite rapidly beyond the normalization radius, and so care should be taken that rays do not strike the surface beyond this radius. Although the ray tracing algorithm may work, the data may be inaccurate.

The Zernike phase surface does not use any of the parameter data. For details on optimization using the Zernike phase surface type, see the chapter "Optimization" under "Optimizing extra data values". For details on editing the extra data coefficients, see the chapter "Editors Menu".

#### **Zernike phase coefficients sign conventions**

The Zernike surface uses the same sign conventions as the binary 1 surface type does. See the section "The binary 1 surface" for a discussion of these sign conventions.

### **The extended polynomial surface**

This feature is only available in the EE edition of ZEMAX.

The extended polynomial surface is similar to the polynomial surface, except more terms are allowed. The surface also supports a base conic asphere surface upon which the polynomial aspheric terms are added. The surface sag is of the form:

$$z = \frac{c r^2}{1 + \sqrt{1 - (1 + k)c^2 r^2}} + \sum_{i=0}^N A_i E_i(x,y)$$

where  $N$  is the number of polynomial coefficients in the series, and  $A_i$  is the coefficient on the  $i^{th}$  extended polynomial term. The polynomials are a simple power series in  $x$  and  $y$ . The first term is  $x$ , then  $y$ , then  $x^2$ ,  $x^2y$ ,  $y^2$ , etc. There are 2 terms of order 1, 3 terms of order 2, 4 terms of order 3, etc. The maximum order is 18, which makes a maximum of 189 polynomial aspheric coefficients.

For example, the 12th term in the polynomial expansion, which is extra data term number 14, is the coefficient on the term in  $x^2y^2$ . The coefficients  $A_i$  all have units which are the same as the lens units, such as millimeters or inches.

### EXTRA DATA DEFINITIONS FOR EXT. POLYNOMIAL SURFACES

Extra Data Number	Description
1	Maximum term number.

Extra Data Number	Description
2	Normalization radius. All ray-intercept points are divided by this number to determine the x and y coordinates for polynomial evaluation.
3-191	Polynomial terms.
192 and above	Not used.

The "Maximum term number" is used to specify the maximum polynomial term to be used in calculating the surface sag. This number is provided simply to speed the ray tracing calculation, as terms beyond this number are ignored. For example, if the last term in the series you wish to use is  $xy^3$ , which is term number 13, then specify "13" in the maximum term number column. This number can also be set to zero to temporarily "turn off" the polynomial sag terms. Note that the term number is 13, because it is the 13th term in the polynomial expansion, and NOT 15, which is the "extra data number", which is simply the position in the extra data list of parameters. The "extra data number" is always two greater than the term number! The maximum term number is the same as "N" in the surface sag equation.

The "extra data number" is always two greater than the term number!

The normalization radius is used to scale the X and Y intercept coordinates of the ray. This is done so that the coefficients all have units of lens units. For example, if lens units are millimeters (mm), the normalization radius is 100, and the coefficient on polynomial term  $x^3$  is 4.0, the sag on the surface due to this one term at  $x = -100$  mm is -4.0 mm, the sag at  $x = +100$  mm is 4.0 mm, and the sag at  $x = 50$  mm is 0.5 mm (one half cubed times 4 mm).

The extended polynomial surface does not use any of the parameter data. For details on optimization using the extended polynomial surface type, see the chapter "Optimization" under "Optimizing extra data values". For details on editing the extra data coefficients, see the chapter "Editors Menu".

For modeling surfaces that are not well described by polynomials, see the grid sag surface description later in this chapter.

## The binary optic 1 surface

This feature is only available in the EE edition of ZEMAX.

Binary optics, also called kinoforms, are similar to holograms and diffraction gratings in that small grooves or lines across the optical surface impart a change in phase of the wavefront passing through the surface. ZEMAX does not model the wavelength-scale grooves directly. Instead, ZEMAX uses the phase advance or delay represented by the surface locally to change the direction of propagation of the ray. Other effects, such as scattering, efficiency, or multiple order diffraction are ignored. Binary surfaces can have zero thickness, with no index change across the surface, or may have different media on either side of the surface. In this case, the refraction due to the material change will be accounted for as well as the diffraction effects. Full exact polychromatic ray and OPD tracing is supported.

The binary optic 1 surface is similar to the extended polynomial surface, except the polynomial terms represent the variation in phase (rather than surface height) across the optic surface. The coefficients therefore have units of radians rather than lens units. The shape of the binary optic 1 surface is identical to the even asphere surface type; planes, spheres, conics, and polynomial aspheres up to order 16 are supported. The sag of the surface is given by the following expression:

$$z = \frac{cr^2}{1 + \sqrt{1 - (1+k)c^2r^2}} + \alpha_1r^2 + \alpha_2r^4 + \alpha_3r^6 + \alpha_4r^8 + \alpha_5r^{10} + \alpha_6r^{12} + \alpha_7r^{14} + \alpha_8r^{16},$$

where the terms are identical to those in the even asphere surface model. See that section earlier in this chapter for a complete discussion. The binary optic 1 surface adds phase to the ray according to the following polynomial expansion:

$$\Phi = \sum_{i=0}^N A_i E_i(x,y)$$

where  $N$  is the number of polynomial coefficients in the series, and  $A_i$  is the coefficient on the  $i^{\text{th}}$  extended polynomial term. The polynomials are a simple power series in  $x$  and  $y$ , as described in the section "The extended polynomial surface". The coefficients  $A_i$  all have units of radians ( $2\pi$  radians is one wave).

#### EXTRA DATA DEFINITIONS FOR BINARY OPTIC 1 SURFACES

Extra Data Number	Description
1	Maximum term number.
2	Normalization radius. All ray-intercept points are divided by this number to determine the $x$ and $y$ coordinates for polynomial evaluation.
3-191	Polynomial terms.
192 and above	Not used.

The "Maximum term number" is used to specify the maximum polynomial term to be used in calculating the surface phase. This number is provided simply to speed the ray tracing calculation, as terms beyond this number are ignored. For example, if the last term in the series you wish to use is  $x y^3$ , which is term number 13, then specify "13" in the maximum term number column. This number can also be set to zero to temporarily "turn off" the polynomial phase terms. Note that the term number is 13, because it is the 13th term in the polynomial expansion, and NOT 15, which is the "extra data number", which is simply the position in the extra data list of parameters. The "extra data number" is always two greater than the term number!

The "extra data number" is always two greater than the term number!

The normalization radius is used to scale the  $X$  and  $Y$  intercept coordinates of the ray. This is done so that the coefficients all have units of radians. For example, if lens units are millimeters (mm), the normalization radius is 100, and the coefficient on polynomial term  $x^3$  is 4.0, the phase on the surface due to this one term at  $x = -100$  mm is -4.0 radians, the phase at  $x = +100$  mm is 4.0 radians, and the phase at  $x = 50$  mm is 0.5 radians (one half cubed times 4 radians).

The binary optic 1 surface uses eight terms of the parameter data for the polynomial aspheric surface profile description. For details on optimization using the binary optic 1 surface type, see the chapter "Optimization" under "Optimizing extra data values". For details on editing the extra data coefficients, see the chapter "Editors Menu".

For modeling phase surfaces that are not well described by polynomials, see the grid phase surface description later in this chapter.

#### **Binary optic coefficients sign conventions**

It is important to understand the binary optic surface phase coefficients sign convention. Imagine a collimated beam being focused by an infinitesimal thin binary optic surface. If the collimated beam becomes a focused, converging beam on the other side of the binary optic, then the path length of a marginal ray is longer than the path length of the axis ray. Therefore, the binary optic must add a negative path length to the marginal ray. Using this convention, a positive power binary optic has a negative quadratic phase coefficient. Although the sign convention chosen by ZEMAX is arbitrary, it may be important to know what the convention is as far as fabrication decisions are concerned. It is always a good idea to test a few simple cases to verify the sign convention being used by the software before committing a design to fabrication.

#### **The binary optic 2 surface**

This feature is only available in the EE edition of ZEMAX.

Binary optics, also called kinoforms, are similar to holograms and diffraction gratings in that small grooves or lines across the optical surface impart a change in phase of the wavefront passing through the surface. ZEMAX does

not model the wavelength-scale grooves directly. Instead, ZEMAX uses the phase advance or delay represented by the surface locally to change the direction of propagation of the ray. Other effects, such as scattering, efficiency, or multiple order diffraction are ignored. Binary surfaces can have zero thickness, with no index change across the surface, or may have different media on either side of the surface. In this case, the refraction due to the material change will be accounted for as well as the diffraction effects. Full exact polychromatic ray and OPD tracing is supported.

The binary optic 2 surface is similar to the extended asphere surface, except there are additional polynomial terms which represent the variation in phase (rather than surface height) across the optic surface. The coefficients therefore have units of radians rather than lens units. The shape of the binary optic 2 surface is identical to the even asphere surface type; planes, spheres, conics, and polynomial aspheres up to order 16 are supported. The sag of the surface is given by the following expression:

$$z = \frac{c r^2}{1 + \sqrt{1 - (1 + k)c^2 r^2}} + \alpha_1 r^2 + \alpha_2 r^4 + \alpha_3 r^6 + \alpha_4 r^8 + \alpha_5 r^{10} + \alpha_6 r^{12} + \alpha_7 r^{14} + \alpha_8 r^{16},$$

where the terms are identical to those in the even asphere surface model. See that section earlier in this chapter for a complete discussion. The binary optic 2 surface adds phase to the ray according to the following polynomial expansion:

$$\Phi = \sum_{i=1}^N A_i \rho^{2i}$$

where  $N$  is the number of polynomial coefficients in the series,  $A_i$  is the coefficient on the  $2i^{th}$  power of  $\rho$ , which is the normalized radial aperture coordinate. The first extra data term is  $N$ , the number of terms, which can be zero to exclude all binary effects, or any integer up to 198 (or the  $\rho^{396}$  term). Extra data value number 2 is the normalization radius. Extra data values 3 through 200 are the coefficients on  $\rho^2$  through  $\rho^{396}$ .

#### EXTRA DATA DEFINITIONS FOR BINARY OPTIC 2 SURFACES

Extra Data Number	Description
1	Maximum term number. The maximum is 198.
2	Normalization radius. All ray-intercept points are divided by this number to determine the x and y coordinates for polynomial evaluation.
3	Coefficient in radians on $\rho^2$
4	Coefficient in radians on $\rho^4$
n+2	Coefficient in radians on $\rho^{2n}$

The "Maximum term number" is used to specify the maximum polynomial term to be used in calculating the surface phase. This number is provided simply to speed the ray tracing calculation, as terms beyond this number are ignored. For example, if the last term in the series you wish to use is  $\rho^{14}$ , which is term number 7, then specify "7" in the maximum term number column. This number can also be set to zero to temporarily "turn off" the polynomial phase terms. Note that the term number is 7, because it is the 7th term in the polynomial expansion, and NOT 9, which is the "extra data number", which is simply the position in the extra data list of parameters. The "extra data number" is always two greater than the term number!

The "extra data number" is always two greater than the term number!

The normalization radius is used to scale the X and Y intercept coordinates of the ray. This is done so that the coefficients all have units of radians. For example, if lens units are millimeters (mm), the normalization radius is 100, and the coefficient on polynomial term  $\rho^4$  is 4.0, the phase on the surface due to this one term at  $x = 100$  mm is 4.0 radians, and the phase at  $x = 50$  mm is 0.25 radians (one half to the fourth times 4 radians).

The binary optic 2 surface uses all eight terms of the parameter data for the optional polynomial aspheric surface profile description. For details on optimization using the binary optic 2 surface type, see the chapter "Optimization" under "Optimizing extra data values". For details on editing the extra data coefficients, see the chapter "Editors Menu".

For modeling phase surfaces that are not well described by polynomials, see the grid phase surface description later in this chapter.

### **Binary optic coefficients sign conventions**

See the section "The binary 1 surface" for a discussion of sign conventions.

## **The extended cubic spline surface**

This feature is only available in the EE edition of ZEMAX.

The extended cubic spline surface is similar to the cubic spline surface, except more terms are allowed. The standard cubic spline surface supports eight and only eight terms, while the extended cubic spline supports between 4 and 198 terms. The surface sag is defined by a list of sag points evenly spaced along the radius of a rotationally symmetric shape. The sag must always be zero at zero radius, and so this point is not entered into the spreadsheet. The surface is continuous with no cusp at the origin.

If you have a list of sag points that describes your optical surface (as might be generated from a computer program other than ZEMAX) and you wish to enter them into ZEMAX, use the load capability of the extra data editor. This feature is described in the chapter "Editors Menu".

### EXTRA DATA DEFINITIONS FOR EXT. CUBIC SPLINE SURFACES

Extra Data Number	Description
1	Maximum number of spline points.
2	Step size in lens units between points along the radius.
3	The sag at a distance of one step size away from the vertex.
4...n	The sag at a distance of 2... (n-2) step sizes away from the vertex.

The "Maximum number of spline points" is used to specify how many points define the surface. This number may be between 4 and 198. At least 4 points are required to define the curved surface.

The extended cubic spline surface does not use any of the parameter data. For details on optimization using the extended cubic spline surface type, see the chapter "Optimization" under "Optimizing extra data values". For details on editing the extra data coefficients, see the chapter "Editors Menu".

Spline surfaces can cause rough or ragged ray tracing results. A more general and much smoother solution is to use the grid sag surface, which is not restricted to rotationally symmetry. See the discussion later in this chapter.

## **The extended asphere surface**

This feature is only available in the EE edition of ZEMAX.

The extended asphere surface is similar to the "even asphere" surface described earlier in this chapter. However, the extended asphere can support aspheric coefficients up to order 396, whereas the even asphere is limited to 16. Also, a slightly different method is used for computing the coefficients of the polynomial terms. The sag of the surface is given by the following expression:

$$z = \frac{c r^2}{1 + \sqrt{1 - (1 + k)c^2 r^2}} + \sum_{i=1}^N \alpha_i r^{2i},$$

where the first expression is identical to the standard surface, and the second term is a power series sum over a normalized radial coordinate. The normalized radial coordinate  $\rho$  is used because then the coefficients  $\alpha_i$  all have units of lens units (millimeters, or whatever). The extra data terms are defined in the following table.

### EXTRA DATA DEFINITIONS FOR EXTENDED ASPHERE SURFACES

Extra Data Number	Description
1	Maximum term number. The maximum is 198.
2	Normalization radius. All ray-intercept points are divided by this number to determine the x and y coordinates for polynomial evaluation.
3	Coefficient in lens units on $\rho^2$
4	Coefficient in lens units on $\rho^4$
n+2	Coefficient in lens units on $\rho^{2n}$

The "Maximum term number" is used to specify the maximum polynomial term to be used in calculating the surface sag. This number is provided simply to speed the ray tracing calculation, as terms beyond this number are ignored. For example, if the last term in the series you wish to use is  $\rho^{14}$ , which is term number 7, then specify "7" in the maximum term number column. This number can also be set to zero to temporarily "turn off" the polynomial phase terms. Note that the term number is 7, because it is the 7th term in the polynomial expansion, and NOT 9, which is the "extra data number", which is simply the position in the extra data list of parameters. The "extra data number" is always two greater than the term number!

The "extra data number" is always two greater than the term number!

The normalization radius is used to scale the X and Y intercept coordinates of the ray. This is done so that the coefficients all have units of lens units. For example, if lens units are millimeters (mm), the normalization radius is 100, and the coefficient on polynomial term  $\rho^8$  is 4.0, the sag on the surface due to this one term at  $x = 100$  mm is 4.0 millimeters, and the sag at  $x = 75$  mm is 0.40045 millimeters (0.75 to the eighth times 4.0 millimeters).

The extended asphere surface does not use any of the parameter data. For details on optimization using the extended asphere surface type, see the chapter "Optimization" under "Optimizing extra data values". For details on editing the extra data coefficients, see the chapter "Editors Menu".

For modeling surfaces that are not well described by polynomials, see the grid sag surface description later in this chapter.

## The extended odd asphere surface

This feature is only available in the EE edition of ZEMAX.

The extended odd asphere surface is similar to the "odd asphere" surface described earlier in this chapter. However, the extended odd asphere can support aspheric coefficients up to order 198, whereas the odd asphere is limited to 8. Also, a slightly different method is used for computing the coefficients of the polynomial terms. The sag of the surface is given by the following expression:

$$z = \frac{c r^2}{1 + \sqrt{1 - (1 + k)c^2 r^2}} + \sum_{i=1}^N \alpha_i \rho^i,$$

where the first expression is identical to the standard surface, and the second term is a power series sum over a normalized radial coordinate. The normalized radial coordinate  $\rho$  is used because then the coefficients  $\alpha_i$  all have units of lens units (millimeters, or whatever). The extra data terms are defined in the following table.



## EXTRA DATA DEFINITIONS FOR EXTENDED ODD ASPHERE SURFACES

Extra Data Number	Description
1	Maximum term number. The maximum is 198.
2	Normalization radius. All ray-intercept points are divided by this number to determine the x and y coordinates for polynomial evaluation.
3	Coefficient in lens units on $\rho^1$
4	Coefficient in lens units on $\rho^2$
n+2	Coefficient in lens units on $\rho^n$

The "Maximum term number" is used to specify the maximum term to be used in calculating the surface sag. This number is provided simply to speed the ray tracing calculation, as terms beyond this number are ignored. For more information, see "The extended asphere surface" in the previous pages.

### The variable line space grating surface

This feature is only available in the EE edition of ZEMAX.

The variable line space grating surface models a special grating with straight, but unevenly spaced lines. The grating is described in "Variable line-space gratings: new designs for use in grazing incidence spectrometers", M. Hettrick and S. Bowyer, Applied Optics, Vol. 22, No. 24, p3921 (1983). The grating is designed to form images upon a concave spherical surface. A converging beam is used to illuminate the grating in reflection. The grating lines are parallel to the local x axis, and lie in the x-z plane. Only plane gratings are supported, and the feature has only been tested when the grating is used in reflection (the glass type must be "MIRROR").

Although the hologram surface type could be used to model the ideal grating, in practice holographic construction is not always possible. A compromise is made in the fabrication, where the grating lines are straight and parallel (as opposed to curved for the hologram case). This introduces aberrations into the beam, which this surface model accounts for.

Five parameters are used to describe the surface: M, the diffraction order upon playback; L, the (positive) radius of the concave focal surface,  $\cos(\alpha_0)$ , the cosine of the angle the incident beam makes with the plane of the grating,  $\cos(\beta_0)$ , the cosine of the angle the reflected beam makes with the plane of the grating, and  $\lambda_0$ , the construction wavelength. This is the wavelength at which the grating will reflect the axis ray along the angle  $\beta_0$ .

The grating period in lines per micron is given by

$$T = \frac{1}{\lambda_0} \left[ \frac{L \cos \beta_0 - y}{\sqrt{y^2 + L^2 - 2yL \cos \beta_0}} - \frac{L \cos \alpha_0 - y}{\sqrt{y^2 + L^2 - 2yL \cos \alpha_0}} \right]$$

where y varies across the grating surface. The variable line space grating surface uses five of the parameter surfaces, but none of the extra data surfaces.

#### PARAMETER DEFINITIONS FOR VARIABLE LINE SPACE GRATING SURFACES

Parameter 1	Parameter 2	Parameter 3	Parameter 4	Parameter 5	Parameter 6-8
<i>M</i>	<i>L</i>	$\cos \alpha_0$	$\cos \beta_0$	$\lambda_0$	Not used

### The elliptical grating surface

This feature is only available in the EE edition of ZEMAX.

The elliptical grating surface is a combination of an elliptical surface with polynomial aspheric terms and a grating with straight, equally spaced fringes. The grating lines are parallel to the local x-axis. Other orientations can be simulated by using a coordinate break surface before and after the grating surface. For a plane grating, rays traced to the grating are refracted according to the equation

$$n_2 \sin \theta_2 - n_1 \sin \theta_1 = \frac{M\lambda}{d} = M\lambda T,$$

where  $d$  is the grating spacing (always in microns),  $\theta_2$  is the refracted angle,  $\theta_1$  is the incident angle,  $M$  is the diffraction order,  $\lambda$  is the wavelength (always in microns),  $n_1$  and  $n_2$  are the indices of refraction before and after the grating, and  $T$  is the grating period in lines per micron. ZEMAX uses the definition for  $T$  (lines per micron) rather than  $d$  (microns per line).  $T$  can be set to zero which effectively eliminates the grating component. This is useful for modeling elliptical aspheres which are not gratings. The medium behind the grating surface may be air, glass, or a mirror. ZEMAX only models gratings to the extent of deviating ray paths. Other properties, such as efficiency and relative transmission are not supported. If the grating spacing is too small (or if  $T$  is too large) to satisfy the grating relation, then a "Ray missed surface" error will be reported.

The surface shape is an ellipse with added polynomial aspheric terms. The sag of the surface is given by the following expression:

$$z = \frac{c u^2}{1 + \sqrt{1 - u^2}} + \sum_{i=1}^N A_i E_i(x, y),$$

where

$$u^2 = a^2 x^2 + b^2 y^2.$$

In these expressions,  $x$ ,  $y$ , and  $z$  are the ray-surface intercept coordinates,  $a$ ,  $b$ , and  $c$  are the coefficients which define the shape of the ellipse, and the extended polynomials represented by  $E$  are defined in the section "The extended polynomial surface" found elsewhere in this chapter.

#### PARAMETER DEFINITIONS FOR ELLIPTICAL GRATING SURFACES

Parameter 1	Parameter 2	Parameter 3	Parameter 4	Parameter 5	Parameter 6-8
Grating lines per micron	Diffraction order	a	b	c	Not used

### The superconic surface

This feature is only available in the EE edition of ZEMAX.

The most common form of a polynomial aspheric surface uses a power series expansion in the radial coordinate  $r$  to define the surface sag, where  $r$  is defined by

$$r^2 = x^2 + y^2.$$

For example, the even aspheric surface described earlier in this chapter uses such an expansion. Since  $r$  does not depend upon  $z$ , the expansion term is the distance from the vertex to the point on the surface as projected on to the tangent plane. Generally, the departure of the asphere from the tangent plane increases with radial aperture. As the departure increases, the power series expansion parameter  $r$  corresponds to a point on the tangent plane which is farther from the point on the surface. This causes the expansion to have poor convergence.

A novel solution proposed by Alan Greynolds of Breault Research Organization is to instead expand in powers of the distance from the vertex to the point on the surface. The expansion is then in terms of

$$s^2 = x^2 + y^2 + z^2.$$

Starting with the conic equation for a surface

$$kz^2 - 2Rz + x^2 + y^2 + z^2 = 0,$$

where  $k$  is the conic constant and  $R$  is the radius of curvature, a general power series expansion can be made of the form

$$Az^2 - 2Bz + C = 0 .$$

The constants are defined as

$$A = \frac{k}{R} ,$$

$$B = 1 + U_1s^2 + U_2s^4 + \dots , \text{ and}$$

$$C = \frac{s^2}{R} + V_1s^4 + V_2s^6 + \dots ,$$

where U and V are coefficients which define the aspheric shape. Note that if all the U and V terms are zero, a standard conic surface results. If A is also zero, then the superconic becomes a simple sphere. The coefficients A,  $U_1$ , and  $V_1$  together form a Cartesian oval. These properties make the superconic stable when optimizing for the coefficients.

The superconic can be used to model surfaces which otherwise would require aspheric terms of very high order. ZEMAX can model superconics with up to 198 terms, but in practice designs rarely use more than 5 terms in the expansion.

The superconic surface uses none of the parameter terms. The extra data terms used are defined in the following table.

#### EXTRA DATA DEFINITIONS FOR SUPERCONIC SURFACES

Extra Data Number	Description
1	Maximum term number. The maximum is 198, but 4-10 is typical.
2	$U_1$
3	$V_1$
even n	$\frac{U_n}{2}$
odd n	$\frac{V_{n-1}}{2}$

### The extended Fresnel surface

This feature is only available in the EE edition of ZEMAX.

The Fresnel surface described earlier in this chapter models a plane surface which has the refractive (or reflective) power of a curved surface. The extended Fresnel surface increases the flexibility of this model by supporting a plane, spherical, conic, or polynomial aspheric substrate on which a plane, spherical, conic, or polynomial Fresnel pattern is etched.

The surface sag is identical to the even aspheric surface:

$$z_s = \frac{c_s r^2}{1 + \sqrt{1 - (1 + k_s)c_s^2 r^2}} + \alpha_1 r^2 + \alpha_2 r^4 + \alpha_3 r^6 + \alpha_4 r^8 + \alpha_5 r^{10} + \alpha_6 r^{12} + \alpha_7 r^{14} + \alpha_8 r^{16} .$$

See the section on even aspheric surfaces earlier in this chapter for details. The previous expression is used only to compute the ray-surface intercept. Once the intercept is found, the refraction (or reflection) of the surface is determined by the local slope of the Fresnel facets, which depends upon both the Fresnel power and the substrate power. The Fresnel power is given by the gradient of the Fresnel sag, described by an expression virtually identical to the even asphere sag expression:

$$z_F = \frac{c_F r^2}{1 + \sqrt{1 - (1 + k_F)c_F^2 r^2}} + \sum_{i=0}^n \alpha_i r^{2i} .$$

The only difference is that the latter expression need not use all 8 terms if n is less than 8, the maximum accepted value. Note that the curvature (symbol c), the conic constant (symbol k), and all of the polynomial coefficients (symbol  $\alpha$ ) are independent for the substrate sag and Fresnel portions of the surface. The refraction at the surface accounts for both the substrate sag and the Fresnel sag, while the ray-surface intercept depends only upon the substrate sag.

The substrate sag radius, conic, and polynomial terms are all specified in the Lens Data Editor, just like the even asphere surface. The Fresnel sag terms are specified in the Extra Data Editor. However, the Extra Data Editor uses curvature (the reciprocal of radius) rather than radius for the Fresnel sag. The Extra Data Values used are summarized in the following table.

EXTRA DATA DEFINITIONS FOR EXTENDED FRESNEL SURFACES

Extra Data Number	Description
1	Maximum aspheric polynomial term number, n. The maximum is 8.
2	The curvature (NOT RADIUS) of the Fresnel surface. This parameter will effect the refraction of the surface, but not the shape of the substrate.
3	The conic constant of the Fresnel surface. This parameter will effect the refraction of the surface, but not the shape of the substrate.
4	Coefficient in radians on $r^2$
n+3	Coefficient in radians on $r^{2n}$

The "Maximum aspheric polynomial term number" is used to specify the maximum polynomial term to be used in calculating the surface phase. This number is provided simply to speed the ray tracing calculation, as terms beyond this number are ignored. As with any complex surface model, extreme care should be taken to evaluate the accuracy and appropriateness of this model, especially where fabrication decisions are concerned.

Because there is no reliable way to compute the phase through a Fresnel surface which is not a plane, any calculation that requires OPD data, such as OPD fans, MTF, and Zernike coefficients, will not be supported if a non-plane substrate Fresnel surface is present in the lens description.

## The grid sag surface

This feature is only available in the EE edition of ZEMAX.

The grid sag surface has a shape defined by a base plane, sphere, conic asphere, or polynomial asphere plus additional sag terms defined by a rectangular array of sag values. The surface shape is determined by a bicubic spline interpolation of the sag values. The sag can be described at discrete points by

$$Z_{base} = \frac{c r^2}{1 + \sqrt{1 - (1 + k)c^2 r^2}} + \sum_{i=1}^8 \alpha_i r^{2i},$$

$$Z_{ij} = Z_{base} + Z(x_i, y_j), \text{ where}$$

$$x_i = delx*(i-ic), \quad y_j = dely*(j-jc), \text{ and}$$

$$ic = \frac{n_x + 1}{2}, \quad jc = \frac{n_y + 1}{2}, \quad 1 \leq i \leq n_x, \quad 1 \leq j \leq n_y,$$

where delx and dely are the spacings between grid points in the x and y directions, and nx and ny are the number of points in the grid in each direction. The nx and ny values must be odd, so that one grid point lies exactly at the coordinates x = 0, y = 0. Furthermore, the sag value at this grid point must always be zero. The terms defining  $Z_{base}$  are identical to those defined for the even asphere surface.

In addition to the sag data, ZEMAX requires the first derivatives in each direction and the cross derivative at each point in the grid. All of this data must be computed and tabulated outside of ZEMAX, arranged in the proper file format, and then read using the Extra Data Editor "Import" feature. The proper file format is as follows:

```
nx ny delx dely unitflag
z dz/dx dz/dy d2z/dxdy
.
.
.
```

The first line in the file contains five values, which define the number of points in the x and y directions, the increment in the x and y directions, and an integer unit flag indicating the units of the data: 0 for mm, 1 for cm, 2 for in, and 3 for meters. If the unit flag is not included, no scaling of the data is performed. If the unit flag is provided, any required scaling to the current lens units will be performed. Note that sag and cross derivative values have dimensions, and are therefore scaled, but first derivative values are dimensionless, and are not scaled.

The remaining  $nx \times ny$  lines of the file contain four numbers each: the sag, the x derivative of the sag, the y derivative of the sag, and the cross derivative  $d/dxdy$ . These values are required to insure smooth bicubic interpolation of the sag over the boundary points. The first line in the file corresponds to the upper left corner of the surface, and each point that follows is read across the face of the surface from left to right. After  $nx$  points, the  $nx+1$  point is read in as the first value in row 2, and so on until  $nx \times ny$  points have been read in. The file must be in ASCII and end in the extension .DAT.

To use the grid sag surface, first change the surface type to grid sag. Then from the Extra Data Editor, choose, Tools, Import, and specify the correct surface number and DAT file name. The data will be loaded into memory, but will not be displayed on the Extra Data Editor. The reason for not displaying the data is the potentially large number of points. The grid size is only limited by available memory. Each point in the grid requires 4 8-byte double precision values and 1 1-byte value, or 33 bytes. A 255 by 255 grid file would require approximately 2 Megabytes of memory. Fortunately, the bicubic interpolation algorithm is very smooth, with exact results at the grid points, and large grid files are generally not required for reasonably smooth surface shapes.

Once the grid file is loaded, the values can be checked using the surface data feature. The data is thereafter stored with the lens file, and the original DAT file is ignored. If the DAT file must be changed, the file needs to be reloaded using the import feature in the Extra Data Editor.

An example C language program, GRIDSAMP.C, shows the correct method for creating a grid DAT file; the example used is a spherical surface. The GRIDSAMP program also shows how to use finite differences to compute the derivatives when only the sag formula of a surface is known analytically.

If rays are traced beyond the defined portion of the grid, a ray miss error is returned by the ray tracing algorithm. No optimization or tolerancing of grid data is possible.

## The grid phase surface

This feature is only available in the EE edition of ZEMAX.

This surface is nearly identical to the grid sag surface. The key differences are:

- The units of sag are radians of phase instead of units of length.

- The unitflag data is only used to scale the delx, dely, and derivative values.

- The surface shape is the same as a standard surface. The radius and conic columns therefore change the surface shape.

The file format and general information provided in the grid sag description is otherwise valid.

### **Phase coefficients sign conventions**

See the section "The binary 1 surface" for a discussion of sign conventions.

## The generalized Fresnel surface

This feature is only available in the EE edition of ZEMAX.

This surface represents a generalization of the Fresnel surface described earlier in this chapter. The basic concept of a Fresnel surface, as far as ray tracing is concerned, is that the overall shape of the surface substrate is independent of the local slope. The generalized Fresnel uses a polynomial aspheric substrate model, identical to the "even aspheric" surface described earlier in this chapter. The sag of the surface is given by the following expression:

$$z = \frac{c r^2}{1 + \sqrt{1 - (1 + k)c^2 r^2}} + \alpha_1 r^2 + \alpha_2 r^4 + \alpha_3 r^6 + \alpha_4 r^8 + \alpha_5 r^{10} + \alpha_6 r^{12} + \alpha_7 r^{14} + \alpha_8 r^{16},$$

where the terms are identical to those in the even asphere surface model. See that section for a complete discussion.

Once the ray has intercepted the surface, the ray reflects or refracts as if the surface had a shape described by a polynomial expansion of the form:

$$z = \sum_{i=0}^N A_i E_i(x,y),$$

where  $N$  is the number of polynomial coefficients in the series, and  $A_i$  is the coefficient on the  $i^{th}$  extended polynomial term. The polynomial terms are identical to those described in the "extended polynomial" surface section earlier in this chapter. It is the gradient of the extended polynomial terms alone that determines the reflective or refractive properties of the surface, not the surface substrate shape.

One application for this surface type is modeling faceted surfaces. For example, a flat substrate may consist of a series of small faceted planes, which would reflect or refract the light as though the surface was tilted. This can be simulated using a flat substrate and a linear  $x$  or  $y$  tilt term in the polynomial coefficients.

Because there is no reliable way to compute the phase through a Fresnel surface which is not a plane, any calculation that requires OPD data, such as OPD fans, MTF, and Zernike coefficients, will not be supported if a non-plane substrate Fresnel surface is present in the lens description.

## The periodic surface

This feature is only available in the EE edition of ZEMAX.

This surface shape is described by the following expression:

$$z = \frac{c r^2}{1 + \sqrt{1 - (1 + k)c^2 r^2}} - A \left[ \frac{1}{4} [1 + \cos(2\pi\alpha x)] [1 + \cos(2\pi\beta y)] - 1 \right],$$

where  $A$  is the amplitude in lens units of the modulation, and  $\alpha$  and  $\beta$  are the periods of the oscillations in the  $x$  and  $y$  directions, respectively. Note this expression is the sag of a sphere plus the cosine modulation. The sag of the surface is exactly zero at the vertex, and the amplitude is the peak to valley amplitude. The periods are measured in inverse lens units.

This surface uses three parameter values, and none of the extra data values.

### PARAMETER DEFINITIONS FOR PERIODIC SURFACES

Parameter 1	Parameter 2	Parameter 3	Parameter 4-8
A	$\alpha$	$\beta$	Not used

## The toroidal hologram surface

The toroidal hologram surface shape is identical to the toroidal surface described earlier in this chapter (see that section for details). The curve in the Y-Z plane is defined by:

$$z = \frac{c y^2}{1 + \sqrt{1 - (1 + k)c^2 y^2}} + \alpha_1 y^2 + \alpha_2 y^4 + \alpha_3 y^6 + \alpha_4 y^8 + \alpha_5 y^{10} + \alpha_6 y^{12} + \alpha_7 y^{14} .$$

The radius and conic values, along with the parameter values, are used to define the surface shape. The extra data values are used to define the holographic properties of the surface. The terms used to define the holographic construction beams are identical to those described in the earlier sections on the Hologram 1 and Hologram 2 surface types. Extra data values 1-6 are the x, y, and z construction points for sources 1 and 2, value 7 is the construction wavelength, and value 8 is the playback order M.

There is also a ninth extra data value which is used as a "flag" to indicate if the two construction beams are converging or diverging. If both beams are converging or diverging to or from the construction points, then the flag value should be +1. If one beam is converging and one is diverging, then the flag should be -1. Any value less than zero is the same as -1, and any value greater than zero is the same as +1. This flag value serves to distinguish these two cases, which is the only difference between the Hologram 1 and 2 surface types.

#### PARAMETER DEFINITIONS FOR TOROIDAL HOLOGRAM SURFACES

Parameter 1	Parameter 2	Parameter 3	Parameter 4	Parameter 5	Parameter 6	Parameter 7	Parameter 8
Radius of Revolution	$\alpha_1$	$\alpha_2$	$\alpha_3$	$\alpha_4$	$\alpha_5$	$\alpha_6$	$\alpha_7$

#### EXTRA DATA DEFINITIONS FOR TOROIDAL HOLOGRAM SURFACES

Extra Data Number	Description
1	Construction coordinate X1 for first construction beam.
2	Construction coordinate Y1 for first construction beam.
3	Construction coordinate Z1 for first construction beam.
4	Construction coordinate X2 for second construction beam.
5	Construction coordinate Y2 for second construction beam.
6	Construction coordinate Z2 for second construction beam.
7	Construction wavelength.
8	Playback order.
9	Converge/Diverge flag.

### The Jones matrix surface

This feature is only available in the EE edition of ZEMAX.

This surface is used to define an arbitrary polarizing component. The surface shape is always plane. The Jones matrix modifies a Jones vector (which describes the electric field) according to

$$\begin{bmatrix} E'_x \\ E'_y \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} E_x \\ E_y \end{bmatrix},$$

where A, B, C, D,  $E_x$ , and  $E_y$  are all complex numbers. In the lens data editor, ZEMAX provides cells for defining A real, A imag, etc. Note that the z component of the electric field is unaffected by the Jones matrix surface. Only polarization analysis features consider the effects of this surface type. See the chapter "Polarization Analysis" for details. This surface uses eight parameter values, and none of the extra data values.

## PARAMETER DEFINITIONS FOR JONES MATRIX SURFACES

Parameter 1	Parameter 2	Parameter 3	Parameter 4	Parameter 5	Parameter 6	Parameter 7	Parameter 8
Ar	Ai	Br	Bi	Cr	Ci	Dr	Di

### The atmospheric refraction surface

This feature is only available in the EE edition of ZEMAX.

This surface is used to simulate the effects of refraction through the Earth's atmosphere when viewing a star or point source. The atmosphere has a small but non-zero dispersion which introduces a tilt term to the incoming wavefront which depends upon wavelength. ZEMAX uses a model based upon those described in the following publications:

P. K. Seidelmann, Ed., "Refraction - Numerical Integration", Section 3.281, Explanatory Supplement to the Astronomical Almanac, pp. 141-143, University Science Books, Mill Valley, 1992

C. Y. Hohenkerk and A. T. Sinclair, NAO Technical Note 63, Royal Greenwich Observatory Science and Engineering Research Council, 1985.

Six parameters are supplied to the model: The observed zenith angle of the source in degrees, the height of the observer above sea level in meters, the ambient temperature at the observer in Kelvin, the atmospheric pressure at the observer in millibars, the relative humidity (a number between 0.0 and 1.0), and the latitude of the observer in degrees.

ZEMAX computes the atmospheric refraction angle in radians for all defined wavelengths, then subtracts from all wavelengths the amount of refraction at the primary wavelength; thereby using the primary wavelength as a reference. Atmospheric refraction manifests itself as a small tilt in the OPD fan plot, or a slight chief ray offset similar to lateral color in the ray fan plot. All refraction is assumed to occur in the y direction only.

## PARAMETER DEFINITIONS FOR ATMOSPHERIC REFRACTION SURFACES

Parameter 1	Parameter 2	Parameter 3	Parameter 4	Parameter 5	Parameter 6
Zenith	Height	Temperature	Pressure	Humidity	Latitude

### The zone plate surface

This feature is only available in the EE edition of ZEMAX.

This surface is used to model a Fresnel Zone Plate (FZP). A FZP consists of a refractive or reflective surface upon which annular zones of varying depth are cut or etched. Typically, the spacing between grooves is large compared to the wavelength, and therefore FZP's are entirely refractive, and not diffractive, devices. The varying depth of the annular zones imparts a slight phase change to different zones of the beam, due to the change in path length through the material.

The FZP is modeled by altering the local depth of the surface. Edge effects, such as rays striking the side of the zone, and diffraction from the edges, are ignored. The FZP is defined by a normalized optical thickness of each zone. Each zone is spaced a fixed radius apart, and up to 198 zones may be defined. Three parameter values are required: the mode, the  $\Delta r$ , and the reference wavelength,  $\lambda_0$ .

If the mode is zero, then the steps are not interpolated; each zone has an optical thickness equal to the optical thickness value of the zone with the next largest radius. This results in discontinuous phase steps, like a staircase. If the mode is unity, then the optical thickness is interpolated between adjacent zones. Although the phase will be continuous, the first derivative of the phase will generally be discontinuous at the zone boundaries. The " $\Delta r$ " is the step in radius between adjacent zones in lens units. The "reference wavelength" is the wavelength in microns used to normalize the optical thickness. For example, if the FZP defines the boundary between a material with index 1.55 and air, with index 1.00, and the reference wavelength is .45 microns, then an optical thickness of 1 wave corresponds to a groove depth of



$$T = \frac{\lambda_0}{n_2 - n_1} = \frac{0.45 \mu}{0.55} = 0.818182 \mu .$$

This thickness is the actual groove depth if the optical thickness is given to be 1.0. ZEMAX uses this normalized optical thickness for convenience. To specify a zone introduces one-half wave of phase at the reference wavelength, simply enter an optical thickness of 0.5. At other wavelengths, the phase will be greater or less depending upon the dispersion of the two adjacent media.

The optical thickness is defined at n points, separated by  $\Delta r$ :  $r_1, r_2, r_3, \dots r_n$ . The number of points, "n", is specified in the first column of the Extra Data Editor. The remaining terms are defined in columns 2 through n+1 in the Extra Data Editor.

#### PARAMETER DEFINITIONS FOR ZONE PLATE SURFACES

Parameter 1	Parameter 2	Parameter 3	Parameter 4-6
Mode	$\Delta r$	Ref. Wave	Not Used

#### EXTRA DATA DEFINITIONS FOR ZONE PLATE SURFACES

Extra Data Number	Description
1	Number of points
2	Optical thickness at $\Delta r$
3	Optical thickness at $2 \Delta r$
4	Optical thickness at $3 \Delta r$
n+1	Optical thickness at $n \Delta r$

### The user defined surface

This feature is only available in the EE edition of ZEMAX.

The user defined surface (UDS) is an exceptionally powerful, flexible, and fast way of implementing surfaces not already built into ZEMAX. UDS's may be any shape, have any refractive, reflective, or diffractive properties, may impart any arbitrary phase to the beam, and may be followed by either homogeneous media or by a gradient index media of arbitrary form.

The secret to this great flexibility is that all the properties of the surface are defined by the user in a separate C or C++ program compiled and linked into ZEMAX dynamically using the Windows Dynamic Link Library (DLL) capability. The DLL must contain functions which return to ZEMAX all the data required to draw the surface, trace rays, compute refraction angles, and for gradient index, to determine the index as a function of position within the media following the surface.

Because the DLL is compiled code, and because ZEMAX passes to the DLL a pointer to the data that needs to be computed, UDS's are very fast; nearly as fast as native ZEMAX code (there is a small amount of overhead due to the function call).

The power of the UDS comes at a price, although it is a reasonable one. The UDS does require that the user have a suitable compiler or development tool that can generate 32 bit Windows compatible DLL's. It is also assumed that the user can write the required code, and most importantly, insure that the code is reliable and bug-free. To maximize speed, ZEMAX performs very little error checking on data returned by the DLL, and so buggy UDS DLL's are quite capable of bringing ZEMAX to a crash.

*For this reason, technical support on the implementation of UDS's is strictly limited to illustrating that the provided sample files work correctly.* If you need a UDS DLL, and do not possess the desire or ability to write them yourself, please feel free to contact FSI for a quote on developing a custom DLL to meet your requirements. FSI has considerable experience in developing ray tracing algorithms, and can generally write DLL's at very competitive rates on very short notice.

## **The UDS DLL**

The best way to learn how to create a UDS DLL is to start by copying a sample DLL to a new file name, and then edit the sample to suit your requirements. A good portion of the DLL source code is "boiler plate" which is common to all DLL's.

There are three files that are required to define a UDS DLL: a C (or C++) source code file such as MY\_SURF.C, a header file called USERSURF.H which is already defined for you, and a definition file called MY\_SURF.DEF. The DEF is identical for every DLL, only the name changes to match the C file name. Clearly, you can call DLL's anything you want; MY\_SURF is just a sample name.

Only the C file needs to be modified. The file is essentially 2 functions; the first is "DllEntryPoint" which is a "boiler plate" name, and never needs to be modified. The other function is "UserDefinedSurface", and this is the function that is modified for each surface type. The function name cannot be changed; ZEMAX looks for a function of this name. This is also the purpose of the DEF file, it simply states that the function "UserDefinedSurface" is "exported" which means ZEMAX can execute the function.

ZEMAX passes two structures to the function, and these structures are defined in the USERSURF.H file. Inside the UserDefinedSurface function is a C "switch-case" construct. ZEMAX passes a "type" parameter, which is a number between 0 and 7, inclusive, that indicates what data ZEMAX needs computed and returned to ZEMAX. The type codes are described in the following table:

UDS DLL Type Codes

TYPE	What ZEMAX wants the DLL to compute
0	The surface name, radial symmetry status, and grin status
1	The name of the parameter columns
2	The name of the extra data columns
3	the sag and alternate sag of the surface
4	A paraxial ray trace to and refraction through the surface
5	A real ray trace to and refraction through the surface
6	The index and first derivatives for gradient index propagation
7	The default data when the user first selects the surface type

Extensive comments are provided in the sample DLL source code files. The DLL files can be found in the directory \ZEMAX\DLL, and any new DLL files must be placed there as well.

### **Comments about refractive and reflective UDS DLL's**

For conventional homogeneous surfaces which are reflective or refractive, but not gradient or diffractive, start with the file US\_STAND.C. That file is the source code for US\_STAND.DLL, which is a clone of the built-in ZEMAX standard surface. It is not exactly the same code, and is somewhat slower than the form of equations used by ZEMAX, however, it is functionally equivalent. US\_STAND.C includes a generic Snells' law refraction function which also works for reflection. The DLL must provide the sag, ray intercept equations, surface normals, optical path, and paraxial refraction functionality.

### **Comments about gradient index UDS DLL's**

Gradient surfaces should use US\_GRIN1.C as a starting point. The code is very similar to US\_STAND.C, except a flag is returned indicating that the surface is a gradient type, and data type "6" is implemented to return the index of refraction given X, Y, Z, and all the parameter and extra data. The first derivative of the index in the X, Y, and Z directions must also be provided.

### **Comments about diffractive UDS DLL's**

Diffractive optics are very similar to standard optics, except the rays are further deviated by the derivative of the phase as a function of X and Y, and the optical path length needs to be modified to include the phase change.

Generally, "refraction" for diffractive optics is defined by:

$$l' = l + \frac{\lambda}{2\pi} \frac{\partial \phi}{\partial x}$$

$$m' = m + \frac{\lambda}{2\pi} \frac{\partial \phi}{\partial y}$$

where  $l$  and  $m$  are the direction cosines, and the  $z$  direction cosine,  $n$ , is computed to make the magnitude of the direction vector unity, and  $\phi$  is the phase in radians.

A sample DLL, US\_GRATE.DLL, illustrates the diffractive computation by cloning the ZEMAX grating surface.

### **Lenslet arrays using UDS DLL's**

Lenslet arrays are easily modeled using the user defined surface. Basically, the ray trace determines which segment of the array is struck, then uses the local lens curvature to determine the refraction. The source code and DLL are provided as US\_ARRAY.C and US\_ARRAY.DLL, respectively.

### **Error handling and UDS's**

The convention ZEMAX uses internally is to return a zero value if the DLL computed a meaningful result, and no error occurred. Otherwise, the DLL should return -1.

The exception is when ray tracing, either paraxial or real. If the ray misses the surface, it should return the surface number. If the ray total internally reflects (TIR's) then the return value should be the negative of the surface number.

ZEMAX uses these error codes to help provide meaningful diagnostics to both the user and various ZEMAX features.

This page intentionally left blank.

## Introduction

Solves are functions which actively adjust specific values. Solves can be specified on curvatures, thicknesses, glasses, semi-diameters, conics, and parameters. Solves are set by double clicking on the cell you wish to place the solve on. The solve type control screen appears. Select the solve type from the drop-down list. Some solves require one or two additional parameters. The available solves and required parameters are summarized in the following table. Parameters may also have "fixed" or "variable" status; these are not solves, however, they are listed for completeness. To turn solves off, select the "FIXED" option from the drop-down list. Note that this simply turns off the solve, it does not change the last value. Pressing Ctrl-V twice with the cursor on the solved parameter will also remove any solve, and this is the quicker way to do so.

### SUMMARY OF SOLVES

Solve type	First parameter	Second parameter	Code
Curvature : Fixed			
Curvature : Variable			V
Curvature : Marginal ray angle	Angle		M
Curvature : Chief ray angle	Angle		C
Curvature : Pick up	Surface	Scale factor	P
Curvature : Marginal ray normal			N
Curvature : Chief ray normal			N
Curvature : Aplanatic			A
Curvature : Element power	Power		X
Curvature : Concentric with surface	Surface to be concentric to		S
Curvature : Concentric with radius	Surface to be concentric with		R
Curvature : F/#	Paraxial F/#		F
Thickness : Fixed			
Thickness : Variable			V
Thickness : Marginal ray height	Height	Pupil zone	M
Thickness : Chief ray height	Height		C
Thickness : Edge thickness	Thickness	Radial height (use zero for semi-diameter)	E
Thickness : Pick up	Surface	Scale factor	P
Thickness : Optical path difference	OPD	Pupil zone	O
Thickness : Position	Surface	Length from surface	T
Thickness : Compensator	Surface	Sum of surface thicknesses	S
Thickness : Center of curvature	Surface to be at the COC of		X
Glass : Fixed			
Glass: Model			
Glass : Pick up	Surface		P
Glass: Substitute			S
Semi-Diameter : Automatic			
Semi-Diameter : Fixed			U
Semi-Diameter : Pick up	Surface		P
Semi-Diameter : Maximum			M

Solve type	First parameter	Second parameter	Code
Conic : Fixed			
Conic : Variable			V
Conic : Pick up	Surface	Scale factor	P
Parameter : Fixed			
Parameter : Variable			V
Parameter : Pick up	Surface	Scale factor	P

## Curvature : Marginal ray angle

---

The effective focal length of the lens can be controlled by placing a paraxial marginal ray angle solve on the curvature of the last surface before the image. For example, suppose a lens has a 20 mm entrance pupil diameter, and it is desired to constrain the effective focal length to 100 mm. This requires a marginal ray exit angle from the last surface of  $-.1$  (this number is 20 divided by 2 divided by 100, the minus sign is because the ray is converging, or going down toward the image plane). See the F number solve.

## Curvature : Chief ray angle

---

The chief ray angle solves work the same way as the marginal ray angle, except the paraxial chief ray is used for the calculations instead. The chief ray solve is useful for maintaining a particular magnification or for maintaining collimation. The letter "C" will appear in the curvature cell to indicate that a chief ray angle solve is active on that surface.

## Curvature : Pick up

---

Another useful curvature solve is the 'pick up' solve. Specifying a surface number tells ZEMAX to insert at this surface the value from some other surface. An optional scale factor allows the value to be modified. This is useful for example in systems where the light passes through the same optics twice, such as a null lens. This pickup feature allows variables to be coupled together, and they will change under the influence of other solves, editing, or optimization, in accordance with the hierarchy described below.

Pickup solves should only be used when picking up data from previous surfaces; picking up data from surfaces which follow the solve will lead to unpredictable results. The letter "P" will appear in the curvature cell to indicate that a pickup solve is active on that surface.

## Curvature : Marginal ray normal

---

This solve will force the surface to be normal to the paraxial marginal ray. This is also called an image-centered surface. These special surfaces introduce no spherical aberration or coma. The letter "N" will appear in the curvature cell to indicate that a normal ray angle solve is active on that surface.

## Curvature : Chief ray normal

---

This solve will force the surface to be normal to the paraxial chief ray. This is also called a pupil-centered surface. These special surfaces introduce no coma, astigmatism, or distortion. The letter "N" will appear in the curvature cell to indicate that a normal ray angle solve is active on that surface.

## Curvature : Aplanatic

---

This solve will force the surface to be aplanatic with respect to the paraxial marginal ray. These special surfaces introduce no spherical aberration, coma, or astigmatism. The letter "A" will appear in the curvature cell to indicate that an aplanatic solve is active on that surface.

## Curvature : Element power

---

The power of an element is given by:

$$\phi = c_1(n_2 - n_1) + c_2(n_3 - n_2) - c_1(n_2 - n_1) * c_2(n_3 - n_2) * \frac{t_2}{n_2}.$$

This solve adjusts the value of  $c_2$ , to maintain the specified element power. The solve is assumed to be on the second of two adjacent surfaces. This solve is ignored if the surface number is less than 2, or if  $n_3$  and  $n_2$  are identical at the primary wavelength.

### **Curvature : Concentric with surface**

---

This solve will force the curvature of the surface such that the surface is concentric about the specified surface. The specified surface must precede the surface on which the solve is placed. The letter "S" will appear in the cell to indicate that the surface concentric solve is active on that surface.

### **Curvature : Concentric with radius**

---

This solve will force the curvature of the surface such that the surface is concentric about the same point that the specified surface is concentric to. The specified surface must precede the surface on which the solve is placed. The letter "R" will appear in the cell to indicate that the radius concentric solve is active on that surface.

### **Curvature : F number**

---

This solve will force the curvature of the surface such that the marginal ray angle exiting the surface is  $-1/2F$  where  $F$  is the paraxial  $F/\#$ . The letter "F" will appear in the cell to indicate that the  $F/\#$  solve is active on that surface.

### **Thickness : Marginal ray height**

---

The most common thickness solve is the marginal ray height solve, which can be used to constrain the image plane to the paraxial focus. To move the image plane to the paraxial focus, double click on the thickness of the last surface before the image surface, and select "Marginal Ray Height" from the drop-down list which appears. Since the marginal ray has a height of zero (it crosses the axis) at the paraxial image plane (assuming a rotationally symmetric system), we would like to set this last thickness to bring the image surface to the point where the marginal ray height is zero. The default height value is zero, which is desired, and so simply click on OK to exit the screen, and the solve immediately adjusts the thickness appropriately.

In principle, the solve could have been set for some other marginal ray height by entering a value other than zero for height, the first optional parameter. Note that any surface can have a marginal ray height solve, not just the thickness before the image plane. The "Height" is the height of the marginal ray at the next surface (again, not necessarily the image surface).

The third value, "Pupil Zone", allows the ray pupil coordinate to be defined. The default is zero, which indicates that a paraxial ray should be used. Any non-zero value indicates that a real marginal ray is to be used. The zone value must be between -1 and 1. This is the  $P_y$  coordinate, or the normalized entrance pupil coordinate in the  $y$  direction. This solve can be used to constrain that a particular ray, such as the .7 zonal ray, to have zero transverse aberration on axis. The letter "M" will appear in the thickness cell to indicate that a marginal ray height solve is active on that surface.

### **Thickness : Chief ray height**

---

This is similar to the marginal ray height solve, except the paraxial chief ray is used. This solve is useful for locating a surface at a pupil plane. The letter "C" will appear in the thickness cell to indicate that a chief ray height solve is active on that surface.

### **Thickness : Edge thickness**

---

This solve dynamically adjusts the spacing between two surfaces to maintain a specified distance between the surfaces at a specified radial aperture. This is useful for preventing negative, or overly sharp edges on elements. The letter "E" will appear in the thickness cell to indicate that an edge thickness solve is active on that surface. If the radial aperture is set to zero, then the current semi-diameter will be used. See the chapter "Conventions and Definitions" for an important discussion on the definition of edge thickness.

## **Thickness : Pick up**

---

Pickup solves also work with thicknesses, see the "Curvature : Pick up" section for a complete description.

## **Thickness : Optical path difference**

---

This solve will actively adjust the thickness to maintain a specific optical path difference at a specific pupil coordinate. The two parameters to be set are the OPD in primary waves, and the pupil zone at which to evaluate the OPD. For example, to maintain the focal position so that the real marginal ray has the same optical path length as the real chief ray, define an OPD solve on the last thickness before the image plane. Set the OPD parameter to zero, the pupil zone to 1.0, and press escape. The letter "O" will appear to indicate the OPD solve is active. Perform an OPD plot and verify that the OPD is in fact zero at the pupil edge. Only the primary wavelength is used, and only the on-axis field is considered. More complicated OPD constraints are available by using optimization, described in a separate chapter. The letter "O" will appear in the thickness cell to indicate that an OPD solve is active on that surface.

## **Thickness : Position**

---

The position solve maintains the distance from a specified reference surface. If the reference surface precedes the surface on which the solve is placed, then the sum of the thicknesses from the reference surface to the surface following the solve surface will be maintained at the specified value. If the reference surface follows the surface on which the solve is placed, then the sum of the thicknesses from the solve surface to the reference surface will be maintained at the specified value. If the reference surface is the same as the solve surface, then the thickness of the solve surface will be set at the solve length value.

The position length solve is particularly useful for maintaining the length of a portion of a zoom lens to a fixed value. The solve can also be used to meet a total lens length constraint. In either case, the solve may eliminate optimization variables and operands, enhancing optimization convergence and speed. The letter "T" will appear in the thickness cell to indicate that a total length solve is active on that surface. See the chapter "Advanced topics" for a discussion of zoom lens design using this solve.

## **Thickness : Compensator**

---

The thickness compensator solve is very similar to the position solve. This solve maintains the thickness on a surface so that the sum of the surface thickness and the thickness on another "reference" surface equals a constant. In equation form, this solve maintains the condition  $T = S - R$ , where S is the sum of the thicknesses of the two surfaces, and R is the thickness of the reference surface. The reference surface must precede the surface with the solve.

## **Thickness : Center of curvature**

---

This solve will adjust the thickness of the surface to place the following surface at the center of curvature of any prior surface. The letter "X" will appear in the cell to indicate that the center of curvature solve is active on that surface.

## **Glass : Model**

---

The model glass state is not a solve. The model glass is used to idealize a glass by describing the dispersion using three values: the index of refraction at d-light, the Abbe number, and a term describing the partial dispersion. For more information, see the Chapter "Optimization".

## **Glass : Pick up**

---

Pickup solves also work with glasses, see the "Curvature : Pick up" section for a complete description.

## **Glass : Substitute**

---

The substitute state is not a solve. If the glass solve type is set to "Substitute", then the global optimization algorithms are permitted to change the glass type during optimization. This is similar to a variable status in that the glass type can change. See the Chapter "Global Optimization" for details. To prevent certain glasses from



being chosen during optimization, choose "Exclude Substitution" for the glass that should be avoided on the glass catalog dialog box; see the Chapter "Using Glass Catalogs" for details.

## **Semi-Diameter : Pick up**

---

Pickup solves also work with semi-diameters, see the "Curvature : Pick up" section for a complete description.

## **Semi-Diameter : Maximum**

---

The maximum solve is used for setting the semi-diameter value to the maximum required across multiple configurations. For example, a zoom lens which has three configurations will generally have three different semi-diameter values for every surface that is using "automatic" semi-diameters. The maximum solve will compute the semi-diameter for each configuration, and then use the largest of the values.

## **Conic : Pick up**

---

Pickup solves also work with conics, see the "Curvature : Pick up" section for a complete description.

## **Parameter : Pick up**

---

Pickup solves also work with all 8 Parameters, see the "Curvature : Pick up" section for a complete description.

## **Suggestions for use**

---

There is a hierarchy by which solves are computed. The solves are computed sequentially from the first surface to the image surface, in this order: curvature, thickness, glass, semi-diameter, conic, and then parameters.

Because curvature and thickness solves can affect the entrance pupil position, ZEMAX does not allow solves which rely upon ray tracing, such as marginal ray height solves, to be placed before the stop surface. This will avoid ambiguous or incorrect settings for the solved parameters. There is no way to predict the results of solves placed before the stop. A similar problem arises when a marginal ray solve is used on the last surface to control the focal length - the aperture definition must be entrance pupil diameter (rather than F/#) to keep the solution unique.

Solves are highly efficient, and should be used when possible instead of variables during optimization. For example, it is usually better to place a curvature solve on the last surface to control effective focal length than to optimize for focal length explicitly.

This page intentionally left blank.

## Introduction

---

The optimization feature provided by ZEMAX is quite powerful, and is capable of improving lens designs given a reasonable starting point and a set of variable parameters. Variables can be curvatures, thicknesses, glasses, conics, parameter data, extra data, and any of the numeric multi-configuration data. ZEMAX uses an actively damped least squares method. The algorithm is capable of optimizing a merit function composed of weighted target values; these target values are called "operands". ZEMAX has several different default merit functions, described in a subsequent section. These merit functions can be changed easily using the Merit Function Editor. For details on this procedure see the section "Modifying the merit function".

Optimization requires three steps: 1) a reasonable system which can be traced, 2) specification of the variables, and 3) specification of the merit function. A reasonable system is a rather loose concept which simply means that poorly conceived designs are not likely to be transformed into exceptional designs by the optimization algorithm (although there are exceptions). The variables, and there must be at least one for the optimization algorithm to be able to make any progress, are specified on the various editors, as described in the next section. To reach the optimization screen, select Tools, Optimization. You must specify all variables before using optimization.

The algorithm used by the optimization feature described in this chapter is designed to find the "local" minimum of the specified merit function. However, ZEMAX-XE and EE also have a capability to search for a "global" minimum of the merit function. The global minimum is the lowest possible value for the merit function, and if the merit function is selected appropriately, this implies the best possible solution to the problem. The global optimization feature is not for novice users, and is not appropriate for interactive designing. For details see the chapter "Global Optimization".

## Selecting variables

---

Variables for optimization are specified by pressing Ctrl-V when the highlighted bar is on the parameter to be varied in the Lens Data Editor. Note that Ctrl-V is a toggle. The Multi-Configuration and Extra Data Editors also contain numeric data that may be made variable by using Ctrl-V. Glasses cannot be made variable directly because they are discrete. To optimize glasses, see "Optimizing glass selection" later in this chapter.

## Defining the default merit function

---

The merit function is a numerical representation of how closely an optical system meets a specified set of goals. ZEMAX uses a list of operands which individually represent different constraints or goals for the system. Operands represent goals such as image quality, focal length, magnification, and many others.

The merit function is proportional to the square root of the weighted sum of the squares of the difference between the actual and target value of each operand in the list. The merit function is defined this way so a value of zero is ideal. The optimization algorithm will attempt to make the value of this function as small as possible, and so the merit function should be a representation of what you want the system to achieve. You do not have to use the default merit function, you may construct your own as described in a later section.

The easiest way to define a merit function is to select the Tools, Default Merit Function option on the Merit Function Editor menu bar. A dialog box will appear which allows selection of options for the default merit function. Each option is explained in the following paragraphs.

### **Selecting the type of optimization**

Several different types of merit functions are available. The default merit function is constructed using four key choices: The optimization type, data type, reference point, and integration method. The choices are described in the following tables.

## DEFAULT OPTIMIZATION TYPES

Name	Description
RMS	RMS is an abbreviation for Root-Mean-Square. This type is by far the most commonly used. The RMS is the square root of the average value of the squares of all the individual errors.
PTV	PTV is an abbreviation for Peak To Valley. There are rare cases where the RMS is not as important as the maximum extent of the aberrations. For example, if all the rays need to land within a circular region on a detector or fiber. In these cases, the Peak-To-Valley (PTV) may be a better indicator of performance. This merit function type attempts to minimize the PTV extent of the errors.

## DEFAULT OPTIMIZATION DATA

Name	Description
Wavefront	Wavefront is the aberration measured in waves.
Spot Radius	The radial extent of the transverse ray aberrations in the image plane.
Spot X	The x extent of the transverse ray aberrations in the image plane.
Spot Y	The y extent of the transverse ray aberrations in the image plane.
Spot X and Y	Both the x and y extent of the transverse ray aberrations in the image plane. The x and y components are considered separate, and both are optimized together. This is similar to Spot Radius, except the signs of the aberrations are retained, which yields better derivatives. Note that in computing the radius of an aberration, the sign information is lost.

## DEFAULT OPTIMIZATION REFERENCE POINTS

Name	Description
Centroid	The RMS or PTV computation of the "data" is referenced to the centroid of all the data coming from that field point. <b>Centroid optimization is preferred in virtually all cases, especially for wavefront optimization.</b> For wavefront optimization, reference to the centroid subtracts out piston, x-tilt, and y-tilt of the wavefront, none of which degrade image quality. Reference to the chief ray does not subtract out wavefront piston or tilt, which leads to sub-optimal optimization. Centroid optimization also yields more meaningful results when coma is present, since coma shifts the image centroid away from the chief ray location. Historically, the chief ray has been used because it is simpler to compute, but this complexity has been absorbed into ZEMAX with virtually no performance loss.
Chief	The RMS or PTV computation of the "data" is referenced to the chief ray at the primary wavelength.

Note that a very broad range of default merit functions may be constructed. There are 2 types, 5 data choices, and 2 reference points, for a total of 20 different default merit functions.

### **Physically significant merit functions**

Note that the numerical value of the merit function is physically significant. If the merit function is RMS-Wavefront-Centroid, then the numerical value of the merit function is the RMS wavefront error in waves. If the merit function is RMS-Spot Radius-Chief, then a value of 0.145 means the RMS spot size is 0.145 lens units. If the lens units were millimeters, this would correspond to 145 microns RMS.

Novice designers often ask why the RMS spot radius merit function should yield a different optimum design than the RMS wavefront merit function. The basic reason they are different is that ray aberrations are proportional to the derivative of the wave aberrations. Therefore, it is unreasonable to expect that the minimum of one corresponds to the minimum of the other. A general rule of thumb to use is that if the system is close to diffraction limited (say a PTV wavefront error of less than two waves) then use wavefront error. Otherwise, use spot radius.

Generally speaking, the merit functions which use the centroid as a reference are superior to those that use the chief ray. Most diffraction based performance measures, such as MTF or encircled energy, improve when the RMS wavefront error referenced to the centroid decreases. However, it is always best to reoptimize a final design with the various merit functions to verify which one provides the best performance for the system being designed. For example, the RMS wavefront centroid reference often yields better low frequency MTF response, but worse medial frequency response, than the RMS chief ray reference optimization does.

### **Selecting the pupil integration method**

There are two different pupil integration methods used to construct the merit function: Gaussian quadrature (GQ) or rectangular array (RA). The GQ algorithm is VASTLY superior for almost all cases of practical interest. The GQ algorithm uses a carefully selected and weighted ray set to accurately compute the RMS or PTV error over a uniformly illuminated entrance pupil (strictly speaking, the PTV algorithm is not a GQ algorithm, but it is very similar). The weighting for all rays is applied according to the weights set on the wavelength and field data dialog boxes and by the GQ merit function algorithm. For RMS merit functions, the weighting and ray set selection used is based on a method described in a paper by G. W. Forbes, JOSA A 5, p1943. For the PTV merit functions, the ray set is based on solutions to the Chebyshev polynomials, described in Numerical Recipes, Cambridge University Press (1989). If you are interested in detailed information on the basis and accuracy of these methods, see these references. GQ is much, much more accurate than any other known method, and requires fewer rays. Therefore, you get the best of both worlds: greater speed and greater accuracy. The GQ algorithm requires specification of the number of "Rings" and the number of "Arms", and these terms are defined in subsequent paragraphs. The only drawback to GQ is that it does not work if there are surface apertures in the optical system, or if the pupil is not uniformly illuminated. For these systems, GQ is probably no longer a better choice than RA. GQ does work fine when used with vignetting factors, since the ray pattern is simply redistributed.

The RA algorithm traces a grid of rays through the pupil. The "Grid" size determines the number of rays traced, and is described in a subsequent paragraph. The "Delete Vignetted" option (also described later) allows the vignetted rays to be deleted from the ray set. Vignetted rays in this context are those rays clipped by surface apertures, not rays which have been altered by the use of vignetting factors (see the chapter "Conventions and Definitions"). The advantage to the RA algorithm is the ability to accurately account for the effects of vignetting in the merit function. This is useful in systems such as obscured telescopes and camera lenses which intentionally clip troublesome rays. The disadvantage to the RA algorithm is speed and accuracy. Usually, more rays are required to achieve a given degree of accuracy than the GQ algorithm. The bottom line: don't use RA unless you are using surface apertures.

### **Rings**

The "Rings" setting is only used in the GQ algorithm. It determines how many rays are traced at each field and at each wavelength. For on-axis fields (zero degrees field angle in a rotationally symmetric system), the number of rays is equal to the number of rings. For all other fields in symmetric systems, the number of rays traced per ring is equal to half the number of "arms" (defined in the next paragraph). Only half the rays are traced because the left-right symmetry of the system is exploited. Each set of rays is traced for each defined wavelength. For example, if you have one on-axis field, two off-axis fields, three wavelengths, and four rings selected, the number of rays traced is  $3 * (4 + 4*3 + 4*3) = 84$ . For systems without rotational symmetry, the number of rays per ring is the number of "arms" independent of field. In the prior example, this means  $3 * 3 * 4 * 6 = 216$  rays. ZEMAX automatically calculates these numbers for you; the only reason it is described here is so you will understand how the default merit function is defined. Optimization runs are longer if more rays are traced.

### **Arms**

The "Arms" setting is also only used in the GQ algorithm. It determines how many radial arms of rays in the pupil are traced. By default six equally spaced (in angle) arms are traced (or three if the system is rotationally symmetric). This number may be changed to eight, ten, or twelve. For most common optical systems, six is sufficient.

You should select the number of rings and the number of arms according to the order of aberrations present in your system. A simple way of determining the correct number of rings is to select the minimum number, one. Then go to the optimization dialog box and note the merit function. Now go back to the default merit function tool, and select two rings. If the merit function changes by more than a few percent, go back and select three, and so on until the merit function does not change significantly (perhaps 1%). Repeat the procedure for the number of arms (six arms is almost always plenty). Selecting more rings or arms than required will not improve the optimization

performance, it will only slow the algorithm down needlessly. Tracing more rays than required will not help you find better solutions!

Selecting more rings or arms than required will not improve the optimization performance, it will only slow the algorithm down needlessly.

### **Grid**

The "Grid" is only used by the RA algorithm, and the value determines the number of rays to be used. The grid size can be 4x4 (16 rays per field per wavelength), 6x6 (36 rays per field per wavelength) etc. Rays on the grid are automatically omitted if they fall outside the entrance pupil, so the actual number of rays used will be lower than the grid size squared. Selecting a larger grid size generally yields more accurate results at the expense of slower execution. However, there may be an advantage in choosing a large grid density, and then selecting the "Delete Vignetted" checkbox (described in the next paragraph). The reason is that a large grid density will fill the pupil with rays, and then the operands which are vignetted will be deleted. The result is a reasonable number of rays which accurately reflect the aperture of the system.

### **Delete Vignetted**

The "Delete Vignetted" checkbox option is only used by the RA algorithm. If selected, then each ray in the merit function will be traced through the system, and if it is vignetted by a surface aperture, if it misses any surface, or if it is total internal reflected at any surface, the ray is deleted from the merit function. This keeps the total number of rays in the merit function to a minimum. The disadvantage is that if the vignetting changes as the design is optimized, then the merit function may have to be regenerated. It is always a better choice to use the vignetting factors and then use the GQ algorithm than to delete vignetted rays if possible. Vignetting factors can be adjusted, if required, during optimization using SVIG in the merit function.

Note that ZEMAX will attempt to trace any ray defined in the merit function regardless if that ray is vignetted or not. For example, if the chief ray height is targeted using REAY, and there is a central obscuration that vignettes the chief ray, ZEMAX will still trace the ray and use the operand results as long as the ray can be traced. ZEMAX does not check to see if defined rays are vignetted, because this introduces substantial overhead during optimization.

In general, avoid vignetting of rays by surface apertures, and use vignetting factors to shape the beam size when possible. To optimize on the fraction of unvignetted rays, a macro must be defined to perform the required computations. However, this method is very prone to stagnation during optimization because small changes in lens parameters lead to discrete changes in the merit function as rays abruptly jump from being vignetted to not being vignetted.

### **Setting Boundary values**

Boundary constraints may be automatically generated and included in the default merit function by checking "Use Boundary Values:" on. If selected, then MNCG, MXCG, and MNEG operands will be added to the merit function to constrain the minimum center thickness, maximum center thickness, and minimum edge thickness for glass surfaces, respectively. MNCA, MXCA, and MNEA operands will be added to the merit function to constrain the minimum center thickness, maximum center thickness, and minimum edge thickness for air spaces, respectively.

The automatic boundary constraint feature is meant to save some manual entry of routine boundary constraints. More complex lenses, such as those with mirrors, coordinate breaks, or multi-configurations, usually require additional boundary constraints to be added to the merit function manually.

### **Start At**

The "Start At" option is used to add the default merit function at a specific position within the Merit Function Editor operand list. ZEMAX will attempt to place the default merit function after the existing targets. The algorithm used to determine the starting pointing may be ineffective if the default merit function has been edited. To control where the default start at value will be, see the DFMS operand definition.

### **Assume Axial Symmetry**

If selected, then the default merit function will exploit the left-right and rotational symmetry of the lens when constructing and evaluating the merit function. Fewer rays will be traced, accelerating the optimization with no

loss of accuracy. In systems with coordinate breaks or non-rotationally symmetric systems, the default is unselected, which means symmetry will not be exploited. Overriding the default system symmetry is useful, however, if you are designing a lens that ZEMAX thinks is non-symmetric, but the lack of symmetry does not affect the aberrations. For example, if tilted but flat fold mirrors are present, these mirrors do not eliminate the left-right symmetry of the system, but ZEMAX will by default assume symmetry does not exist. Some gradient index surfaces also use non-symmetric index variation terms which are often zero (they are used only for tolerancing). Select the checkbox to accelerate the optimization in these cases. See also the "USYM" operand description.

If you do not understand this feature, just use the default.

### **Ignore Lateral Color**

By default, ZEMAX references all RMS or PTV computations to a common reference point for each field. All of the rays are traced for all wavelengths for each field point, and the primary wavelength chief ray or the centroid of all the rays is used as the reference point. If "Ignore Lateral color" is selected, then ZEMAX computes an independent reference point for each wavelength as well. This is useful for designing systems that intentionally divide the beam by wavelength, such as a prism or spectrometer system. This option will cause the merit function to optimize each color spot independently.

## **Pitfalls with the default merit function**

---

The default merit function is easy to set up, numerically efficient, and suitable for a large number of optimization problems. However, most optical designs require extensions or modifications to the default as the design progresses. ZEMAX offers significant flexibility in the definition of the merit function, as described in the following sections.

Note that if the field or wavelength values or weights are changed, you must reconstruct the default merit function. If you are using the RA algorithm, reconstruct the default merit function if the vignetting influence changes appreciably during optimization.

If field or wavelength values or weights are changed, reconstruct the merit function.

## **Optimization with apodized beams**

---

If no pupil apodization has been specified (see the "System" chapter for details on specifying the pupil apodization) then ZEMAX assumes uniform illumination when constructing the default merit function.

If the illumination is not uniform, then the rays in the default merit function are weighted according to the apodization factor. Because the Gaussian quadrature method is designed to work with uniform pupils, only the RA algorithm should be used if apodization is used.

Since the rays selected may be insufficient to adequately represent an apodized beam, use a large number of rays (described previously) when using apodization factors. See the chapter "Conventions and definitions" for more information about apodization.

## **Modifying the merit function**

---

The merit function can be modified by the user. To change the merit function, select Editors, Merit Function from the main menu bar. New operands can be added to the list, or others deleted, using the insert and delete keys. The current merit function value and the value of each operand can be updated by selecting Tools, Update.

Operands are set by typing the name in the first column and then filling in the remaining data fields. There are eight fields that may be required to define an operand: Int1, Int2, Hx, Hy, Px, Py, target, and weight. The Int values are integer parameters whose meaning depends upon the operand selected. Usually, Int1 is the surface indicator, and Int2 is the wavelength indicator, but not always. Not all of the operands use all of the fields provided.

For those operands that use Int1 to indicate the surface number, the parameter specifies at which surface the target should be evaluated. Similarly, the Int2 value, when used as a wavelength specifier, describes which wavelength to use. Int2 must be an integer value equal to the wavelength number. The Int1 and Int2 parameters have other uses as described later.

Many of the operands use Hx, Hy, Px, and Py; these are the normalized field and pupil coordinates (see "Normalized field and pupil coordinates" in the chapter "Conventions and definitions"). Note that ZEMAX does not check to see if the specified Hx, Hy, Px, and Py coordinates are within the unit circle. For example a pupil coordinate of (1, 1) is actually outside the entrance pupil, but you will not get an error message when tracing these rays unless the rays cannot physically be traced.

The target is the desired value of the specified parameter. The difference between the target and the value of the operand is squared, and summed over all operands to yield the merit function. The value of the target and the operand itself is unimportant in optimization, only the difference between the two. The larger the difference, the greater the contribution to the merit function.

The weight is the relative importance of that parameter. The weight can be any number from 0 up, except for the special case of -1. When the weight is zero, the optimization algorithm calculates but ignores the operand. If the weight is greater than zero, then the operand will be treated as an "aberration" to be minimized along with the merit function. If the weight is less than zero, ZEMAX will set the weight to be exactly -1, and this indicates that operand will be treated as a Lagrangian multiplier. The Lagrangian multipliers force the optimization algorithm to find a solution which exactly meets the specified constraint, regardless of the effect on the other operands.

The merit function is defined as:

$$MF^2 = \frac{\sum W_i (V_i - T_i)^2 + \sum (V_j - T_j)^2}{\sum W_i}$$

The summation over i includes only positive weighted operands, while the summation over j includes only Lagrangian multiplier operands. This convention is chosen so that adding Lagrangian multipliers to control boundary conditions has no effect on the merit function if the conditions are met.

Generally Lagrangian multipliers should not be used unless there is a compelling need to do so. Although there are exceptions, Lagrangian multipliers generally slow the optimization rate, and perform poorly if the optical system is far from the merit function minimum. Using weighted constraints is often more reliable, although some readjustment of the weights may be required between optimization calls.

Not all operands use all of the data fields. The following tables describe the available operands. The first table is a "quick reference" guide which categorizes the operands by general subject. The second table provides a detailed description of each operand (listed alphabetically) and states which operands use which data fields. Note in particular that some of the operands (such as SUMM) use the Int1 and Int2 values to indicate something other than surface and wavelength. If an operand does not use a data field, a "-" is shown.

#### OPTIMIZATION OPERANDS BY CATEGORY

Category	Related Operands
First-order optical properties	EFFL, PIMH, PMAG, AMAG, ENPP, EXPP, LINV, WFNO, POWR, EPDI, ISFN, OBSN, EFLX, EFLY
Aberrations	SPHA, COMA, ASTI, FCUR, DIST, DIMX, AXCL, LACL, TRAR, TRAX, TRAY, TRAI, OPDC, PETZ, PETC, RSCH, RSCE, RWCH, RWCE, ANAR, ZERN, TRAC, OPDX, RSRE, RSRH, RWRE, TRAD, TRAE, TRCX, TRCY, DISG, FCGS, FCGT, DISC
MTF data	MTFT, MTFS, MTFA, MSWT, MSWS, MSWA, GMTA, GMTS, GMTT
Encircled energy	DENC
Constraints on lens data	TOTR, CVVA, CVGT, CVLT, CTVA, CTGT, CTLT, ETVA, ETGT, ETLT, COVA, COGT, COLT, DMVA, DMGT, DMLT, TTHI, VOLU, MNCT, MNET, MXCT, MXET, MNCG, MNEG, MXCG, MXEG, MNCA, MNEA, MXCA, MXEA, ZTHI, SAGX, SAGY, CVOL, MNSD, MXSD, XXET, XXEA, XXEG, XNET, XNEA, XNEG, TTGT, TTLT, TTVA
Constraints on parameter data	P1VA, P1GT, P1LT, P2VA, P2GT, P2LT, P3VA, P3GT, P3LT, P4VA, P4GT, P4LT, P5VA, P5GT, P5LT, P6VA, P6GT, P6LT, P7VA, P7GT, P7LT, P8VA, P8GT, P8LT
Constraints on extra data	XDVA, XDGT, XDLT



Category	Related Operands
Constraints on glass data	MNIN, MXIN, MNAB, MXAB, MNPD, MXPDP, RGLA
Constraints on paraxial ray data	PARX, PARY, PARZ, PARR, PARA, PARB, PARC, PANA, PANB, PANC, PATX, PATY, YNIP
Constraints on real ray data	REAX, REAY, REAZ, REAR, REAA, REAB, REAC, RENA, RENB, RENC, RANG, OPTH, DXDX, DXDY, DYDX, DYDY, RETX, RETY, RAGX, RAGY, RAGZ, RAGA, RAGB, RAGC, RAIN, PLEN, HHCN, RAID
Constraints on element positions	GLCX, GLCY, GLCZ, GLCA, GLCB, GLCC
Changing system data	CONF, PRIM, SVIG
General math operands	ABSO, SUMM, OSUM, DIFF, PROD, DIVI, SQRT, OPGT, OPLT, CONS, QSUM, EQUA, MINN, MAXX, ACOS, ASIN, ATAN, COSI, SINE, TANG
Multi-configuration (zoom) data	CONF, ZTHI, MCOV
Gaussian beam data	GBWA, GBW0, GBWZ, GBWR
Gradient index control operands	I1GT, I2GT, I3GT, I4GT, I5GT, I6GT, I1LT, I2LT, I3LT, I4LT, I5LT, I6LT, I1VA, I2VA, I3VA, I4VA, I5VA, I6VA. GRMN, GRMX, LPTD, DLTN
Optimization with ZPL macros	ZPLM
Merit function control operands	BLNK, ENDX, USYM, DMFS, SKIS, SKIN

### OPTIMIZATION OPERANDS AND DATA FIELD USAGE

NAME	Description	Int1	Int2	Hxy, Pxy
ABSO	Absolute value.	Op #	-	-
ACOS	Arcosine of the value of the specified operand number. If flag is 0, then the units are radians, otherwise, degrees.	Op #	Flag	-
AMAG	Angular magnification. This is the ratio of the image to object space paraxial chief ray angles. Not valid for non-paraxial systems.	-	Wave	-
ANAR	Angular aberration radius measured in the image plane with respect to the primary wavelength chief ray. This quantity is defined as $1 - \cos\theta$ , where $\theta$ is the angle between the chief ray and the ray being traced. See TRAR.	-	Wave	Yes
ASIN	Arcsine of the value of the specified operand number. If flag is 0, then the units are radians, otherwise, degrees.	Op #	Flag	-
ASTI	Astigmatism in waves contributed by the specified surface. If the surface value is zero, the sum for the entire system is used. This is the third order astigmatism calculated from the Seidel coefficients, and is not valid for non-paraxial systems.	Surf	Wave	-
ATAN	Arctangent of the value of the specified operand number. If flag is 0, then the units are radians, otherwise, degrees.	Op #	Flag	-
AXCL	Axial color in lens units. This is the paraxial focal plane separation between the two most extreme wavelengths defined. The distance is measured along the Z axis. Not valid for non-paraxial systems.	-	-	-
BLNK	Does nothing. Used for separating portions of the operand list. This operand does not use the target or weight columns.	-	-	-
COGT	Boundary operand that constrains the conic of surface "Surf" to be greater than the specified target value.	Surf	-	-
COLT	Boundary operand that constrains the conic of surface "Surf" to be less than the specified target value.	Surf	-	-
COMA	Coma in waves contributed by the specified surface. If the surface value is zero, the sum for the entire system is used. This is the third order coma calculated from the Seidel coefficients, and is not valid for non-paraxial systems.	Surf	Wave	-

NAME	Description	Int1	Int2	Hxy, Pxy
CONF	Configuration. This operand is used to change the configuration number during merit function evaluation, which permits optimization across multiple configurations. This operand does not use the target or weight columns.	New #	-	-
CONS	Constant value. This is used to enter in constant values for use in other operand computations. The value will be identical to the target value.	-	-	-
COSI	Cosine of the value of the specified operand number. If flag is 0, then the units are radians, otherwise, degrees.	Op #	Flag	-
COVA	Conic value. Returns the conic constant of a surface.	Surf	-	-
CTGT	Center thickness greater than. This boundary operand constrains the center thickness of surface "Surf" to be greater than the specified target value. See also "MNCT".	Surf	-	-
CTLT	Center thickness less than. This boundary operand constrains the center thickness of surface "Surf" to be less than the specified target value. See also "MXCT".	Surf	-	-
CTVA	Center thickness value. Constrains the center thickness of surface "Surf" to be equal to the specified target value.	Surf	-	-
CVGT	Curvature greater than. This boundary operand constrains the curvature of surface "Surf" to be greater than the target value.	Surf	-	-
CVLT	Curvature less than. This boundary operand constrains the curvature of surface "Surf" to be less than the target value.	Surf	-	-
CVOL	Cylinder volume. This operand computes the volume in cubic lens units of the smallest cylinder that will contain the specified range of surfaces. Only the vertex positions and semi-diameters are used in the calculation, not the sag. The range of surfaces should not include any coordinate breaks.	First surf	Last surf	-
CVVA	Curvature value. This operand constrains the curvature of surface "Surf" to be equal to the specified target value.	Surf	-	-
DENC	Diffraction Encircled Energy. This operand computes the radius in microns to the specified fraction of encircled energy, at the specified field. The calculation is always polychromatic and referenced to the centroid. The Int1 parameter must be an integer (1, 2, ...) where 1 yields 8x8 sampling, 2 yields 16x16 sampling, etc. The Int2 must be a valid field number (1, 2, ...). The Hx value is the fraction of energy, and must be greater than 0 and less than 1. If the sampling is too low, the radius returned is a 1e+10.	Sampling	Field	(see left)
DIFF	Difference of two operands (OP#1 - OP#2). The two arguments are the row numbers of the operands to subtract.	Op #1	Op #2	-
DIMX	Distortion maximum. This is similar to DIST, except it specifies only an upper bound for the absolute value of the distortion. The Field integer can be zero, which specifies the maximum field coordinate be used, or any valid field number. Note the maximum distortion does not always occur at the maximum field coordinate. The value returned is always in units of percentage, for the system as a whole. This operand may not be valid or non-rotationally symmetric systems.	Field	Wave	-
DISC	Distortion, calibrated. This operand computes the calibrated distortion across the field of view, and returns the absolute value of the maximum deviation from linearity of the f-theta condition. This operand is extremely useful for designing f-theta lenses.	-	Wave	-
DISG	Generalized distortion in percent. This operand computes the distortion for any ray in the pupil, from anywhere in the field, at any wavelength, using any field point as a reference. The method used and assumptions made are the same as for the grid distortion plot described in the Analysis Menu Chapter.	Ref Fld	Wave	Yes

NAME	Description	Int1	Int2	Hxy, Pxy
DIST	Distortion in waves contributed by surface "Surf". If the surface value is zero, the sum for the entire system is used. Also, if the surface number is zero, the distortion is given in percent instead. This is the third order distortion calculated from the Seidel coefficients, and is not valid for non-paraxial systems.	Surf	Wave	-
DIVI	Division of first by second operand. The two arguments are the row numbers of the operands to divide.	Op #1	Op #2	-
DLTN	Delta N. Computes the difference between the maximum and minimum index of refraction on axis for a gradient index surface. The minimum and maximum z coordinates used are computed by accounting for the sag of both ends of the surface. See the section "Using gradient index operands".	Surf	Wave	-
DMFS	Default merit function start. This operand is simply a flag to indicate where the default merit function should be appended, if one is subsequently created. The row number after this operand will appear as the default "Start At" row on the default merit function dialog box.	-	-	-
DMGT	Diameter greater than. This boundary operand constrains the diameter of surface "Surf" to be greater than the specified target value. The diameter is twice the semi-diameter value displayed on the main spreadsheet.	Surf	-	-
DMLT	Diameter less than. This boundary operand constrains the diameter of surface "Surf" to be less than the specified target value. The diameter is twice the semi-diameter value displayed on the main spreadsheet.	Surf	-	-
DMVA	Diameter value. This operand constrains the diameter of surface "Surf" to be equal to the specified target value. The diameter is twice the semi-diameter value displayed on the main spreadsheet.	Surf	-	-
DXDX	Derivative of transverse x-aberration with respect to x-pupil coordinate. This is the slope of the ray fan plot at the specified pupil coordinate.	-	Wave	Yes
DXDY	Derivative of transverse x-aberration with respect to y-pupil coordinate. This is the slope of the ray fan plot at the specified pupil coordinate.	-	Wave	Yes
DYDX	Derivative of transverse y-aberration with respect to x-pupil coordinate. This is the slope of the ray fan plot at the specified pupil coordinate.	-	Wave	Yes
DYDY	Derivative of transverse y-aberration with respect to y-pupil coordinate. This is the slope of the ray fan plot at the specified pupil coordinate.	-	Wave	Yes
EFFL	Effective focal length in lens units. This is paraxial, and may not be accurate for non-paraxial systems.	-	Wave	-
EFLX	Effective focal length in the local x plane of the specified range of surfaces at the primary wavelength.	First surf	Last surf	-
EFLY	Effective focal length in the local y plane of the specified range of surfaces at the primary wavelength.	First surf	Last surf	-
ENDX	End execution. Terminates the computation of the merit function. All remaining operands are ignored.	-	-	-
ENPP	Entrance pupil position in lens units, with respect to the first surface. This is the paraxial pupil position, valid only for centered systems.	-	-	-
EPDI	Entrance pupil diameter in lens units.	-	-	-

NAME	Description	Int1	Int2	Hxy, Pxy
EQUA	Equal operand. This operand constrains all operands within the specified range of operands to have the same value within the tolerance specified by the target. The value of this operand is computed by finding the average of the range of values, and then summing the absolute value of the errors between each operand and the average if the error exceeds the target value. See SUMM and OSUM.	First	Last	-
ETGT	Edge thickness greater than. This boundary operand constrains the edge thickness of surface "Surf" to be greater than the specified target value. The edge thickness is calculated at the semi-diameter radius along the +y axis if code is zero, the +x axis if code is 1, the -y axis if code is 2, and the -x axis if code is 3. See also "MNET".	Surf	Code	-
ETLT	Edge thickness less than. This boundary operand constrains the edge thickness of surface "Surf" to be less than the specified target value. The edge thickness is always calculated at the semi-diameter radius along the +y axis if code is zero, the +x axis if code is 1, the -y axis if code is 2, and the -x axis if code is 3. See also "MXET".	Surf	Code	-
ETVA	Edge thickness value. Constrains the edge thickness of surface "Surf" to be equal to the specified target value. The edge thickness is always calculated at the semi-diameter radius along the +y axis if code is zero, the +x axis if code is 1, the -y axis if code is 2, and the -x axis if code is 3. See also "MNET".	Surf	Code	-
EXPP	Exit pupil position in lens units, with respect to the image surface. This is the paraxial pupil position, valid only for centered systems.	-	-	-
FCGS	Generalized field curvature, sagittal. The field curvature value for any field point, at any wavelength. The value is generalized to return reasonable results even for non-rotationally symmetric systems; see the Field Curvature feature in the Analysis Menu Chapter.	-	Wave	Hx,Hy
FCGT	Generalized field curvature, tangential; see FCGS.	-	Wave	Hx,Hy
FCUR	Field curvature in waves contributed by the specified surface. If the surface value is zero, the sum for the entire system is used. This is the third order field curvature calculated from the Seidel coefficients, and is not valid for non-paraxial systems.	Surf	Wave	-
GBW0	Minimum Gaussian beam waist in image space of specified surface. If Hx is non-zero, then the computation is for the x-direction beam, otherwise, it is for the y-direction. See the Gaussian beam feature for details.	Surf	Wave	See left
GBWA	Gaussian beam size on surface "Surf". If Hx is non-zero, then the computation is for the x-direction beam, otherwise, it is for the y-direction. See the Gaussian beam feature for details.	Surf	Wave	See left
GBWZ	Z-coordinate of image space Gaussian beam waist relative to surface. If Hx is non-zero, then the computation is for the x-direction beam, otherwise, it is for the y-direction. See the Gaussian beam feature for details.	Surf	Wave	See left
GBWR	Gaussian beam radius of curvature at the specified surface. If Hx is non-zero, then the computation is for the x-direction beam, otherwise, it is for the y-direction. See the Gaussian beam feature for details.	Surf	Wave	See left
GLCA	Global x-direction normal vector component of surface "Surf".	Surf	-	-
GLCB	Global y-direction normal vector component of surface "Surf".	Surf	-	-
GLCC	Global z-direction normal vector component of surface "Surf".	Surf	-	-
GLCX	Global vertex x-coordinate of surface "Surf".	Surf	-	-
GLCY	Global vertex y-coordinate of surface "Surf".	Surf	-	-

NAME	Description	Int1	Int2	Hxy, Pxy
GLCZ	Global vertex z-coordinate of surface "Surf".	Surf	-	-
GMTA	Geometric MTF average of sagittal and tangential response. The Int1 parameter must be an integer (1, 2, ...) indicating the number of hexapolar rings to use. Usually a value of 8 or more is required. The Int2 can be a valid wavelength number, or 0 for polychromatic. The Hx value must be a valid field number (1, 2, ...). Hy is the spatial frequency in cycles per mm. Px is a flag; if zero, then the diffraction limit will be used to scale the results (recommended) otherwise, no scaling is done. See the discussion "Using MTF operands" in this chapter for details.		See left.	
GMTS	Geometric MTF sagittal response. See GMTA for details.		See left.	
GMTT	Geometric MTF tangential response. See GMTA for details.		See left.	
GRMN	Gradient index minimum index. This boundary operand sets the minimum allowable index of refraction value for the gradient index surface "Surf" at the wavelength number "Wave". The index is checked at six places: the front vertex, front +y top, front +x side, rear vertex, rear +y top, and the rear +x side. See also "InGT", "InLT" and "GRMX".	Surf	Wave	-
GRMX	Gradient index maximum index. This boundary operand sets the maximum allowable index of refraction value for the gradient index surface "Surf" at the wavelength number "Wave". The index is checked at six places: the front vertex, front +y top, front +x side, rear vertex, rear +y top, and the rear +x side. See also "InGT", "InLT" and "GRMN".	Surf	Wave	-
HHCN	Test for the hyper-hemisphere condition. ZEMAX traces the specified ray to the specified surface, and computes the x, y, and z intercept coordinates. Then, the x and y coordinates only are used in the sag expression for that surface to see what z coordinate results. If the z coordinates are not the same, then HHCN returns 1, otherwise it returns zero. This operand can be used to prevent optimizations from reaching solutions that require hyper-hemispheric surface shapes.	Surf	Wave	Yes
InGT	Index "n" greater than. This boundary operand constrains the index of refraction at wavelength number "Wave" of gradient index surface "Surf" at one of six points inside the gradient index lens. For n=1, the point is the front vertex, n=2 is the front +y top, n=3 is the front +x side, n=4 is the rear vertex, n=5 is the rear +y top, and n=6 is the rear +x side. In all cases the operand bounds the index at the specified point to be greater than the specified target value. For example, "I4GT" constrains the minimum index at the rear vertex of surface (i.e. the vertex of the next surface) of the gradient index lens. In all cases the +y top and +x side distance is defined by the larger of the front and rear semi-diameters set on the main spreadsheet. See also "GRMN" and "GRMX", which are similar operands that are easier to use.	Surf	Wave	-
InLT	Index "n" less than. This operand is similar to "InGT" except it constrains the maximum value of the index of refraction rather than the minimum. See "InGT" for a complete description of the parameter "n".	Surf	Wave	-
InVA	This operand is similar to "InGT" except it simply constrains the current value of the index of refraction. See "InGT" for a complete description of the parameter "n".	Surf	Wave	-
ISFN	Image space F/#. This operand is the paraxial infinite conjugate F/#. See "WFNO".	-	-	-
LACL	Lateral color. This is the y-distance between the paraxial chief ray intercepts of the two extreme wavelengths defined. Not valid for non-paraxial systems.	-	-	-

NAME	Description	Int1	Int2	Hxy, Pxy
LINV	Lagrange invariant of system in lens units. The paraxial marginal and chief ray data are used to compute this value.	-	Wave	-
LPTD	This boundary operand constrains the slope of the axial gradient index profile from changing signs within a gradient index component. See the section "Using gradient index operands".	Surf	-	-
MAXX	Returns the largest value within the indicated range of operands. See MINN.	First	Last	-
MCOV	Multi-configuration operand value. This is used to directly target or compute values in the multi-configuration editor.	MC operand #	Config #	-
MINN	Returns the smallest value within the indicated range of operands. See MAXX.	First	Last	-
MNAB	Minimum Abbe number. This boundary operand constrains the Abbe number of surfaces between "First Surf" and "Last Surf" to be greater than the specified target value. See also "MXAB".	First surf	Last surf	-
MNCA	Minimum center thickness air. This boundary operand constrains each of the center thicknesses of surfaces from "First surf" to "Last surf" which have air (i.e. no glass) as the glass type to be greater than the specified target value. See also "MNCT" and "MNCG". This operand controls multiple surfaces simultaneously.	First surf	Last surf	-
MNCG	Minimum center thickness glass. This boundary operand constrains each of the thicknesses of surfaces from "First surf" to "Last surf" which have a non-air glass type to be greater than the target value. See also "MNCT" and "MNCA". This operand controls multiple surfaces simultaneously.	First surf	Last surf	-
MNCT	Minimum center thickness. This boundary operand constrains each of the center thicknesses of surfaces from "First surf" to "Last surf" to be greater than the specified target value. See also "MNCG" and "MNCA". This operand controls multiple surfaces simultaneously.	First surf	Last surf	-
MNEA	Minimum edge thickness air. This boundary operand constrains each of the edge thicknesses of surfaces from "First surf" to "Last surf" which have air (i.e. no glass) as the glass type to be greater than the specified target value. See also "MNET", "MNEG", "ETGT", and "XNEA". This operand controls multiple surfaces simultaneously. The boundary applies to the top "+y" edge of the surface only; see XNEA for constraining non-rotationally symmetric surfaces.	First surf	Last surf	-
MNEG	Minimum edge thickness glass. This boundary operand constrains each of the edge thicknesses of surfaces from "First surf" to "Last surf" which have a non-air glass type to be greater than the specified target value. See also "MNET", "MNEA", "ETGT", and "XNEG". This operand controls multiple surfaces simultaneously. The boundary applies to the top "+y" edge of the surface only; see XNEG for constraining non-rotationally symmetric surfaces.	First surf	Last surf	-
MNET	Minimum edge thickness. This boundary operand constrains each of the edge thicknesses of surfaces from "First surf" to "Last surf" to be greater than the specified target value. See also "MNEG", "MNEA", "ETGT", and "XNET". This operand controls multiple surfaces simultaneously. The boundary applies to the top "+y" edge of the surface only; see XNET for constraining non-rotationally symmetric surfaces.	First surf	Last surf	-
MNIN	Minimum index at d-light. This boundary operand constrains the Nd value of surfaces between "First Surf" and "Last Surf" to be greater than the specified target value. See also "MXIN".	First surf	Last surf	-

NAME	Description	Int1	Int2	Hxy, Pxy
MNPD	Minimum $\Delta P_{g,F}$ . This boundary operand constrains the deviation of the partial dispersion of surfaces between "First Surf" and "Last Surf" to be greater than the specified target value. See also "MXPDP".	First surf	Last surf	-
MNSD	Minimum semi-diameter. Constrains the semi-diameter to be larger than the specified target over the surface range.	First surf	Last surf	-
MSWA	Modulation square-wave transfer function, average of sagittal and tangential. See "MTFT" for details.	Sampling	Wave	(see left)
MSWS	Modulation square-wave transfer function, sagittal. See "MTFT" for details.	Sampling	Wave	(see left)
MSWT	Modulation square-wave transfer function, tangential. See "MTFT" for details.	Sampling	Wave	(see left)
MTFA	Modulation transfer function, average of sagittal and tangential. See "MTFT" for details.	Sampling	Wave	(see left)
MTFS	Modulation transfer function, sagittal. See "MTFT" for details.	Sampling	Wave	(see left)
MTFT	Modulation transfer function, tangential. This computes the diffraction MTF. The Int1 parameter must be an integer (1, 2, ...) where 1 yields 8x8 sampling, 2 yields 16x16 sampling, etc. The Int2 can be a valid wavelength number, or 0 for polychromatic. The Hx value must be a valid field number (1, 2, ...). Hy is the spatial frequency in cycles per mm. If the sampling is set too low for accurate computation of the MTF, then the MTF operands all return zero. See the discussion "Using MTF operands" in this chapter for details.	Sampling	Wave	(see left)
MXAB	Maximum Abbe number. This boundary operand constrains the Abbe number of surfaces between "First Surf" and "Last Surf" to be less than the specified target value. See also "MNAB".	First surf	Last surf	-
MXCA	Maximum center thickness air. This boundary operand constrains each of the thicknesses of surfaces from "First surf" to "Last surf" which have air (i.e. no glass) as the glass type to be less than the target value. See also "MXCT" and "MXCG". This operand controls multiple surfaces simultaneously.	First surf	Last surf	-
MXCG	Maximum center thickness glass. This boundary operand constrains each of the center thicknesses of surfaces from "First surf" to "Last surf" which have a non-air glass type to be less than the target value. See also "MXCT" and "MXCA". This operand controls multiple surfaces simultaneously.	First surf	Last surf	-
MXCT	Maximum center thickness. This boundary operand constrains each of the center thicknesses of surfaces from "First surf" to "Last surf" to be less than the specified target value. See also "MXCG" and "MXCA". This operand controls multiple surfaces simultaneously.	First surf	Last surf	-
MXEA	Maximum edge thickness air. This boundary operand constrains each of the edge thicknesses of surfaces from "First surf" to "Last surf" which have air (i.e. no glass) as the glass type to be less than the specified target value. See also "MXET", "MXEG", "ETLT", and "XXEA". This operand controls multiple surfaces simultaneously. The boundary applies to the top "+y" edge of the surface only; see XXEA for constraining non-rotationally symmetric surfaces.	First surf	Last surf	-
MXEG	Maximum edge thickness glass. This boundary operand constrains each of the edge thicknesses of surfaces from "First surf" to "Last surf" which have a non-air glass type to be less than the target value. See also "MXET", "MXEA", "ETLT", and "XXEG". This operand controls multiple surfaces simultaneously. The boundary applies to the top "+y" edge of the surface only; see XXEG for constraining non-rotationally symmetric surfaces.	First surf	Last surf	-

NAME	Description	Int1	Int2	Hxy, Pxy
MXET	Maximum edge thickness. This boundary operand constrains each of the edge thicknesses of surfaces from "First surf" to "Last surf" to be less than the specified target value. See also "MXEG", "MXEA", "ETLT", and "XXET". This operand controls multiple surfaces simultaneously. The boundary applies to the top "+y" edge of the surface only; see XXET for constraining non-rotationally symmetric surfaces.	First surf	Last surf	-
MXIN	Maximum index at d-light. This boundary operand constrains the Nd value of surfaces between "First Surf" and "Last Surf" to be less than the specified target value. See also "MNIN".	First surf	Last surf	-
MXPD	Maximum $\Delta P_{g,F}$ . This boundary operand constrains the deviation of the partial dispersion of surfaces between "First Surf" and "Last Surf" to be less than the specified target value. See also "MNPD".	First surf	Last surf	-
MXSD	Maximum semi-diameter. Constrains the semi-diameter to be less than the specified target over the surface range.	First surf	Last surf	-
OBSN	Object space numerical aperture. This is only useful for finite conjugate systems, and is calculated on axis at the primary wavelength.	-	-	-
OFF	This operand indicates an unused entry in the operand list. OFF operands are automatically converted to BLNK operands upon evaluation of the merit function. OFF is only used to indicate that the merit function operand type was not recognized.	-	-	-
OPDC	Optical path difference with respect to chief ray in waves.	-	Wave	Yes
OPDX	Optical path difference with respect to the diffraction centroid. OPDX has the same restrictions that TRAC does; see TRAC for a detailed discussion.	-	Wave	Yes
OPGT	Operand greater than. This is used to make any operand an inequality constraint.	Op #	-	-
OPLT	Operand less than. This is used to make any operand an inequality constraint.	Op #	-	-
OPTH	Optical path length. This is the distance, in lens units, the specified ray travels to the surface "Surf". The distance is measured from the object for finite conjugates; otherwise the distance is referenced to the first surface. The optical path accounts for the index of refraction of the media, and for phase adding surfaces such as gratings and binary optics. See PLEN.	Surf	Wave	Yes
OSUM	Sums the values of all operands between the two specified operands. See SUMM.	First	Last	-
PnGT	Parameter "n" greater than. This boundary operand constrains the value of parameter n of surface "Surf" to be greater than the target value. The symbol "n" should be replaced by a number between 1 and 8. For example, "P3GT" constrains the value of parameter 3 to be greater than the target value. The parameter values have different meanings depending upon the surface type. See the chapter "Using ZEMAX" for a description of the parameter values.	Surf	-	-
PnLT	Parameter "n" less than. This boundary operand constrains the value of parameter n of surface "Surf" to be less than the target value. The symbol "n" should be replaced by a number between 1 and 8. For example, "P5LT" constrains the value of parameter 5 to be less than the target value. The parameter values have different meanings depending upon the surface type. See the chapter "Using ZEMAX" for a description of the parameter values.	Surf	-	-



NAME	Description	Int1	Int2	Hxy, Pxy
PnVA	Parameter "n" value. This boundary operand constrains the value of parameter n of surface "Surf" to be equal to the target value. The symbol "n" should be replaced by a number between 1 and 8. For example, "P1VA" constrains the value of parameter 1 to be equal to the target value. The parameter values have different meanings depending upon the surface type. See the chapter "Using ZEMAX" for a description of the parameter values.	Surf	-	-
PANA	Paraxial ray x-direction surface normal at the ray-surface intercept. This is the x component of the surface normal vector at the intersection point of the specified paraxial ray and the surface "Surf", in the local coordinate system.	Surf	Wave	Yes
PANB	Paraxial ray y-direction surface normal at the ray-surface intercept. This is the y component of the surface normal vector at the intersection point of the specified paraxial ray and the surface "Surf", in the local coordinate system.	Surf	Wave	Yes
PANC	Paraxial ray z-direction surface normal at the ray-surface intercept. This is the z component of the surface normal vector at the intersection point of the specified paraxial ray and the surface "Surf", in the local coordinate system.	Surf	Wave	Yes
PARA	Paraxial ray x-direction cosine of the ray after refraction from the surface "Surf". This is the x component of the specified paraxial ray vector after refraction, in the local coordinate system.	Surf	Wave	Yes
PARB	Paraxial ray y-direction cosine of the ray after refraction from the surface "Surf". This is the y component of the specified paraxial ray vector after refraction, in the local coordinate system.	Surf	Wave	Yes
PARC	Paraxial ray z-direction cosine of the ray after refraction from the specified surface. This is the z component of the specified paraxial ray vector after refraction, in the local coordinate system.	Surf	Wave	Yes
PARR	Paraxial ray radial coordinate in lens units at the specified surface. This is the radial distance from the local axis to the intersection of the surface "Surf" and the specified paraxial ray, in the local coordinate system.	Surf	Wave	Yes
PARX	Paraxial ray x-coordinate in lens units at the surface "Surf".	Surf	Wave	Yes
PARY	Paraxial ray y-coordinate in lens units at the surface "Surf".	Surf	Wave	Yes
PARZ	Paraxial ray z-coordinate in lens units at the surface "Surf".	Surf	Wave	Yes
PATX	Paraxial ray x-direction ray tangent. This is the tangent of the angle the paraxial ray makes in the X-Z plane after refraction from surface "Surf".	Surf	Wave	Yes
PATY	Paraxial ray y-direction ray tangent. This is the tangent of the angle the paraxial ray makes in the Y-Z plane after refraction from surface "Surf".	Surf	Wave	Yes
PETC	Petzval curvature in inverse lens units. Not valid for non-paraxial systems.	-	Wave	-
PETZ	Petzval radius of curvature in lens units. Not valid for non-paraxial systems.	-	Wave	-
PIMH	Paraxial image height at the paraxial image plane for the specified wavelength. Not valid for non-paraxial systems.	-	Wave	-
PLEN	Path length. This operand computes the total optical path length (including index of refraction) between surfaces 1 and 2 for the specified ray, which is always traced at the primary wavelength. See OPTH.	Surf 1	Surf 2	Yes
PMAG	Paraxial magnification. This is the ratio of the paraxial chief ray height on the paraxial image plane to the object height. Only useful for finite conjugate systems. Note the paraxial image plane is used even if the system is not at paraxial focus.	-	Wave	-

NAME	Description	Int1	Int2	Hxy, Pxy
POWR	The surface power (in inverse lens units) of the surface "Surf". This operand only works for standard surfaces.	Surf	Wave	-
PRIM	Primary wavelength. This is used to change the primary wavelength number during merit function evaluation. This operand does not use the target or weight columns.	-	Wav #	-
PROD	Product of two operands. The two arguments are the row numbers of the operands to multiply.	Op #1	Op #2	-
QSUM	Quadratic sum. This operand squares and then adds all operands between the first and last operand operand (inclusive), then takes the square root of the sum. See also SUMM, OSUM, EQUA.	First	Last	-
RAGA	Global ray x-direction cosine. This is the direction cosine of the ray in the global coordinate system. The origin of the global coordinate system is at the object surface for finite conjugates, otherwise it is at the first surface.	Surf	Wave	Yes
RAGB	Global ray y-direction cosine. See "RAGA".	Surf	Wave	Yes
RAGC	Global ray z-direction cosine. See "RAGA".	Surf	Wave	Yes
RAGX	Global ray x-coordinate. This is the coordinate in lens units in the global coordinate system. The origin of the global coordinate system is at the object surface for finite conjugates, otherwise it is at the first surface.	Surf	Wave	Yes
RAGY	Global ray y-coordinate. See "RAGX".	Surf	Wave	Yes
RAGZ	Global ray z-coordinate. See "RAGX".	Surf	Wave	Yes
RAID	Real ray angle of incidence. This is the angle in degrees between the surface normal and the ray.	Surf	Wave	Yes
RAIN	Real ray angle of incidence. This is the cosine of the angle between the surface normal and the ray before refraction at that surface. Does not return correct results if the surface prior to surf is a gradient index surface.	Surf	Wave	Yes
RANG	Ray angle in radians with respect to z axis. The angle is measured with respect to the local Z axis.	Surf	Wave	Yes
REAA	Real ray x-direction cosine of the ray after refraction from the surface "Surf". This is the x component of the specified real ray vector after refraction, in the local coordinate system.	Surf	Wave	Yes
REAB	Real ray y-direction cosine of the ray after refraction from the surface "Surf". This is the y component of the specified real ray vector after refraction, in the local coordinate system.	Surf	Wave	Yes
REAC	Real ray z-direction cosine of the ray after refraction from the surface "Surf". This is the z component of the specified real ray vector after refraction, in the local coordinate system.	Surf	Wave	Yes
REAR	Real ray radial coordinate in lens units at the surface "Surf".	Surf	Wave	Yes
REAX	Real ray x-coordinate in lens units at the surface "Surf".	Surf	Wave	Yes
REAY	Real ray y-coordinate in lens units at the surface "Surf".	Surf	Wave	Yes
REAZ	Real ray z-coordinate in lens units at the surface "Surf".	Surf	Wave	Yes
RENA	Real ray x-direction surface normal at the ray-surface intercept.	Surf	Wave	Yes
RENB	Real ray y-direction surface normal at the ray-surface intercept.	Surf	Wave	Yes
RENC	Real ray z-direction surface normal at the ray-surface intercept.	Surf	Wave	Yes
RETX	Real ray x-direction ray tangent (slope).	Surf	Wave	Yes
RETY	Real ray y-direction ray tangent (slope).	Surf	Wave	Yes

NAME	Description	Int1	Int2	Hxy, Pxy
RGLA	Reasonable glass. This operand restricts the deviation the index, Abbe, and deviation of the partial dispersion values may take from actual glasses in the currently loaded glass catalogs. See the section "Optimizing glass selection" for a complete discussion. The constraint is active over the surface range specified.	First Surf	Last Surf	-
RSCE	RMS spot size (ray aberrations) with respect to the geometric image centroid, measured in lens units. This operand is similar to RSCH, except the reference point is the image centroid instead of the chief ray. See RSCH for details.	Rings	Wave	Hx, Hy
RSCH	RMS spot size (ray aberrations) with respect to chief ray. This operand uses a Gaussian quadrature method to estimate the RMS spot size at a specified field coordinate and wavelength. The number returned is in lens units. The method used is only accurate for systems with circular pupils. The Int1 column is used to specify the number of rings of rays traced (use no more than required to converge the result). Only Hx and Hy are used to define the field point, Px and Py are not used. If the "wave" value is zero, then a wavelength weighted polychromatic calculation is performed.	Rings	Wave	Hx, Hy
RSRE	RMS spot size (ray aberrations) with respect to the geometric image centroid, measured in lens units. This operand is similar to RSCE, except a rectangular grid of rays is used instead of the Gaussian quadrature method. This operand always considers vignetting. A grid value of 1 will trace 4 rays, 2 will trace a 2x2 grid per quadrant (16 rays), 3 will trace 3x3 rays per quadrant (36 rays), and so on. Symmetry is considered.	Grid	Wave	Hx, Hy
RSRH	Similar to RSRE, except the reference point is the chief ray.	Grid	Wave	Hx, Hy
RWCE	RMS wavefront error with respect to the diffraction centroid. This operand is useful for minimizing the wavefront variance, which is proportional the Strehl ratio and the area under the MTF curve. The units are waves. See RWCH. See RSCH for details.	Rings	Wave	Hx, Hy
RWCH	RMS wavefront error with respect to the chief ray. The units are waves. This RMS really refers to the standard deviation of the wavefront, since the mean OPD is subtracted out. See RWCE. See RSCH for details.	Rings	Wave	Hx, Hy
RWRE	Like RSRE, except for wavefront error rather than spot size.	Grid	Wave	Hx, Hy
RWRH	Like RSRH, except for wavefront error rather than spot size.	Grid	Wave	Hx, Hy
SAGX	The sag in lens units of the surface "Surf" in the XZ plane at the semi-diameter distance.	Surf	-	-
SAGY	The sag in lens units of the surface "Surf" in the YZ plane at the semi-diameter distance.	Surf	-	-
SINE	Sine of the value of the specified operand number. If flag is 0, then the units are radians, otherwise, degrees.	Op #	Flag	-
SKIS	Skip if symmetric. If the lens is rotationally symmetric, then computation of the merit function continues at the specified operand number.	Op #		
SKIN	Skip if not symmetric. See SKIS.	Op #		
SPHA	Spherical aberration in waves contributed by the specified surface. If the surface value is zero, the sum for the entire system is used.	Surf	Wave	-
SQRT	Square root of operand. The argument is the row number of the operand value to take the square root of.	Op #	-	-
SUMM	Sum of two operands. The two arguments are the row numbers of the operands to add. See OSUM.	Op #1	Op #2	-
SVIG	Sets the vignetting factors. When included, updates the vignetting factors for the current configuration.	-	-	-

NAME	Description	Int1	Int2	Hxy, Pxy
TANG	Tangent of the value of the specified operand number. If flag is 0, then the units are radians, otherwise, degrees.	Op #	Flag	-
TOTR	Total track (length) of lens in lens units.	-	-	-
TRAC	Transverse aberration radial direction measured in the image plane with respect to the centroid. Unlike most other operands, TRAC critically depends upon the placement of other TRAC operands within the Merit Function Editor to work correctly. TRAC operands must be grouped together by field position and wavelength. ZEMAX traces all TRAC rays with a common field point together, and then uses the collective data to compute the centroid of all the rays. Each ray individually is then referenced to the computed centroid. This operand should only be entered into the Merit Function Editor by the Default Merit Function tool, and is not recommended for use directly by the user.	-	Wave	Yes
TRAD	The x component of the TRAR only. TRAD has the same restrictions that TRAC does; see TRAC for a detailed discussion.		Wave	Yes
TRAE	The y component of the TRAR only. TRAE has the same restrictions that TRAC does; see TRAC for a detailed discussion.		Wave	Yes
TRAI	Transverse aberration radius measured at the specified surface with respect to the chief ray. Similar to TRAR, except a surface other than the image surface may be specified.	Surf	Wave	Yes
TRAR	Transverse aberration radial direction measured in the image plane with respect to the chief ray. See ANAR.	-	Wave	Yes
TRAX	Transverse aberration x direction measured in the image plane with respect to the chief ray.	-	Wave	Yes
TRAY	Transverse aberration y direction measured in the image plane with respect to the chief ray.	-	Wave	Yes
TRCX	Transverse aberration x direction measured in the image plane with respect to the centroid. See TRAC. This operand should only be entered into the Merit Function Editor by the Default Merit Function tool, and is not recommended for use directly by the user.	-	Wave	Yes
TRCY	Transverse aberration y direction measured in the image plane with respect to the centroid. See TRAC. This operand should only be entered into the Merit Function Editor by the Default Merit Function tool, and is not recommended for use directly by the user.	-	Wave	Yes
TTGT	Total thickness greater than. This boundary operand constrains the total thickness (including both front and back surface sags) of surface "Surf" to be greater than the specified target value. The thickness is calculated at the semi-diameter radius along the +y axis if code is zero, the +x axis if code is 1, the -y axis if code is 2, and the -x axis if code is 3.	Surf	Code	-
TTHI	Sum of thicknesses from first to last specified surface. Note that the sum is inclusive, it is not the thickness between the two surfaces.	First surf	Last surf	-
TTLT	Total thickness less than. See TTGT.	Surf	Code	-
TTVA	Total thickness value. See TTGT.	Surf	Code	-
USYM	If present in the merit function, this operand instructs ZEMAX to assume radial symmetry exists in the lens even if ZEMAX detects symmetry does not exist. This speeds execution of the merit function in some special cases. See "Assume Axial Symmetry" in the default merit function description earlier in this chapter.	-	-	-
VOLU	Volume of element in cubic cm. The resulting value is only valid for regions bounded by plane and spherical surfaces.	Surf	-	-
WFNO	Working F/#. This is computed from the angle the real marginal ray makes with respect to the chief ray in image space.	-	-	-

NAME	Description	Int1	Int2	Hxy, Pxy
XDGT	Extra data value greater than. The Int2 number must be between 1 and 199 to indicate which of the extra data values is selected.	Surf	Number	-
XDLT	Extra data value less. The Int2 number must be between 1 and 199 to indicate which of the extra data values is selected.	Surf	Number	-
XDVA	Extra data value. The Int2 number must be between 1 and 199 to indicate which of the extra data values is selected.	Surf	Number	-
XNEA	Minimum edge thickness for air surfaces. This operand checks the edge thickness at numerous points around the perimeter of the surface, and insures all points are at least the minimum specified thickness. See "MNEA".	First surf	Last surf	-
XNEG	Minimum edge thickness for glass surfaces. This operand checks the edge thickness at numerous points around the perimeter of the surface, and insures all points are at least the minimum specified thickness. See "MNEG".	First surf	Last surf	-
XNET	Minimum edge thickness. This operand checks the edge thickness at numerous points around the perimeter of the surface, and insures all points are at least the minimum specified thickness. See "MNET".	First surf	Last surf	-
XXEA	Maximum edge thickness for air surfaces. This operand checks the edge thickness at numerous points around the perimeter of the surface, and insures all points are no more than the maximum specified thickness. See "MXEA".	First surf	Last surf	-
XXEG	Maximum edge thickness for glass surfaces. This operand checks the edge thickness at numerous points around the perimeter of the surface, and insures all points are no more than the maximum specified thickness. See "MXEG".	First surf	Last surf	-
XXET	Maximum edge thickness. This operand checks the edge thickness at numerous points around the perimeter of the surface, and insures all points are no more than the maximum specified thickness. See "MXET".	First surf	Last surf	-
YNIP	YNI-paraxial. This number is the product of the paraxial marginal ray height times the index times the angle of incidence. This quantity is proportional to the narcissus contribution of the specified surface. See Applied Optics, Vol. 21, 18, p3393.	Surf	Wave	-
ZERN	Zernike coefficient. The Int1, Int2, Hx, and Hy data values are used to specify the Zernike term number (0-36), the wavelength number, the sampling density (1 = 8x8, 2 = 16x16, etc.), and the field position, respectively.	Term	Wave	See left
ZPLM	Used for optimizing numerical results computed in ZPL macros. See the section "Optimizing with ZPL macros".	Macro #	Data#	-
ZTHI	This operand controls the variation in the total thickness of a range of surfaces over multiple configurations. It is similar to the TTHI operand, except it is an inequality operator. The target value specified is the maximum allowed difference between the TTHI at each defined configuration. For example, if there are 3 configurations where TTHI 3 8 would evaluate to 17, 19, and 18.5, respectively, ZTHI will return 2 (i.e. 19-17) if the target is less than 2. Otherwise, ZTHI returns the target value. To keep all zoom configurations the same length, use a target of 0.	First surf	Last surf	-

The operational operands (SUMM, OSUM, DIFF, PROD, DIVI, SQRT) along with the parametric operands (CVGT, CVLT, CTGT, CTLT, etc..) can be used to define very general and complex optimization operands, as discussed in the section "Defining complex operands", which can be found later in this chapter.

Because of the dimensional differences between parameters such as effective focal length (tens of millimeters or more) and RMS spot size (microns), usually a weighting of one is sufficient for quantities measured in lens units. However, the residual value of the effective focal length with this weighting is not likely to be zero. Increasing the weighting will bring the resulting system closer to the desired effective focal length. This effect is often

noticeable when defining ETGT (edge thickness greater than) operands. Usually ETGT with a target of zero will often yield a value just slightly less than zero. Rather than increase the weight, it is much simpler to provide a target value of .1, or some such number.

After making changes to the operand list, the current values of each operand can be updated by selecting Tools, Update. This is also useful for checking to see what the current values of each operand are, and which has the greatest contribution to the merit function. The percent contribution is defined as

$$\%contrib_i = \frac{100 W_i (V_i - T_i)^2}{\sum W_j}$$

The merit function is automatically saved with the lens file.

## Understanding boundary operands

---

The boundary operands such as MNCT, CTGT, DIMX, and others behave somewhat differently than specific target operands such as TRAR and REAY. When you specify a boundary on a parameter, you specify the target value as the definition of the boundary. For example, to maintain a minimum center thickness on surface 5 of 10 mm, you might use a command such as CTGT 5 10 (where the 5 is in the Int1 column and 10 is in the target column). If you update the merit function and then observe the "value" column of that operand, there are two possibilities for the value: 1) if the boundary is violated, that is, the center thickness is less than 10, then the actual value of the thickness will be displayed, or 2) if the boundary is not violated, that is, the center thickness is greater than 10, then the value 10 will be displayed.

The rule is simple- if the boundary is violated, the actual value is shown; if it is not violated, the value is set to the target and is therefore ignored by the optimization algorithm. If during optimization the boundary becomes violated, then the value will automatically be updated and the optimization algorithm will attempt to correct the offending parameter.

The boundary operands which constrain a range of surfaces are slightly more complicated. These multiple surface operands return values which represent the total effect of all violated boundaries within the specified surface range. For example, the operand MNCT 1 10 will constrain the minimum center thickness of surfaces 1 through 10. If the target is 3.0, which defines the boundary, then the difference between the value of the operand and the target is the sum of the difference between 3.0 and the thicknesses of all surfaces between 1 and 10 whose center thickness is less than 3.0. If only one surface in the range has a center thickness less than 3.0, say 2.5, then the operand has a value of 2.5. If a second surface is added that has a thickness less than 3.0, say 2.2, then the operand will have a value of 1.7 (2.5 minus .8; the .8 is 3.0 - 2.2). The total difference between the target of the operand and the value is 3.0 - 1.7 or 1.3. This difference of 1.3 is due to the violation of 0.5 by the first surface and another 0.8 by the second surface.

If calculating the value of these boundary operands seems confusing, don't worry; ZEMAX does all the calculations for you. All you need to do is to specify the boundary type (such as MNCT or MNET) the boundary range (surface 1 through 10, or whatever) and the desired value (3 mm or whatever). If all the boundary constraints are met, then the operand value is equal to the target, otherwise, the value will be different and the merit function will increase. The increased merit function will cause the optimization algorithm to seek a reduction of the operand contribution.

If a boundary operand does not seem to work, there are several things to check:

- 1) Make sure the variables you have defined can have some effect on the boundary operands. A common mistake is to specify MNCT and have some "frozen" thickness within the surface range. If the thickness violates the boundary and it is not variable, ZEMAX can't fix it. The operands DO NOT ignore violated but frozen boundaries.
- 2) If there is a small residual error, try increasing the boundary value. For example, if MNCT is used with a target of 0.0, and the value is a small number (like -.001) the problem is not that the operand doesn't work, it is simply that the residual error is too small to increase the merit function significantly. It is usually better to increase the target to 0.1, or some other number, rather than to increase the weight. Increasing the weight will only lead to a smaller violation (like -.0000001) rather than meeting the boundary.

3) Check to see if there is a reasonable contribution to the merit function. You can easily check this with the percent contribution column. By looking at the percent contribution column, verify that the operand in question has enough influence on the total merit function. If it does not, increase the weight, or see the preceding paragraph for advice on changing the target.

Understanding the boundary operands is a crucial part of mastering ZEMAX optimization, and with a little practice you will find them to offer excellent control and flexibility.

## Using MTF operands

These operands are only supported in the XE and EE editions of ZEMAX.

The MTF operands such as MTFT, MTFS, and MTFA provide a capability to directly optimize the diffraction MTF. This is a powerful capability, however, using the MTF operands requires some care on the part of the user.

For systems not close to the diffraction limit, geometric equivalent MTF operands are provided: GMTT, GMTS, and GMTA. These should be used in place of the diffraction operands for systems with greater than about 2 waves of aberration.

The MTF operands calculate the complete diffraction or geometric MTF exactly like the plots available on the Analysis, Diffraction menu option. Therefore, any optical system for which the MTF plot produces invalid data (due to excessive OPD in the pupil, see the chapter "Analysis" for details) will also produce meaningless data during optimization. For example, it is unrealistic to optimize a lens for MTF starting from plane parallel plates, because of course the MTF generally cannot be accurately computed for such a system. Also, MTF optimization is considerably slower than RMS spot size or RMS wavefront error optimization, typically 5 to 50 times slower. MTF optimization should never be used during global optimization for this reason. Note that if you use both a MTFT and a MTFS operand, the MTF is computed twice, even if the spatial frequency is the same. Therefore, use the MTFA operand instead, if possible. If the sampling is too low for accurate computation of the MTF, then the MTF operands return zero rather than a meaningless number.

The slow execution speed may be noticeable when updating the Merit Function Editor display, and entering and exiting the optimization dialog box. At these times the merit function is updated. If several MTF operands have been entered, ZEMAX may take several minutes on a slow computer just to refresh the screen.

A good approach is to interactively design your system using RMS wavefront error. Generally speaking, systems with low RMS wavefront error will have reasonable MTF performance. After the design is very nearly in the final form, then try switching over to MTF optimization for a "touch-up". It is also a good idea to completely erase the default merit function before entering any MTF operands, except of course for any boundary constraints needed (such as MNCT).

The MTF operands use the operand data columns such as Int1, Int2, and Hx differently than most operands. The Int1 column determines the sampling density to be used. A value of 1 specifies an 8x8 grid should be used, 2 specifies a 16x16 grid, and so on. Use the smallest grid size for which the data are computed accurately; again see the "Analysis" chapter for details.

The Int2 column selects the wavelength, like most operands. However, a value of zero may be entered to specify a polychromatic calculation. In this case, the wavelength-weighted polychromatic MTF is computed. Obviously, this is slower than a monochromatic MTF calculation.

The Hx column is used to specify the field position, and it must be an integer between 1 and the number of defined fields.

The Hy column is the spatial frequency in cycles per millimeter, independent of the lens system units. Any value may be entered; the operand will return 0 if it is past the incoherent cut-off frequency. The value does not need to be an integral multiple of some fraction; the exact value of the MTF is computed using a cubic spline fit to the neighboring data points, just like the MTF data plots.

The target and weight columns are used just like any other operand. Of course, if the target is set to 1.0, and the frequency is non-zero, then the value of the operand can never reach the target.

## Performing an optimization

---

To begin optimization, choose Tools, Optimization from the main menu bar. The optimization control dialog box will appear with the following options.

### OPTIMIZATION OPTIONS

Item	Description
Automatic	Executes until ZEMAX determines the system is no longer improving significantly.
1 Cycle	Executes a single optimization cycle.
5 Cycles	Executes 5 optimization cycles.
10 Cycles	Executes 10 optimization cycles.
50 Cycles	Executes 50 optimization cycles.
Terminate	Terminates a running optimization, and returns control back to the dialog box.
Exit	Closes the optimization dialog box.
Auto Update	If checked, ZEMAX will automatically update and redraw all open windows at the end of each optimization cycle. This allows monitoring of the optimization progress using any analysis feature.
# CPU's	Selects the number of CPU's over which to spread the optimization task. More than 1 may be selected, even on a single CPU computer, in which case the single CPU will time share the multiple simultaneous tasks. The default is the number of processors detected by the operating system.

Selecting automatic will cause the optimizer to run until no progress is being made. The other options will run the specified number of cycles. Automatic mode is highly recommended. The time required to run a given optimization cycle varies enormously with the number of variables, the complexity of the system, the number of solves, the number of operands, and of course the computer speed. If the cycle is taking too long, or if it appears to be hung up, or if you feel the design is not progressing adequately, click on Terminate to end the optimization run.

When the optimization begins, ZEMAX first updates the system merit function. If any of the operands cannot be computed, the optimization cannot begin, and an error message will be displayed. Operands cannot be computed if they require the tracing of rays which miss surfaces or which undergo total internal reflection (TIR) at an index boundary. If such an error message appears, usually the starting lens prescription is in error, or the ray targets are incorrectly defined (this will not happen with the default merit functions, but might happen with user defined rays). ZEMAX can automatically recover if the merit function cannot be evaluated during the course of optimization; only the starting system need be adequate to compute all operands in the merit function.

## Defining complex operands

---

Although the default merit function, coupled with a few predefined operands, is perfectly suitable for the majority of optical designs, there are times when an unusual constraint needs to be added to the merit function. Rather than define a very large number of very specific operands, ZEMAX allows you to build your own operands out of simple building blocks.

ZEMAX allows very general operand definitions. There are two tricks to creating these operands. First, use certain operands with zero-weighting to define the parameters you need, and second, use the operational operands to define relationships between them. For example, suppose you require that the thickness of surface 3 and the thickness of surface 4 sum to 10. There is an operand that does this, TTHI. The command structure would look like this:

Number	Type	Int1	Int2	Target	Weight
1	TTHI	3	4	10	1

However, for illustration only, note that there is an alternate way of calculating the same thing:

Number	Type	Int1	Int2	Target	Weight
1	CTVA	3		0	0
2	CTVA	4		0	0



Number	Type	Int1	Int2	Target	Weight
3	SUMM	1	2	10	1

Operand 1 uses the Center Thickness VALue (CTVA) command to extract the value of the thickness of surface 3. Similarly, operand 2 is used to extract the thickness of surface 4. The zero weighting on both operands insures that the optimization algorithm ignores the constraint; it is only used as an intermediate step. Operand 3 now sums two operands: number 1 and number 2. The result is the sum of the thicknesses of surfaces 3 and 4 is the value of operand 3, and this has a non-zero weight. The optimization algorithm will attempt to drive the sum to 10.

Why go to all the trouble of this three-step process if a single TTHI command would do the same thing? The reason is that this approach can be extended to develop very general operands. For example, suppose you wanted the radius of curvature of surface 5 to be centered on the vertex of surface 8. Study the following commands to see if you understand how this is done:

Number	Type	Int1	Int2	Target	Weight
1	CVVA	5		0	0
2	TTHI	5	7	0	0
3	PROD	1	2	1	1

The CVVA command extracts the curvature of surface 5, the curvature we want to control. TTHI 5 7 calculates the distance from surface 5 to surface 8 (note we only sum to surface 7 to get to surface 8, since the thickness of surface 8 gives the distance to surface 9). Since the curvature of the surface is the reciprocal of the radius, the product of the curvature and the distance must be one; hence the target is 1 for operand 3. Operand 3 is also the only weighted operand in the sequence.

Now consider the requirement that the thickness of surface 5 must be greater than twice the radius of curvature of surface 4 plus the conic constant of surface 2 (this is nonsensical, but illustrative of the flexibility in the approach):

Number	Type	Int1	Int2	Target	Weight
1	CTVA	5		0	0
2	CVVA	4		0	0
3	CONS			2	0
4	DIVI	3	2	0	0
5	COVA	2		0	0
6	SUMM	4	5	0	0
7	DIFF	1	6	0	0
8	OPGT	7		0	1

Operand 1 extracts the (center) thickness of surface 5. Operand 2 extracts the value of the curvature of surface 4. Operand 3 sets a constant of two, and operand 4 divides the value 2 by the curvature (yielding twice the radius of curvature). COVA extracts the conic, and SUMM adds operands 2 and 4. Operand 7 takes the difference of the thickness and twice the radius plus the conic. Since we want the former to exceed the latter, we set an operand greater than constraint; the only one to have a non-zero weighting.

## Optimizing glass selection

Optimization of glasses is handled somewhat differently than other data. Optimizing the glass choice directly is a difficult and unpredictable process because there does not exist a continuum of glasses on the glass map. There are two methods for dealing with this problem. The first method is to idealize the glass dispersion using a few simple numerical values, and then optimize the numerical values while constraining the values to be similar to available glasses. The second method is to directly alter the glass types, and then reoptimize to see if the new glasses yield a better solution.

The first method is used by the conventional optimizer described in this chapter; the second method is described in the Chapter "Global Optimization". The global optimization method for glass selection is vastly superior.

ZEMAX can idealize the dispersion of glass using the index at d-light (.5875618 microns), the Abbe number, and a term which describes the deviation of the partial dispersion from the "Normal Line". The index at d-light is given the symbol  $N_d$ . The Abbe number (also called the V-number) is given the symbol  $V_d$  and is defined by

$$V_d = \frac{N_d - 1}{N_F - N_C},$$

where  $N_F$  and  $N_C$  are the indices of refraction at .4861327 and .6562725 microns, respectively. The partial dispersion term is  $\Delta P_{g,F}$ .

ZEMAX uses a formula based upon the typical dispersion of standard glasses to estimate the index at any defined wavelength as a function of the  $N_d$  and  $V_d$  values. This formula is accurate to roughly .0001 for typical glasses.

The  $N_d$ ,  $V_d$ , and  $\Delta P_{g,F}$  values are specified on the glass solve dialog box, which can be reached from the Lens Data Editor. To optimize glasses then requires several steps. First, change the glass of the appropriate surface to a "Model" glass using the glass solve dialog box in the Lens Data Editor. When you change the glass from "Fixed" to "Model", ZEMAX will make a suitable guess for the index, Abbe number, and partial dispersion; you only need to change the values if you wish. The three values can be made variable by clicking on the "Vary" box next to each entry.

The Ctrl-V shortcut on the glass column will also work; it makes the index, Abbe, and partial dispersion variable automatically. The model glass data values can now be optimized using the optimization feature in the usual way.

Unconstrained glass optimization usually will lead to very high index materials being selected. This is because surfaces with high refractivity (a large difference in index across the boundary) need less curvature than low refractivity surfaces to have the same optical power. Lower curvature surfaces introduce less aberration.

Unfortunately, high index materials are expensive, heavy, harder to fabricate, and may be brittle, delicate, or susceptible to stains and scratches. Also, very high index materials do not always exist; there are few glasses (for the visible spectrum) available with an  $N_d$  higher than about 1.9. The  $V_d$  value also is limited to the range of roughly 20 to 80. Therefore, it is essential to limit the  $N_d$  and  $V_d$  values to reasonable ranges during optimization. The partial dispersion deviation also must be limited in range.

There are two ways to limit the  $N_d$ ,  $V_d$ , and  $\Delta P_{g,F}$  values. The simplest way is to add the RGLA operand somewhere in the operand list. The RGLA operand measures the "distance" on the glass map between the index, Abbe number, and partial dispersion of the model glass to the closest glass in the currently loaded catalogs. For example, if you are optimizing the index and Abbe and you have specified that the Schott and Hoya catalogs are used (these are specified on the general data window), the RGLA operand computes the "distance" to each glass in these catalogs. If the smallest "distance" is less than the target value specified for the RGLA operand, then the boundary condition is met, and the value of the operand is equal to the target. If the closest glass is farther than the target value, then the RGLA value is the actual "distance". The "distance" is defined by scaling the Abbe number by 0.01, and the deviation of the partial dispersion by 10. The "distance" between any two glasses is given by

$$d = \left[ (nd_1 - nd_2)^2 + 10^{-4} (Vd_1 - Vd_2)^2 + 10^{+2} (\Delta P_{g,F1} - \Delta P_{g,F2})^2 \right]^{1/2}.$$

The best way to use RGLA is to specify the surface range that covers all of the surfaces you are optimizing. For a target value, start off with 0.05. This will allow the glasses to easily move all over the glass map, since the spacing between various glasses is usually less than 0.05. After optimization, decrease the target to roughly 0.02 and reoptimize. This will encourage the optimized system to choose index and Abbe numbers reasonably near actual glasses.

The other method for constraining index and Abbe values is to use the MNIN, MXIN, MNAB, and MXAB controls. These operands are mnemonics for Minimum and Maximum Index and Abbe values, and they are documented in the preceding tables. These operands can be used to restrict the optimization to specific rectangles on the glass map. It may be useful to use RGLA along with MXIN, for example, to restrict the glass selection to existing glasses with an index lower than some value.

At some point you will want to convert your variable index data back to a real glass. There will usually not be a perfect match between the optimized  $N_d$  and Abbe values and those of an actual glass in the current catalog. However, ZEMAX will search through the catalog and find the "best fit" glass using a least-squares criterion similar

to the RGLA definition above (the partial dispersion term is omitted). The glass in the catalog which differs the least from the variable index parameters is the glass selected. This glass is also reported on the "Surface Data Summary" feature (select Reports, Surface Data). The index of refraction data shown is that calculated from the Nd and Abbe values, not the best fit glass. After converting from a model glass to a real glass, another optimization run is generally required. For systems with delicate chromatic aberration balancing, the best glass choice might never be found using variable glasses, simply because the model glass dispersion is never identical to the dispersion of a real glass.

See the discussion in the Chapter "Global Optimization" for a discussion on an alternate method for optimizing glass selection.

## Optimizing zoom and multi-configuration lenses

---

Optimizing zoom lenses is virtually identical to optimizing conventional single-configuration lenses. See the Chapter "Multi-Configurations" for details.

## Optimizing extra data

---

This discussion is only relevant for users of ZEMAX-EE.

Certain surface types supported by ZEMAX-EE, such as the Zernike, Zernike phase, extended polynomial, and binary optic surface use the extra data values. These extra data values can be edited, loaded from ASCII files, and may be used as variables for optimization. For details on the editor, see the chapter "Editors menu".

To make an extra data value a variable, open up the Extra Data Editor. When the extra data editor appears, move the cursor to the row with the value you want to optimize, and press Ctrl-V (the same command used to set variables on the main screen). The variable will now be optimized when the optimization is run.

There are also several boundary constraints for use with the extra data values. XDVA, XDGT, and XDLT are extra data value, greater than, or less than, respectively. The Int1 column on the spreadsheet display indicates the surface number to which the operand applies, and the Int2 column is used to specify which of the extra data values is to be used.

## Using gradient index operands

---

There are several optimization operands which are used to control the properties of gradient index materials during optimization. Some of them are described below.

### DLTN

DLTN is used to control the maximum total change in index within a gradient index lens. Int1 is used to define the surface number, and Int2 is used to define the wavelength number. DLTN is defined as:

$$DLTN = n_{\max} - n_{\min}$$

The min and max index values are computed at the extreme z coordinates, Z min and Z max. Z min and Z max are the Z coordinates of the minimum and maximum axial positions of the blank used to make the lens, before the shaping begins. For a convex surface, they correspond to the vertex. For a concave surface, they correspond to the maximum sag at that surface.

### LPTD

LPTD is used to control the profile of the gradient within the material. Only the Int1 value is used to define the surface number of the gradient index surface. LPTD is an acronym for LightPath Technology Delta, and the constraint is used to keep a nonlinear profile monotonically increasing or decreasing. It only needs to be used when the quadratic or cubic term of the axial gradient is variable. This operand only affects GRIN 5 surface types.

The LPTD operand should be used with a target of 0. The boundary constraint enforces the following conditions:

$$\frac{\partial n}{\partial z_{\min}} > 0 \text{ and } \frac{\partial n}{\partial z_{\max}} > 0, \text{ or } \frac{\partial n}{\partial z_{\min}} < 0 \text{ and } \frac{\partial n}{\partial z_{\max}} < 0$$

Z min and Z max are the Z coordinates of the minimum and maximum axial positions of the blank used to make the lens, before the shaping begins. For a convex surface, they correspond to the vertex. For a concave surface, they correspond to the maximum sag at that surface. If the residual value of the operand is less than zero, then the target may be decreased slightly (try 0.1). Changing the target is usually more effective than increasing the weight. The value of the LPTD operand must be zero for the blank to be fabricated. Always check the gradient profile to make sure the slope does not change sign.

## Optimizing with ZPL macros

This discussion is only relevant for users of ZEMAX-EE.

There are times when very complex calculations need to be performed, and the results of the computation need to be optimized. ZEMAX supports some of these calculations already, such as the MTF A operand which traces many rays, computes the MTF, and then returns a single resulting number to the Merit Function Editor "value" column. Some limited calculations can be performed within the merit function itself; see for example the discussion in the section "Defining complex operands" earlier in this chapter.

However, there are problems for which only the flexibility of the ZPL macro language is sufficient to perform the required computations. For these problems, the operand ZPLM may be used to call a ZPL macro from within the merit function. The macro performs the required computations, then returns the result using the ZPL OPTRETURN keyword.

ZPLM is simple to use. The Int1 and Int2 values are used to specify the macro number and data field number, respectively. The macro number is used to indicate which ZPL macro should be executed, while the data field number indicates which value computed by the macro should be optimized.

The macro number must be an integer between 0 and 99. If the Int1 value for a ZPLM operand is set to 17, for example, then the macro number is 17, and the macro to be executed must be named ZPL17.ZPL. The macro name must always use a two digit representation of the macro number. If the macro number was 6, then the macro to be executed would be ZPL06.ZPL. The ZPL macro file must reside in the default directory for ZPL macros; see the chapter "File Menu" for details.

The data field number may be any number between 0 and 25. This number refers to a position in a global array associated with the lens in memory. During execution of the macro, the macro keyword OPTRETURN specifies which data field number stores the results of the macro calculation. There are 26 different data fields, so that a single macro call can be used to optimize up to 26 different values simultaneously. For example, suppose you needed a macro which computed the total length of the lens from surface 1 to the image plane (this is in effect a user-defined version of the TOTR operand). The macro might look like this:

```
n = nsur()  
x = 0  
for i = 1, n, 1  
    x = x + thic(i)  
next  
optreturn 0 = x
```

Note the use of the OPTRETURN keyword. This keyword stores the resulting value for "x" in the global array position 0. Suppose this macro was named ZPL15.ZPL. To optimize the resulting value for x, the ZPLM merit function operand would be added to the Merit Function Editor, with Int1 = 15 and Int2 = 0. After updating the merit function, the "value" would be the same as that returned by TOTR, and it can be optimized in the same way.

ZPLM also permits the use of the Hx, Hy, Px, and Py data fields. These data fields can be read by the ZPL macro using the PVHX, PVHY, PVPX, and PVPY ZPL functions, respectively. "PV" is a mnemonic for "Pass Value".

There is one very important thing to know about the data field number. If it is zero, then the macro is executed and the value from OPTRETURN 0 is returned. However, if the data field number is not zero, then the macro is not executed, but any previous value stored from an earlier call to the macro is used instead. The advantage to this convention is substantial. If the macro computes many values, all of which need to be optimized, the macro only needs to be called once, yet multiple ZPLM operands can access the data. This is much more efficient than calling the macro multiple times.

For example, suppose a macro named ZPL11.ZPL computes three values, all of which require optimization. In the macro, the values are stored using OPTRETURN:

```
OPTRETURN 0 = x
OPTRETURN 1 = y
OPTRETURN 2 = z
```

Then three ZPLM operands in the merit function can extract the data and perform the optimization with a single call to the macro:

```
ZPLM 11 0
ZPLM 11 1
ZPLM 11 2
```

The macro ZPL11.ZPL is only called during the evaluation of the ZPLM 11 0 operand. Note the Hx, Hy, Px, and Py values can only be used if the Int2 value is zero, since only in this case is the macro evaluated.

Lastly, it is very important that no changes be made to the lens data during the macro execution. These changes may interfere with the subsequent evaluation of other operands. ZEMAX does not restore the lens being evaluated to the state it was in prior to the evaluation of the ZPLM specified macro. Also, ZPLM should not be used in the middle of a default merit function, but should instead be placed either at the beginning or the end of the merit function definition.

## Suggestions for use

---

If you have a slower computer, you can greatly decrease the computation time by limiting the number of fields and wavelengths used during optimization. In preliminary design stages, it is rarely required to trace all of the rays for all of the wavelengths. Roughly the same effect can be had by deleting some of these rays. Using only three field positions is also reasonable for early work. A numeric coprocessor is essential for reasonable performance.

There are a few other tricks to improve performance. Avoid setting boundary operands on variables unless the optimized solution persists on implausible designs. Boundary operands add computational overhead. Use solves instead of explicit operands whenever possible. For example, use a curvature solve to control the focal length rather than an operand if possible.

Optimization is inseparable from the art of modern lens design, and only practice will make a designer a proficient user of optimization algorithms. Users who are expert at other software optimization algorithms will probably find ZEMAX easier to use, and with a little practice, the mechanics of using the interface will slip into the subconscious, and the designer can concentrate upon the design itself. If you are new to computerized optimization of lenses, there is no better way to learn than to practice.

The design that yields the lowest value of the merit function is called the "global" optimum and is by definition the best possible design. However, there is no known optimization algorithm that can universally find the global optimum, unless you consider "direct search" an optimization algorithm (in other words, try all of the infinite number of possible solutions to see which is best). The art of optical design with computer assistance has two basic components. First, the designer must be able to determine a suitable starting design, and second, he or she must play the role of supervisor during the optimization process. A good supervisor knows when and how to back up and coax the program into a more fruitful direction. Unfortunately, this often requires considerable experience, and even more often, excessive tedium. An experienced designer uses a combination of intuition, analysis, and luck in searching for new, better, design forms. ZEMAX-XE and EE provide an automated capability for performing this global optimum search; the feature is described in the next chapter.

This page intentionally left blank.

## Introduction

---

This feature is only available in the XE and EE editions of ZEMAX. The material presented in this chapter critically depends upon the user having read and understood the material presented in the previous chapter on "Optimization".

The conventional means of optimizing lenses has for decades been the use of the damped least squares algorithm (DLS). DLS has many attractive features; it is efficient, and it is very good at finding the "local" minimum of the merit function. In this context, the word local means the lowest value of the merit function that can be reached from the current position in solution space without ever increasing the merit function (this is something of an idealization, in reality DLS can hop over small regions of increased merit function).

To visualize this problem, imagine you are on a hike, and are trying to find the bottom of a valley from a starting point on the side of a hill. You would like to find the lowest point in the valley. Suppose you cannot see the valley; it is very foggy and all you can see is the terrain very close to where you now stand. You can determine which way is downhill, and proceed in that direction until the slope becomes uphill again; at which point you find a new downhill direction. You might repeat this procedure over and over until a point is reached where all directions are uphill. This lowest point is then a local minimum, and the bottom of at least this valley.

The problem with this method is that once you have arrived at the local minimum, there is no known way (for the general optimization problem) to determine if there is not a better, lower minimum somewhere else. For example, if you were to walk uphill in any direction from this point until you reached a local peak, and then were to proceed onward downhill into the next valley, you will eventually reach a new local minimum. Is this new minimum lower or higher than the previous valley? The only way to find out is to take the hike!

You might ask, since computers are so fast, why not just try out every possible configuration to see which is best? To get a feel for the scope of the problem, consider a simple cemented doublet lens with six degrees of freedom (degrees of freedom manifest themselves as variables for optimization). If you assume that each variable can take on 100 possible values (a coarse sampling), then there are  $1\text{E}+12$  different possible systems. If each system evaluation requires the tracing of 20 rays (a low estimate) and you can trace 1,000,000 rays per surface per second, then the time required is about  $8\text{E}+07$  seconds, or about 2.5 years. For a four-element lens (16 variables) evaluated at three fields and three wavelengths, using 100 rays for evaluation would require  $1\text{E}+32$  system evaluations or many billions of times the age of the universe.

There are approaches to the global optimization problem that (thankfully) do not require unreasonable computational effort. These algorithms include simulated annealing, multistart, expert systems, neural networks, and others. All of these algorithms have strengths and weaknesses which are beyond the scope of this chapter.

## Capabilities of ZEMAX

---

There are two separate global optimization algorithms in ZEMAX, each with a different purpose. The first algorithm you are likely to use is called "global search", and it is used to find new design forms given only the merit function and a starting design. Global search uses a combination of genetic algorithms, multistart, conventional damped least squares, and some expert system heuristics to search for new design forms. The global search algorithm is very good at finding promising design forms, however, it does not usually produce "finished" designs. The second algorithm is used for this purpose.

The second algorithm is called "hammer optimization" (lens designers often talk about hammering on a design to squeeze out the last bit of performance). The hammer algorithm is used for exhaustively searching for the optimum solution once a reasonably good starting point is found, presumably by prior experience or the global search algorithm. The hammer algorithm only requires a partially optimized lens and merit function in the form of a ZMX file.

Although the global optimization algorithms are extremely useful, it is important to realize that there is no guarantee that the true global optimum will always, or even occasionally be found. Of course, there is no way to even determine if any solution is the global optimum, even if it is the best you have ever found (remember the universe-lifetimes scale of the problem).

It is important to realize that there is no guarantee that the true global optimum will always, or even occasionally be found.

Both the global search and hammer algorithms require extensive computational effort to be effective. These algorithms are not intended to be used interactively! (that is what DLS optimization is for). If you set up global optimization, and watch while the computer works, you are bound to be disappointed. Global optimization is highly effective when you set the problem up and let the computer run for many hours, or even several days, but not for ten minutes. The ideal situation is to set up the problem before stopping work for the night, and let global search (or hammer, depending upon your requirements) work overnight. In the morning, you should have useful results to look at.

## **The global search algorithm**

---

Before you begin the global search, you must come up with a very rough starting point. "Very rough" means the design has the correct number of surfaces, a defined stop surface, and initial glasses selected. The fields and wavelengths must be defined. You also need to define a merit function; see the chapter "Optimization" for details on this procedure. The very rough design can be parallel plates of glass with a curvature solve on the last surface to control the focal length. If a solve is not used to control the focal length, then the system should have at least the approximate focal length desired. Also, the variable parameters must all be defined. The lens must be saved before initiating global search.

ZEMAX uses the starting focal length as a scaling parameter, so the initial design should have the correct focal length!

From the main screen, select Tools, Global Search. There are buttons on the dialog box labeled: Begin Search, Terminate, and Exit.

Select Begin Search. The search will begin, although nothing may appear to happen for several minutes. ZEMAX will begin looking at various combinations of lens parameters extracted from the ranges you have defined. After certain criteria are met, ZEMAX will then save the best lens it has found in a file called GLOPT\_01.ZMX. The cycle then repeats, and another lens file is generated. If this new lens found has a higher merit function, it will be saved as GLOPT\_02.ZMX. If it has a lower merit function, GLOPT\_01.ZMX will be renamed GLOPT\_02.ZMX, and the new file will be saved as GLOPT\_01.ZMX. The first ten files are generated in this way, sorted in order from GLOPT\_01.ZMX to GLOPT\_10.ZMX. ZEMAX will report the file name along with the corresponding merit function in the box on the lower right corner of the screen.

When the eleventh lens is generated, ZEMAX will compare the merit function of the new lens to the best ten found so far, and will place it in the correct location in the ten best list, renaming other lens files as required. If the lens has a higher merit function than GLOPT\_10.ZMX, then it is discarded. The cycle repeats indefinitely. Each time a new lens is found which is better than the worst in the list of ten, it is placed in the correct place in the list. After hundreds of lenses have been replaced (which may require many tens of thousands of lenses to be evaluated) the resulting set will hopefully contain some very good designs, or at least some promising forms.

The algorithm also periodically returns to lenses in the list of ten to see if they can be improved upon. Occasionally some lenses will be improved and placed back in the list of ten. If this happens, the older design being replaced is rejected if it has the same basic form as the new lens. This is done to keep some diversity in the list of ten. At no time is the algorithm throwing away good, different designs.

To terminate the search, select Terminate. Depending upon what the algorithm is doing, it may exit immediately, or it may require several seconds. Once the algorithm has terminated, you can click on Exit. You can now open any of the GLOPT\_01.ZMX through GLOPT\_10.ZMX files.

Global search will rarely find the global optimum by itself. The reason is that the global search concentrates the effort on finding new, promising design forms rather than converging exactly on the best possible solution for each form. This latter job is left to a separate algorithm, called "Hammer Optimization", described in the following section.

## **The hammer algorithm**

---

After reviewing the design forms generated by global search, you will probably want to investigate one or two of them. The one with the lowest merit function is not always the best (although it should be if you designed your



merit function well). For example, the second-best solution may be easier to fabricate. Whatever criterion you use to determine the most promising solution, you now want to find the best design possible using this selected lens as a starting point.

Hammer optimization will take the lens and exhaustively attempt to refine it by making adjustments and optimizations. Each time the lens is improved, it will be saved to disk in a file of the same name. Therefore, you may want to put your selected starting point into a new file before beginning the run. Hammer optimization only requires a ZMX file to get started. The hammer optimization screen shows the starting merit function, and the best merit function found so far. Although good results can occasionally be had in several minutes, the algorithm should be allowed to run for several hours, and preferably overnight. To terminate the search, select Terminate, then Exit.

The hammer algorithm can also be used effectively on partially optimized designs that were not generated with global search. Feel free to use the hammer on any design!

## Optimizing glass selection

---

If a glass is made into a "model" glass described by the index, Abbe, and partial dispersion deviation, then the parameters to the model may be made variable and optimized just like any other numerical parameters. However, the model glass method has one serious drawback. After a good solution is found using model glasses, a conversion must be made back from the model glass to a real glass. The design must then be reoptimized using the new glass selection.

Unfortunately, for many systems the newly optimized design will perform worse than the model glass design. Even more perplexing is the fact that the optimal design using real glasses may have a different form than that found using the model glass.

Traditionally, to find a better combination of glass types, the designer would locate alternate glasses on the glass map, substitute in the new selections, and reoptimize. If the new solution was better, the glass choices would be retained; otherwise, a new set of glasses would be evaluated. The procedure would continue as long as the designer could bear to continue the search.

ZEMAX automates this procedure by allowing glasses to have a "substitute" status associated with them. If a glass is marked (using the glass solve dialog box) as a substitute glass, then the global optimization algorithms (both Hammer and Global Search) automatically perform iterative substitution of similar glass types during optimization. This allows ZEMAX to optimize not only the numerical prescription values such as radius and thickness, but also allows direct optimization of real glass selection, without resorting to idealizing the glass dispersion.

To use the substitution feature, simply set the status of each glass free to change to "substitute" on the glass dialog box. Then invoke the global optimizer. ZEMAX will automatically alter the glass types during the search for better designs. To prevent certain glasses from being chosen during optimization, choose "Exclude Substitution" for the glass that should be avoided on the glass catalog dialog box; see the Chapter "Using Glass Catalogs" for details.

## Suggestions for use

---

There are several techniques for maximizing the performance of the global optimization algorithms (both global search and hammer):

- 1) If possible, place the stop on the first surface. If your entrance pupil is imbedded in the system, you can model this by using a dummy first surface for the stop and then use a negative thickness to get to surface 2. This enhances performance because ZEMAX does not need to calculate where the entrance pupil is. You can make this thickness variable if appropriate. This technique does not work well for systems which inherently have entrance pupil distortion, such as wide angle lenses.
- 2) Use a marginal ray angle solve on the last glass surface curvature to control the effective focal length instead of the EFFL operand. Dropping one variable decreases the dimensionality of the problem significantly, and the EFFL operator does require one additional ray trace, slowing down the merit function evaluation.
- 3) Use a marginal ray height solve (for zero ray height) on the last thickness before the image plane. Most lenses are well corrected for the .7 pupil zone on axis. You may want to use another pupil zone if your intuition guides you to. This solve will insure that every design generated by global search is in focus, just as the

curvature solve insures the correct focal length. These two solves together can improve the performance of the global search algorithm by several orders of magnitude. For hammer optimization, replace the solve with a variable to allow for optimal defocus.

4) Use the MNCT and MNET operands. These are essential for avoiding negative center and edge thicknesses. ZEMAX uses these boundary constraints to determine the appropriate range of each variable in the search. The solutions will wander into these unacceptable regions of solution space unless you specifically forbid it.

6) Keep your merit function as simple as possible. Using 2 or 3 rings in the default merit function construction (see the chapter "Optimization") is often a good idea. You can always go to more rings for the hammer optimization.

7) Be willing to run several long runs; perhaps one before starting a design problem to get ideas for design forms, a few more to explore various promising designs, and a last run to insure the design you have settled on cannot be improved (Hammer is particularly good at this last task).

8) Use substitute glasses rather than model glasses, especially when using Hammer. There is not much point in using Hammer on a design that uses model glasses, because ultimately the glasses will need to be converted back to real glasses and then optimized again. Substitute glasses are highly useful, especially late in the design process when the design form has been determined.

9) You may want to dim or turn off your monitor, or use a screen saver to protect the phosphor from burn-in if you plan on running the search for many days.

10) The hammer algorithm may be terminated and then restarted without significant loss of information. As the algorithm proceeds, it automatically saves any improved designs.

11) If the global search algorithm is terminated, the control dialog box must be closed before the search may be resumed. The list of best files is reset, and so any designs found in the previous search will be lost unless they are copied into files with another name.

## Summary

---

Finally, it is again stressed that the global optimum search feature may not ever find the global optimum to any particular problem. It is very good at finding alternative design forms that would have been tedious to discover by hand. It is a very powerful tool to add to your toolbox; it is not a substitute for optical design skills.

The global search algorithm does have a strong random component, and therefore no two runs will yield the exact same solutions every time. Sometimes the solutions will be worse, sometimes better, but usually they are just different for runs of similar duration.

This page intentionally left blank.

## Introduction

---

ZEMAX provides a simple to use but flexible and powerful tolerance development and sensitivity analysis capability. The tolerances available for analysis include variations in construction parameters such as curvature, thickness, position, and aspheric constants.

ZEMAX also supports analysis of decentration of surfaces and lens groups, tilts of surfaces or lens groups about any arbitrary point, irregularity of surface shape, and variations in the values of any of the parameter or extra data. Since the parameter and extra data terms may describe aspheric coefficients, gradient index coefficients, and more, any of these values may also be made part of the tolerance analysis. The various tolerances may be used in any combination to estimate alignment and fabrication error effects on system performance.

Tolerances are defined using simple operands, such as TRAD, which defines a tolerance on a radius. The tolerance operands are automatically saved with the lens file.

Tolerances may be evaluated by several different criteria, including RMS spot size, RMS wavefront error, MTF response, or by a user defined merit function. Additionally, compensators may be defined to model allowable adjustments made to the lens after fabrication. ZEMAX also allows limits to be placed on the change of a compensator.

Tolerances may be computed two ways:

**Sensitivity Analysis:** For a given set of tolerances, the change in the criteria is determined for each tolerance individually.

**Inverse Sensitivity:** For a given permissible decrease in performance for each tolerance individually, the tolerances are computed.

The sensitivity and inverse sensitivity analysis considers the effects on system performance for each tolerance individually. The aggregate performance is estimated by a root-sum-square calculation.

As an alternative way of estimating aggregate effects of all tolerances, a Monte Carlo simulation is provided. This simulation generates a series of random lenses which meets the specified tolerances. By considering all applicable tolerances simultaneously, accurate simulation of expected performance is possible.

The Monte Carlo simulation can generate any number of designs, using normal, uniform, or parabolic statistics.

## The basic procedure

---

Tolerancing a lens consists of these steps:

- 1) Define an appropriate set of tolerances for the lens. Usually, the default tolerance generation feature described in this chapter is a good place to start.
- 2) Add compensators and set allowable ranges for the compensators. The default compensator is the back focal distance, which controls the position of the image surface. Other compensators, such as image plane tilt, may be defined. Using only the back focus compensator greatly speeds up the tolerance process, because "Fast Tolerance" mode may be used.
- 3) Select an appropriate criteria, such as RMS spot size or MTF.
- 4) Select the desired mode, either sensitivity or inverse sensitivity.
- 5) Modify the default tolerances or add new ones to suit the system requirements.
- 6) Perform an analysis of the tolerances.
- 7) Review the data generated by the tolerance analysis, and consider the budgeting of tolerances. If required, return to step 5.

Details on this basic procedure will be provided in subsequent sections.

## Tolerance operands

A tolerance operand has a four letter mnemonic, such as TRAD for Tolerance Radius. Two integer values, abbreviated Int1 and Int2, are associated with the mnemonic to identify the surface or surfaces of the lens to which the tolerance applies. Some tolerance operands use Int1 and Int2 for purposes other than defining surface numbers.

Each tolerance operand also has a minimum and maximum value. These values refer to the maximum acceptable change from the nominal value.

The available tolerance operands are listed in the following table, and are described in detail below.

TOLERANCE OPERANDS

Name	Int1	Int2	Description
Surface tolerances			
TRAD	Surface #	-	Tolerance on radius in lens units
TCUR	Surface #	-	Tolerance on curvature in inverse lens units
TFRN	Surface #	-	Tolerance on radius in fringes
TTHI	Surface #	Comp #	Tolerance on thickness or position in lens units
TCON	Surface #	-	Tolerance on conic constant (dimensionless)
TSDX	Surface #	-	Tolerance on Standard surface x- decenter in lens units
TSDY	Surface #	-	Tolerance on Standard surface y- decenter in lens units
TSTX	Surface #	-	Tolerance on Standard surface tilt (TIR) in x in fringes
TSTY	Surface #	-	Tolerance on Standard surface tilt (TIR) in y in fringes
TIRX	Surface #	-	Tolerance on Standard surface tilt (TIR) in x in lens units
TIRY	Surface #	-	Tolerance on Standard surface tilt (TIR) in y in lens units
TIRR	Surface #	-	Tolerance on Standard surface irregularity
TPAR	Surface #	Parameter #	Tolerance on parameter value of surface
TEDV	Surface #	Extra Data #	Tolerance on extra data value of surface
TIND	Surface #	-	Tolerance on index of refraction
Element tolerances			
TEDX	First Surface	Last Surface	Tolerance on element x- decenter in lens units
TEDY	First Surface	Last Surface	Tolerance on element y- decenter in lens units
TETX	First Surface	Last Surface	Tolerance on element tilt about x in degrees
TETY	First Surface	Last Surface	Tolerance on element tilt about y in degrees
User defined tolerances			
TUDX	First Surface	Last Surface	Tolerance on user defined decenter x
TUDY	First Surface	Last Surface	Tolerance on user defined decenter y
TUTX	First Surface	Last Surface	Tolerance on user defined tilt x
TUTY	First Surface	Last Surface	Tolerance on user defined tilt y
TUTZ	First Surface	Last Surface	Tolerance on user defined tilt z

For each tolerance, a minimum and a maximum value are specified on the Tolerance Data Editor. The tolerances are described in detail below.

### **TRAD: Tolerance on radius**

Used to tolerance directly on the radius of curvature. Min and max are the extreme errors in lens units.

For example, if the nominal radius of a surface is 100 mm, and the min and max TRAD values for that surface are -.50 mm and +.50 mm, then the tolerance analysis will be performed with the radius of the surface set to 99.50 mm and 100.50 mm.

### **TCUR: Tolerance on curvature**

Used to tolerance in units of curvature, which is directly related to power. Min and max are the extreme errors in inverse lens units.

For example, if the nominal radius of a surface is 100 mm, then the nominal curvature is 0.01 inverse mm. If the min and max TCUR values for that surface are -.001 and +.001 inverse mm, then the tolerance analysis will be performed with the radius of the surface set to 111.11 mm and 90.909 mm.

### **TFRN: Tolerance on fringes**

Fringe tolerances are very useful when tolerancing flat or large radii surfaces. Min and max are the extreme errors in (dimensionless) fringes at the primary wavelength. The change in the sag of a surface for small changes in curvature is given approximately by

$$\Delta Z = \frac{r^2}{2} \Delta C,$$

where r is the radial aperture of the surface. The change in the sag is related to the error in fringes by

$$\Delta Z = \frac{\lambda}{2} N,$$

where N is the number of fringes. The factor of one half assumes a double pass Newton's rings type test. For more information see Malacara, Optical Shop Testing.

### **TTHI: Tolerance on thickness**

TTHI is used to tolerance both absolute positions of elements as well as thicknesses of lenses within element groups.

By default, it is assumed that all variations in thickness affect only that surface and any surfaces in contact with that element. For example, if the first lens in a contact doublet has a +1.0 mm change in thickness, the front and rear vertex of the second lens both shift by +1.0 mm.

However, since ZEMAX defines the position of all surfaces by using an offset from the previous surface, simply adding 1.0 mm to the surface will shift all subsequent lenses in the system by +1.0 mm. What is more likely to occur in fabrication is that the +1.0 mm offset would be absorbed by the first air space after the lens group. TTHI can handle this case by allowing a "compensator" surface to be specified. By default, the compensator surface is specified as the first air space which follows the surface being tolerated.

To illustrate, imagine a lens where surface 3 was made of BK7, and surface 4 was made of F2, and surface 5 was air. The nominal thicknesses are 3, 4, and 6 mm, respectively. If a TTHI was defined by the default tolerance algorithm for surface 3, a compensator would be placed on surface 5. If the tolerance value was +.1 mm, then during analysis the thicknesses would be changed to 3.1, 4.0, and 5.9, respectively. Thus, the absolute positions of surfaces 6 through the image plane are unaffected by the change in thickness on surface 3.

The compensator is optional; to disable it, set the compensator to the same surface number as the tolerance, such as TTHI 3 3. For some lens systems, such as those that are assembled by stacking spacers in a tube, the compensator may not be desired.

Int1 is used to define the surface number, Int2 in the compensating surface number, unless Int2 is equal to Int1. Min and max are the extreme errors in lens units.

### **TCON: Tolerance on conic**

TCON is used to define a tolerance on a conic constant. Min and max are the extreme errors, dimensionless.

### **TSDX, TSDY: Tolerance on surface decenters**

TSDX and TSDY are used to analyze decentration in X and Y respectively of a Standard surface type. The Int1 value indicates the number of the surface, and this surface must be a Standard surface type. Surfaces other than Standard surface types may be tolerated using the TEDX and TEDY operands described later. The min and max values are the decentering in lens units.

The analysis of TSDX and TSDY uses the Irregular surface type. See the discussion "Tolerancing with the Irregular surface type" later in this chapter.

### **TSTX, TSTY: Tolerance on surface tilts**

TSTX and TSTY are used to analyze tilts of a Standard surface along the X and Y axes, respectively. The Int1 value indicates the number of the surface, and this surface must be a Standard surface type. Surfaces other than Standard surface types may be toleranced using the TETX and TETY operands described later.

TSTX and TSTY can be used to specify a tolerance on total indicator runout, or TIR, which measures the amount of "wedge" in a lens. See the description of the related TIR operands TIRX and TIRY.

The min and max values are the tilt in units of primary wavelength fringes measured at the maximum radial aperture of the surface where the maximum radial aperture is defined by the semi-diameter of the surface. Unlike the TETX operand, TSTX measures tilt along the x direction of the lens (TETX measures tilt about the x axis, which changes the tilt along the y direction). ZEMAX assumes fringes are measured in a double pass Newton's rings type test. The change in sag as a function of the x or y normalized coordinate for W fringes of tilt is then given by:

$$\Delta Z_x = \frac{\lambda_p W}{2} \rho_x \text{ or } \Delta Z_y = \frac{\lambda_p W}{2} \rho_y .$$

For example, if the semi-diameter of a lens is 10 mm, the TSTX tolerance is 1 fringe, and the primary wavelength is 0.500 microns, then the tangent of the tilt angle is  $0.00025 / 10.0 = 0.25$  microradians about the y axis. The change in sag at the maximum x aperture of the lens is 0.25 microns. A similar discussion applies to TSTY.

The analysis of TSTX and TSTY uses the Irregular surface type. See the discussion "Tolerancing with the Irregular surface type" later in this chapter.

### **TIRX, TIRY: Tolerance on surface TIR**

TIRX and TIRY are used to analyze tilts of a Standard surface along the X and Y axes, respectively. The Int1 value indicates the number of the surface, and this surface must be a Standard surface type. Surfaces other than Standard surface types may be toleranced using the TETX and TETY operands described later.

TIRX and TIRY are used to specify a tolerance on total indicator runout, or TIR, which measures the amount of "wedge" in a lens.

The min and max values are half the amount of "sag" in lens units measured at the maximum radial aperture of the surface where the maximum radial aperture is defined by the semi-diameter of the surface. The change in sag as a function of the x or y normalized coordinate for a TIRX or TIRY value is given by:

$$\Delta Z_x = \frac{TIRX}{2} \rho_x \text{ or } \Delta Z_y = \frac{TIRY}{2} \rho_y .$$

For example, if the TIRX tolerance is 0.10 mm, then the change in sag at the maximum +x aperture of the lens is 0.05 mm, and the deviation at the maximum -x aperture is -0.05 mm, for a "total" TIR of 0.10 mm. A similar discussion applies to TIRY. The analysis of TIRX and TIRY uses the Irregular surface type. See the discussion "Tolerancing with the Irregular surface type" later in this chapter.

### **TIRR: Tolerance on surface irregularity**

TIRR is used to analyze irregularity of a Standard surface. The Int1 value indicates the number of the surface, and this surface must be a Standard surface type. Analysis of irregularity on surface types other than Standard is not directly supported.

Modeling irregularity is somewhat more problematic than other types of tolerances. This is primarily because irregularity by nature is random, and not deterministic such as a change in radius. Therefore, some assumptions about the nature of the irregularity need to be made in order to perform the analysis. The assumption ZEMAX makes is that the irregularity is half spherical aberration, and half astigmatism. This is less restrictive model than assuming 100% astigmatism, because astigmatism cannot be compensated by focus, and is therefore a more serious defect in the lens.

The min and max values are the irregularity in units of primary wavelength fringes measured at the maximum radial aperture of the surface where the maximum radial aperture is defined by the semi-diameter of the surface.

ZEMAX assumes fringes are measured in a double pass Newton's rings type test. For example, a TIRR of "W" fringes would yield a change in sag of the surface of

$$\Delta Z = \frac{\lambda_p W}{4} (\rho^4 + \rho_y^2),$$

where  $\lambda_p$  is the primary wavelength,  $\rho$  is the normalized radial coordinate, and  $\rho_y$  is the normalized radial coordinate in the y direction. The change in wavefront optical path is related to the change in sag and the index of refraction of the two media the surface separates:

$$\Delta OP = \Delta Z \left( \frac{n_2}{n_1} - 1 \right).$$

The analysis of TIRR uses the Irregular surface type. See the discussion "Tolerancing with the Irregular surface type" later in this chapter.

### **TPAR: Tolerance on parameter data**

TPAR is used to tolerance parameter data. The Int1 number is the surface number. The Int2 number must be an integer between 1 and 8 which indicates the parameter number. See the Chapter "Surface Types" for tables listing which parameter numbers are used for each surface model. TPAR is useful for determining tolerances on aspheric coefficients, and more. Min and max are the changes in whatever units are used by the parameter.

### **TEDV: Tolerance on Extra Data Values**

TEDV is the tolerance for extra data values. Extra data values are used by the extra data editor in ZEMAX-EE to define aspheric terms, diffraction coefficients, or other data. Int1 identifies the surface, Int2 identifies the extra data number, and min and max are the extreme errors in whatever units are used by the extra data value. See the Chapter "Surface Types" for tables listing which extra data numbers are used for each surface model.

### **TIND: Tolerance on Index**

TIND is the tolerance on index of refraction. Int1 identifies the surface, and min and max are the extreme errors in index of refraction. The index error is modeled as an "offset" which is independent of wavelength, unless the glass of the surface is a "model" glass defined by the index at d-light, Abbe number, and dPgF. In this latter case, the TIND is interpreted to be a change in the d-light index, which will change the index at all wavelengths in a nonlinear fashion.

### **TEDX, TEDY: Tolerance on element decenters**

TEDX and TEDY are used to analyze decentration in X and Y respectively. The two surface numbers defined by Int1 and Int2 indicate the beginning and ending surfaces of a lens group. The min and max values are the decentering in lens units.

These tolerances require ZEMAX to insert a coordinate break surface before and after the group, and then decenter the entire group as a unit. For this reason, TEDX and TEDY generally should not be used on either side of an existing coordinate break. If you require this capability, see the discussion on TUDX and TUDY.

TEDX and TEDY may also be used to model decenter of surfaces, like TSDX and TSDY. TEDX and TEDY will work for surfaces of any type, including Standard surfaces and non-Standard surfaces, while TSDX and TSDY only work for Standard surfaces. To decenter a single surface using TEDX or TEDY, simply set Int1 and Int2 to the same surface number.

To check to see if ZEMAX is actually doing what you want, see the SAVE command discussed in the section "Trouble shooting the tolerance results".

### **TETX, TETY: Tolerance on element tilts**

TETX and TETY are used to analyze tilts of either a surface or a lens group about the X and Y axes, respectively. The two surface numbers defined by Int1 and Int2 indicate the beginning and ending surfaces of a lens group. The min and max values are the tipping angles in degrees.



These tolerances require ZEMAX to insert a coordinate break surface before and after the lens group, and to use a dummy surface at the end of the group to return to the front vertex. The entire group may then be pivoted about a point as a unit. For this reason, TETX and TETY should not be used on either side of an existing coordinate break. If you require this capability, see the discussion on TUTX and TUTY.

TETX and TETY may also be used to model tilt of single surfaces, sometimes called "wedge", like TSTX and TSTY. TETX and TETY will work for surfaces of any type, including Standard surfaces and non-Standard surfaces, while TSTX and TSTY only work for Standard surfaces. To tilt a single surface using TETX or TETY, simply set Int1 and Int2 to the same surface number.

TETX and TETY by default pivot about the front vertex of a lens group, however it is often advantageous to pivot about some other point. For example, well-designed lens mounts will pivot about the nodal point of a lens to maintain focus during alignment. This case is easily modeled by ZEMAX through the use of a dummy surface at the nodal point. Simply change the starting surf (Int1) to be at the nodal point dummy surface, and the pivoting will be about that point.

### ***TUDX, TUDY, TUTX, TUTY, TUTZ: User Defined Tilts/Decenters***

These five tolerances, TUDX, TUDY, TUTX, TUTY, and TUTZ, are used for more general user defined tilts and decenters. The names are mnemonics for Tolerance User Decenter/Tilt X, Y, and Z. These are very similar to the TEDX, TEDY, TETX, and TETY tolerances. The difference is that ZEMAX does not automatically insert the required coordinate break surfaces to achieve the specified decenters and tilts in the tolerances. To use the TUDX, TUDY, TUTX, TUTY, and TUTZ commands, you must have already defined all of the surfaces specified by Int1 and Int2 to be coordinate break surfaces. This permits some complex pivoting and decentering about arbitrary points, with a certain finesse required. It is important that the coordinate breaks inserted for use with the TUxx tolerances have nominal values of zero for all tilts and decenters.

Min and max are in lens units for TUDX and TUDY, and in degrees for TUTX, TUTY, and TUTZ.

### **Tolerancing with the Irregular surface type**

The tolerance operands TSDX, TSDY, TSTX, TSTY, and TIRR all use the Irregular surface type to model the perturbation to the lens surface. For a detailed description of the Irregular surface type, see the Chapter "Surface Types".

The Irregular surface is essentially a Standard surface with 7 perturbation terms, that model small amounts of x and y decenter, x and y tilt, and spherical aberration, astigmatism, and coma. The sag of the Irregular surface type is given by:

$$z = \frac{c r^2}{1 + \sqrt{1 - (1 + k)c^2 r^2}} + Z_x \rho_x + Z_y \rho_y + Z_s \rho^4 + Z_a \rho_y^2 + Z_c \rho^2 \rho_y,$$

where

$$\rho_x = \frac{x}{r_{\max}}, \rho_y = \frac{y}{r_{\max}}, \rho = \sqrt{\rho_x^2 + \rho_y^2},$$

and rmax is the maximum radial aperture of the lens, defined by the semi-diameter value for the surface. The coefficients Zx, Zy, Zs, Za, and Zc represent the amount of x tilt, y tilt, spherical aberration, astigmatism, and coma, respectively, in lens units at the maximum radial aperture. When using the Irregular surface, ZEMAX automatically converts from units of fringes to lens units.

The x and y coordinates of the previous equations are in the decentered coordinate system defined by

$$x' = x - dx, y' = y - dy,$$

where the primes have been omitted for clarity.

Note that the Irregular surface does not model tilt exactly. True tilt would involve a rotated coordinate system, with the surface shape remaining a true sphere (or conic) with the center of curvature rotated by the tilt angle. Instead, the Irregular surface simply adds a linear sag to the base spherical (or conic) surface shape. For small tilts, the method of simply adding a linear change in sag is equivalent to an actual tilt. Since the primary use of the Irregular surface is to analyze tolerances, which are small changes, the model is quite acceptable.

The overwhelming advantages of using the Irregular surface type are speed, simplicity, and flexibility. Any Standard surface type can be converted to an Irregular surface type without the need for dummy surfaces or coordinate breaks. Furthermore, the aggregate effects of tilt, decenter, and irregularity may all be modeled simultaneously using the Monte Carlo analysis.

When ZEMAX computes the tolerance analysis using the operands TSDX, TSDY, TSTX, TSTY, or TIRR, the surface is first converted from a Standard to an Irregular surface type. This is why only Standard surface types are supported when using these operands. The tolerance perturbation is then simply added to the surface sag for analysis.

## **Tolerance control operands**

There are also a few tolerance control operands which can be entered in the Tolerance Data Editor. These operands are not tolerances, but are used to define compensators, and to save intermediate results for further evaluation.

TOLERANCE CONTROL OPERANDS

Name	Int1	Int2	Description
COMP	Surface #	Code	Sets a compensator. Code is 0 for thickness, 1 for radius, 2 for conic
CPAR	Surface #	Parameter #	Sets a parameter as a compensator.
SAVE	File Number	-	Saves the file used to evaluate the tolerance on the prior row in the editor. See the discussion below.
STAT	Type	# of standard deviations	If type is 0, 1, or 2, the statistics for the Monte Carlo analysis are normal, uniform, or parabolic, respectively.

The tolerance control operands are described in detail below.

### **COMP: Define compensator**

COMP is used to define the surface number and type of compensator to use for tolerance analysis. Int1 is used to specify the surface, and Int2 is used to define the type. Int2 uses a "code" defined as follows:

Int2 = 0, compensator is thickness

Int2 = 1, compensator is radius of curvature

Int 2 = 2, compensator is conic

The min and max values of the COMP operand indicate the maximum change allowed for the compensator. For example, if the nominal value of a compensator is 50, and the min and max values are -1.0 and 1.0, then the compensator will be restricted to lie between 49.0 and 51.0.

### **CPAR: Define parameter compensator**

CPAR is used to define the surface number and parameter number of the compensator to use for tolerance analysis. Int1 is used to specify the surface, and Int2 is used to define the parameter number. For example, to specify parameter 2 of surface 5 is a compensator, use CPAR with Int1 = 5, Int2 = 2.

See the Chapter "Surface Types" for tables defining which parameters are used by which surface types. The min and max values work the same way as they do for COMP.

### **SAVE: Save sensitivity analysis lenses**

The operation of the tolerance feature is not always transparent, especially for systems that use coordinate breaks or unusual merit functions. To get a closer look at what the tolerance routine is actually doing, use the SAVE operand. SAVE is inserted after any tolerance you would like to inspect in more detail.

For example, suppose you had the tolerance operand TEDX in the Tolerance Data Editor. After reviewing the resulting sensitivities, the merit function changes do not appear to make sense. Edit the tolerance operands in the Tolerance Data Editor to add the SAVE command after the TEDX command. for the Int1 value, enter in 0 or any other digit.

The next time the tolerance analysis is run, ZEMAX will save the file used to compute the TEDX tolerance in a file named TSAV000n.ZMX where n is the integer number specified in the Int1 column.

The lens file will be saved in the same directory as the current lens. ZEMAX will save the lens used to evaluate the preceding tolerance in a ZMX format file. Run the tolerance file in the usual way, and then load the filename you specified. It will show you the prescription data, the coordinate breaks, pickups, variables (which are optimally adjusted compensators), and merit function used to determine the merit function data (usually for the maximum tolerance value). This procedure will let you verify the configuration is in fact what you are trying to model, and will give you insight into how ZEMAX adjusts your lens prescription to determine the tolerances.

### **STAT: Define statistics**

STAT is used to defined the statistics "on the fly" during the Monte Carlo analysis. The STAT command takes two integer arguments. Int1 specifies the statistics type, either 0, 1, or 2 for normal, uniform, or parabolic, respectively. The statistics types are defined in detail in a following section.

The Int2 value is used only by the normal distribution, and it defines "n", the number of standard deviations between the mean and the tolerance extremes. See the discussion of the normal distribution for details.

## **Defining default tolerances**

The default tolerances may be defined by selecting Tools, Default Tolerances from the Tolerance Data Editor menu bar. The Tolerance Data Editor may be activated by selecting Editors, Tolerance Data from the main menu.

The default tolerances dialog box consists of several sections grouped by tolerance type:

### **Surface tolerances**

**Radius Tol:** If this box is checked, then the default radius tolerances will be included. The default tolerances may be specified by a fixed distance in lens units or by fringes of power at the primary wavelength. This tolerance is only placed on surfaces which have optical power, which excludes dummy surfaces which have the same index on both sides. If the surface is plano, then the default tolerance is specified as a variation in fringes, even if another option is selected.

**Thickness Tol:** If checked, a thickness tolerance is also specified on each vertex separation. It is assumed that all variations in thickness affect only that surface and any surfaces in contact with that element; therefore, the first air space after the thickness is used as a compensator. See the detailed discussion on TTHI earlier in this chapter for details.

**Decenter X/Y Tol:** If checked, decenter tolerances are added to each individual lens surface. Tolerances are defined as a fixed decenter amount in lens units. ZEMAX uses TSDX and TSDY for Standard surface decenters, and TEDX and TEDI for non-Standard surfaces.

**Tilt X/Y Tol:** If checked, a tilt or "total indicator runout" tolerance in either lens units or fringes of primary wavelength is added to each lens surface. ZEMAX uses TSTX and TSTY for Standard surface tilts in fringes, TIRX and TIRY for Standard surface total indicator runout (tilt or wedge) in lens units, and TETX and TETI for non-Standard surface tilts in degrees. When ZEMAX uses TETX and TETI, which use units of degrees, the conversion from fringes or lens units to degrees is done automatically.

**Irregularity tolerance:** If checked, an irregularity is specified on each Standard surface type. For details, see the TIRR description given earlier.

**Index tolerance:** TIND is used to model changes in the index of refraction.

### **Element tolerances**

**Decenter X/Y Tol:** If checked, decenter tolerances are added to each lens group. Tolerances may be defined as a fixed decenter amount in lens units.

**Tilt X/Y Tol:** If checked, a tilt tolerance in degrees is added to each lens group and surface. It is important to note that lens groups are by default tipped about the vertex of the first surface in the group. See the section on TETX and TETI earlier in this chapter for information on tipping about some other point.

In addition to the tolerance definitions there are two other options on the dialog box:

**Start At Row:** This control indicates where in the Tolerance Data Editor the default tolerances should be placed. If the row number is greater than 1, then the new default tolerances will be appended starting at the specified row number.

**Use Focus Comp:** If checked, then a default compensator of the back focus (the thickness prior to the image surface) will be defined. Using at least one compensator can greatly relax certain tolerances, however whether or not compensators should be used depends upon the specifics of the design. Other compensators may be defined. See the Section "Defining compensators" for more information.

There are also six buttons:

**OK:** Accept these settings and generate the default tolerances.

**Cancel:** Close the dialog box without changing the default tolerances.

**Save:** Save these settings for future use.

**Load:** Restore the previously saved settings.

**Reset:** Restore the settings to the default values.

**Help:** Invoke the online help system.

By default, the Monte Carlo analysis that ZEMAX performs draws random values from a Gaussian "normal" distribution. Once the default tolerances are defined, they are saved with the lens file automatically. If additional surfaces are inserted in the Lens Data Editor, the tolerance surfaces are renumbered automatically.

## **Defining compensators**

---

Several different types of compensators may be defined; thicknesses (most commonly used), curvature, conic constants, or any parameter value, of any surface or surfaces. The parameters are useful for using tilts and decenters of particular components for compensation. The surface to be tilted must already be defined as a coordinate break (or perhaps a tilted surface), with any appropriate pick ups as required.

The image surface focus is specified as a compensator by default. You may add or delete compensators to customize the analysis to your particular situation, and may use as many compensators as you like. Generally, using more compensators will loosen tolerances, and complicate the actual alignment of the system.

All compensators are defined using the COMP tolerance operand, except parameter tolerances, which use the CPAR operand. The use of COMP and CPAR are described in the section "Tolerance control operands".

## **Performing the tolerance analysis**

---

Once all the tolerance operands and compensators have been defined, the tolerance analysis may be performed. To perform a tolerance analysis, select Tolerancing from the Tools menu in the main program menu bar.

The dialog box which appears has several controls which are described below.

**Fast Tolerance Mode:** Essentially, fast tolerancing considers only the change in paraxial back focus error as a compensator; all other compensators are ignored. Fast mode is very useful for rough tolerancing, and is often as much as 50 times faster than standard mode tolerancing.

**Merit:** The Merit control is used to specify what shall be used as a figure of merit. The options are:

**RMS spot radius:** The best choice for systems which are not close to the diffraction limit; for example, systems with more than one wave of aberration. This is the fastest option. ZEMAX always uses a centroid reference for tolerancing.

**RMS wavefront:** The best choice for systems which are close to the diffraction limit; for example, systems with less than one wave of aberration. This is nearly as fast as RMS spot radius. ZEMAX always uses a centroid reference for tolerancing.

**Merit Function:** Uses whatever merit function has been defined for the lens. This is useful for user-defined tolerancing criteria. User-defined merit functions may also be required for systems with non-symmetric fields, or with significant surface apertures which remove rays.

**MTF:** The best choice for systems which require a MTF specification. ZEMAX uses an approximation for the MTF that is very fast to compute and accurate enough for tolerancing purposes. MTF is the slowest of the default criteria. The MTF is the average of the sagittal and tangential response. The frequency at which the MTF is computed is specified in the "MTF Frequency" control.

**MTF Frequency:** If MTF is selected as the Merit, then this control is active and is used for defining the MTF frequency. MTF frequency is always measured in line pairs per mm in the image space.

**Mode:** Mode is either Sensitivity or Inverse. Sensitivity mode computes the change in the merit for each of the extreme values of the tolerances. Inverse computes the value of each tolerance that will result in the decrease in performance specified by Max Criteria. Inverse mode will change the min and max values of the tolerance operands. See the description for "Max Criteria" and "Inverse Sensitivity" elsewhere in this chapter for more information.

**Max Criteria:** When using Inverse mode, this control is active and is used to define the limit of performance acceptable for computing inverse tolerances. For example, suppose the Merit is RMS Spot Size, and the nominal RMS of a system is .035 mm. If Max Criteria is set to 0.050, then ZEMAX will compute the min and max value of each tolerance that degrades the performance to an RMS spot size of 0.050. The Max Criteria value must represent worse performance than the nominal system has. When using MTF as a merit, then Max Criteria is the lower bound on MTF since lower numbers indicate worse performance. See the description for "Mode" and "Inverse Sensitivity" elsewhere in this chapter for more information.

**Fields:** Generally speaking, the field definitions used for optimization and analysis are inadequate for tolerancing. For example, a rotationally symmetric lens may use field definitions of 0, 7, and 10 degrees. For tolerancing purposes, the lack of symmetry in the field definitions may cause inaccurate results when analyzing tilt or decenter tolerances. When constructing a merit function to use for tolerancing ZEMAX can use three different field settings:

**Y-Symmetric:** ZEMAX computes the maximum field coordinate, then defines new field points at +1.0, +0.7, 0.0, -0.7, and -1.0 times the maximum field coordinate, in the Y direction only. All X field values are set to zero. This is the default for rotationally symmetric lenses.

**XY-Symmetric:** Similar to Y-Symmetric, except there are 9 field points used. The 5 Y-Symmetric points are used, and -1.0, -0.7, +0.7, and +1.0 are added in the x direction only.

**User Defined:** Use whatever field definitions exist in the current lens file. This option is required when using vignetting factors, and is highly recommended when tolerancing non rotationally symmetric lenses, or lenses with complex field weighting.

If user defined fields are used, no adjustment of the weights is performed. For the Y-Symmetric case, the center point has a weight of 2.0, all others have a weight of 1.0. For the XY-Symmetric case, the center point has a weight of 4.0, all others unity.

**Sampling:** Sampling is used to set how many rays are traced when computing the tolerance merit function. Higher sampling traces more rays, and gives more accurate results. However, the execution time increases. Usually, a sampling of 3 is sufficient for quality optical systems. Systems with high amounts of aberration require higher sampling than systems with low aberration.

The most reliable method for determining the best sampling setting is to run the tolerance at a sampling of 3, then again for a sampling of 4. If the results change moderately, then use the higher setting. If they change substantially, check the next higher sampling setting. If the results change little, go back to the lower sampling. Setting the sampling higher than required simply increases computation time without increasing accuracy of the results.

**Perform Sensitivity:** If this box is checked, then the sensitivity (or inverse sensitivity) analysis will be performed. If this box is not checked, it bypasses the sensitivity portion and proceeds directly to the Monte Carlo simulation.

**# Opt Cycles:** This determines how rigorously ZEMAX will attempt to optimize compensator values. If set to Auto, then ZEMAX will use a default number of optimization cycles. Generally, the auto setting should be used. For rough tolerancing, a low number, such as 1 or 2, may be used. If the compensators are difficult to optimize, a higher setting may increase accuracy. If too few optimization cycles are selected, then the tolerances will be

pessimistic; the predicted performance will be worse than the actual performance. Higher settings increase accuracy at the expense of run time.

Show Descriptions: If checked, a full description of the meaning of each tolerance operator will be provided in the analysis report. If unchecked, only the tolerance operator abbreviations will be listed.

Show Compensators: By default, the compensator values are not printed out during the sensitivity analysis. If this box is checked, then each compensator value will be printed along with the change in the merit function for each tolerance.

Monte Carlo: This control is used to specify how many Monte Carlo simulations should be performed. The default setting of 20 will generate 20 random lenses which meet the tolerances specified. See the section "Monte Carlo simulations" for more details. The number of Monte Carlo runs may be set to zero, which will omit the Monte Carlo analysis from the summary report.

Statistics: Choose either a Gaussian "normal" distribution, "uniform", or "parabolic" distribution. This setting is only used by the Monte Carlo analysis; see that discussion for details on the statistics mode as well as the "STAT" command which provides detailed control over the statistics model used.

Save MC Runs: This option is used to save a specific number of lens files generated during the Monte Carlo analysis. The value specifies the maximum number of lens files to be saved. For example, suppose 20 is selected. After the first Monte Carlo lens is generated, the lens file will be saved in the file MC\_T0001.ZMX. The second Monte Carlo lens file will be generated, then saved in MC\_T0002.ZMX, and so on. Only the first 20 Monte Carlo lenses will be saved (the last will be MC\_T0020.ZMX). If fewer than 20 Monte Carlo runs were requested, then fewer than 20 lenses would be saved. Be sure you have no lens files with the names MC\_Txxxx.ZMX, as ZEMAX will overwrite these files without warning as the lenses are saved. The purpose of this feature is to allow further study on the lenses being generated by the Monte Carlo feature.

Configuration #: For multi-configuration lenses, indicates which configuration should be used for tolerancing. The selected configuration only will be considered, and the configuration number will be printed on the final report.

Hide All But Worst: If checked, printing of all the sensitivity data will be turned off. This is useful for decreasing the size of the output report. The "hide" check box is normally used in conjunction with the "Show worst" control. The "Show worst" control can be set to sort and display only some number of tolerance operands; this permits limiting the printout to the most severe tolerances only.

Show worst: See "Hide all but worst" above.

List Fields Individually: When selected, ZEMAX will list for the Monte Carlo generated lenses the nominal and predicted performance for each field point separately. This allows detailed inspection of the lens performance as a function of field position. The criterion, best, worst, mean, and standard deviation are listed for every defined field point.

Force Ray Aiming On: If the lens being toleranced is already using ray aiming, then ray aiming will be used when evaluating the tolerances. If ray aiming is not already on, then ray aiming will only be used if this box is checked. Generally, using ray aiming yields more accurate results but slower computation speed. For preliminary or rough tolerance work, leave the switch at the default "off", but for final or precise work, set the switch "on".

Status: This control is used by the tolerance algorithm to provide status messages during computation of the tolerance analysis.

There are also six buttons on the bottom of the dialog box:

OK: Performs the tolerance analysis using the current options.

Cancel: Exits the dialog box without performing the tolerance analysis.

Terminate: Terminates the tolerance analysis.

Save: Save the currently selected options for future use.

Load: Restore the previously saved settings

Reset: Restore the settings to the default values.

Once all of the options have been selected, press OK to begin the tolerance analysis. Details on the method of calculation are provided in the next section.

## **How ZEMAX computes the tolerance analysis**

---

ZEMAX begins the tolerance analysis by saving the lens in a temporary file which will be used to restore the lens after the tolerancing is complete. All of the changes made during tolerancing will ultimately be discarded, and the original lens file will be restored unmodified.

ZEMAX removes all of the variables, and all solves except pickup solves. Pickup solves may cause problems during tolerancing; see the section "Tolerancing with solves" later in this chapter for more information.

The tolerance operators are read, and the compensating parameters defined by COMP and CPAR specified are set as variables.

If ray aiming is on in the lens being toleranced, or if the "Force Ray Aiming On" switch is checked, then ray aiming will be used in evaluating the tolerances, otherwise, ray aiming is left off. Tolerances computed using ray aiming are more accurate, but execution speed is slower. See the Chapter "System Menu" for details on ray aiming.

ZEMAX then uses the Merit, Field, MTF Frequency, and Sampling settings on the tolerance dialog box to define an appropriate merit function for the tolerancing. Since this is done to the temporary file only, the original merit function defined for the lens is not disturbed.

Boundary constraints are added to the merit function to limit the compensators to the min and max boundaries specified using the COMP and CPAR commands.

ZEMAX then calls the optimization function to find the best values for the defined compensators. The resulting lens file is then saved for subsequent use by the tolerancing algorithm. If fast mode is on (see below) then the back focus is simply adjusted instead of optimizing the lens.

The merit function of this lens is considered to be the "nominal" merit function. Note that the nominal merit function value generally will not be the same as the merit function value that is reported on the optimization or merit function editor windows, because ZEMAX constructs a new merit function just for use in tolerancing.

## **Fast Tolerancing Mode**

---

ZEMAX has two methods for evaluating tolerances: fast and standard. If "Fast Tolerance" mode is selected, then several assumptions are made which greatly speeds up the tolerance evaluation. First, all defined compensators and compensator boundary constraints are ignored. Back focus is implemented as a compensator using a simple focus error solve. This means the focus is adjusted to maintain the amount of defocus in the nominal lens design, without attempting to reoptimize the back focus precisely.

These assumptions greatly speed up the tolerance evaluation, resulting in an approximately 50X speed increase in the analysis. Fast mode is highly accurate if back focus is the only compensator needed; typically fast mode results are within a few percent of the much slower standard mode.

If fast mode is not on, then ZEMAX uses the optimization algorithm to find the best value for all the compensators.

Fast Tolerance mode is extremely fast, accurate, and should be used if back focus is the only compensator, and the system is well described by paraxial rays, such as systems with rotational symmetry. Fast tolerance should not be used when the system is highly non-symmetric or there are multiple compensators. When in doubt, run both modes and compare the results.

The sensitivity analysis, inverse sensitivity, and Monte Carlo algorithms all work with fast mode on or off. Since fast mode does not precisely optimize the compensators, fast mode results are generally slightly more pessimistic than standard mode results.

### ***Sensitivity analysis***

---

For the sensitivity analysis, each tolerance is evaluated independently using the following algorithm:

The temporary lens is restored.

The parameter whose tolerance is being evaluated is adjusted to the extreme minimum value. for example, if the tolerance being evaluated is TRAD, and the nominal value is 100 mm, with a minimum tolerance of -0.1

mm, the radius is set to 99.9. If the tolerance is an element tilt or decenter tolerance, dummy coordinate breaks are inserted as required to model the perturbation. For surface tilts and decenters, such as TSDX, TSDY, TSTX, TSTY, TIRX, or TIRY, the Irregular surface type is used if the surface is initially of type Standard.

The compensators are adjusted. The method used depends upon whether or not fast mode is on, see the earlier section for details.

The resulting merit function is printed on the report.

The procedure is repeated for the maximum tolerance.

This basic algorithm is repeated for each tolerance operand.

The value of the sensitivity analysis is that the tolerances which are too loose will, in general, have greater contributions to the increase in the merit function than other tolerances. This technique allows the designer to identify surfaces which are highly sensitive to certain errors, such as tilts or decenters. Different surfaces also will, in general, have very different sensitivities to the various errors. The sensitivity analysis aids in identification of which tolerances need to be tightened, and which might be loosened. It is also valuable for finding the optimum (and minimum) number of compensators, and the required range of adjustment. There are in fact many more applications for this feature; for example, designing lens mounts to maximize compensation leverage.

The amount of output can be overwhelming, especially for lenses with many elements and a corresponding large number of tolerances. Often, the sensitivity to the tolerances varies widely over all the possible tolerances. The "Show worst" control is extremely useful for summarizing the worst offenders, because it sorts the tolerances by contribution to the merit function and then prints them in descending order. The "Hide all but worst" control turns off the bulk of the printing if only the worst offenders are of interest.

After all the individual tolerances are computed, ZEMAX then computes a variety of statistics, the most important of which is the estimated change in the merit function criteria and associated estimated performance. ZEMAX uses a Root Sum Square (RSS) assumption for computing the estimated changes in the performance. For each tolerance, the change in performance from the nominal is squared and then averaged between the min and max tolerance values. The resulting averaged squared values are then summed for all the tolerances, and the square root of the result is taken. The average of the min and max tolerances is taken because the min and the max tolerance cannot both occur simultaneously, and so summing the squares would result in an overly pessimistic prediction. The resulting RSS is the estimated change in performance.

### **Inverse sensitivity**

If an inverse sensitivity is being performed, the tolerances are computed the same way as for the sensitivity analysis. However, the calculation is performed iteratively inside a loop while adjustments are made to the min and max tolerances. The adjustments are made until the resulting merit function is approximately equal to the Max Criteria value defined on the tolerance dialog box.

For example, if the Merit is RMS spot size, the nominal merit is 0.035, and the Max Criteria is 0.050, ZEMAX will adjust the min and max tolerances until the merit is 0.050 at the two extremes. There are exceptions to this rule:

The Max Criteria must represent worse performance than the nominal. An error message will be issued and the tolerance analysis will not be performed if the nominal performance is worse than the Max Criteria.

If the min or the max tolerance adjustment results in better performance than the nominal system, then the tolerance setting is not modified, and the resulting better merit is reported. This can happen when a system improves due to an increase in  $F/\#$  when tolerancing radii.

If the starting value of the tolerance yields performance better than the Max Criteria, the tolerance is not adjusted. This means the tolerance will never be loosened; it can only be tightened during inverse sensitivity analysis. For example, if the nominal is 0.035, and the Max Criteria is 0.050, and the initial tolerance yields a performance of 0.040, the tolerance will not increase. To compute the actual limit, the tolerance must first be loosened on the Tolerance Data Editor, and then inverse sensitivity repeated. This is done to prevent looser than necessary tolerances. Generally, tolerances that are looser than some reasonable value do not decrease manufacturing costs.

The estimated change in performance is computed in the same way as for the sensitivity analysis, using the newly adjusted tolerances.



Note that the inverse sensitivity maximum criteria limit is computed for each tolerance individually. The overall degradation in performance estimate will still be given by the RSS of all the individual increases.

### **Monte Carlo**

Unlike the sensitivity and inverse sensitivity analysis, the Monte Carlo analysis attempts to simulate the effect of all the perturbations at one time. For each Monte Carlo cycle, all of the parameters which have specified tolerances are randomly set using one of three statistical distributions. The default distribution is a modified Gaussian "normal" distribution of the form

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(\frac{-x^2}{2\sigma^2}\right), -n\sigma \leq x \leq n\sigma.$$

The modification is that the randomly selected value  $x$  (measured as an offset from the midpoint between the two extreme tolerances) is restricted to lie within "n" standard deviations of zero. The default value of "n" is two, however "n" may be changed using the STAT command defined earlier. This is done to insure that no selected value will exceed the specified tolerances. The standard deviation is set to one over "n" times half the maximum range of the tolerances. For example, if "n" is 2, and a thickness is nominally 100 mm with a tolerance of plus 3 or minus 1 mm, then the selected value will be drawn from a normal distribution with a 102 mm mean and a range of plus or minus 1 mm, with a standard deviation of 0.5. If "n" is 5, then the standard deviation will be 0.2. The larger "n" is, the more likely the selected value will be closer to the mean value of the tolerance extremes.

The uniform distribution is of the form

$$p(x) = \frac{1}{2\Delta}, -\Delta \leq x \leq \Delta.$$

The  $\Delta$  value is one half of the difference between the max and min tolerance values. Note that the randomly selected value will lie somewhere between the specified extreme tolerances with uniform probability.

The parabolic distribution is of the form

$$p(x) = \left(\frac{3}{2\Delta^3}\right)x^2,$$

where  $\Delta$  is defined exactly the same as for the uniform distribution. The parabolic distribution yields selected values that are more likely to be at the extreme ends of the tolerance range, rather than near the middle, as for the normal distribution.

Note that going from normal to uniform to parabolic statistics yields successively more conservative tolerances.

For each cycle, the compensators are adjusted and then the merit function and compensator values are printed. After all of the Monte Carlo trials a statistical summary is provided.

The value of the Monte Carlo analysis is estimating the performance of the lens considering all of the tolerances simultaneously. Unlike the sensitivity analysis, which identifies the "worst offenders" in the system, the Monte Carlo analysis estimates the real-world performance of a system which meets the tolerances specified. The statistical summary provided may be highly useful for lens systems which are mass-produced. Lenses which are one of a kind, of course do not follow these statistics because of the inadequate sampling. However, the Monte Carlo analysis is still useful because it indicates the probability that a single lens will meet the required specification.

### **Tolerancing multi-configuration (zoom) lenses**

Tolerance analysis can be performed on each configuration of a multiple configuration lens. Simply select the desired configuration from the "Configuration #" drop down box. Generally, it is not required to tolerance all configurations, but that is certainly an option. When using inverse tolerances, successive tolerance analysis in each configuration will yield the tightest tolerances which apply to all configurations collectively.

## Tolerancing with solves

---

Generally speaking, you should delete all solves and variables before tolerancing a lens. It may be more convenient to save the lens in a temporary file for tolerancing if you have many solves or variables to delete that you may need later. Although the tolerancing will still work if you leave solves in your lens file, the results may not be what you expect. The reason is that certain solves, such as paraxial marginal ray height solves, no longer have meaning when the optical system contains tilted or decentered elements. Even if the system is initially rotationally symmetric, most tolerance commands such as TETX and TETY force the system to be non-rotationally symmetric. Pickup solves also can cause strange results because the tolerancing algorithm sometimes will manipulate the surface description data (such as thicknesses) when inserting and deleting coordinate breaks.

There are however occasions where a pickup solve is really what you want. For example, if you are using a lens in double-pass, and a tilt on one element implies a tilt on a subsequent element, then a pickup solve can be used to pickup the tilts from the first element. For this special case, you will need to use the TUTX, and TUTY commands, and the default tolerances will not work. If you have these types of solves in the system, ignore the warning "Solves should be deleted before tolerancing".

## Trouble shooting the tolerance results

---

See the discussion in the section on the SAVE operand.

If any of the calculated tolerance data such as merit functions and percent changes show a value of "Infinity", this means the merit function could not be evaluated for the specified tolerance. Usually the merit function cannot be evaluated because of total internal reflection of some of the ray targets. The statistical data which follows the sensitivity analysis is usually meaningless if any of the tolerances have a merit function value of infinity. One solution is to decrease by a factor of two or more the tolerance value, and then repeat the analysis.

## Pitfalls when tolerancing

---

One possible error is non-physical propagation of rays when using tilt tolerances such as TETX and TETY. If two elements are separated by a very small air space or dummy surface, the default tolerances will include tilting of each element independently. If the element spacing is small, it is possible for one of the elements to be tipped to the point where it "collides" with the other element. This could not happen in practice, however, for small amounts of tilt the tolerance values are still a good indication of performance.

Because the tolerance routine tilts and decenters lenses, the algorithm automatically turns ray aiming "on" (see the Chapter "System Menu" for details about ray aiming). If the nominal merit function the tolerance algorithm reports is different from the expected value, check to see if ray aiming is "off" in the lens file. Try using ray aiming "on" and re-optimizing. Generally, if there is a big difference between merit functions when ray aiming is turned on and off, then it should be left on.

## Summary

---

The tolerance routine is quite flexible and powerful. ZEMAX uses no approximations, extrapolations, or estimations to compute the tolerances. For this reason, it gives useful results for both conventional and complex systems. It is very important to appreciate that tolerancing is a complex procedure, and the algorithms used by ZEMAX to manipulate the lens data are not infallible. Therefore, it is the designers responsibility to verify that the program is computing reasonable results.

This page intentionally left blank.

# MULTI-CONFIGURATIONS

## Introduction

ZEMAX supports a very general capability for defining, analyzing, and optimizing optical systems which are used in multiple configurations. Multi-configurations are used to design zoom lenses, or to optimize lenses tested and used at different wavelengths, or optical systems used in various configurations, to name a few. Like other ZEMAX features, multi-configuration is well integrated. However, like tolerancing, it requires a little more care and practice to become proficient.

ZEMAX uses a simple to learn substitution procedure for defining multiple configurations. The configurations are differentiated by different values for the same parameter. For example, in a zoom lens, the spacings between various elements may take on more than one value. Each set of values used together forms one configuration.

## The first step

By far the most important step is to define one configuration using the normal ZEMAX mode first. It is a good idea to start with the most complex configuration first. If all of the configurations have the same number of elements, pick any one of them. Once you have the basic configuration defined, it is time to define new configurations which are variations of the first. The first configuration does not need to be optimized yet, you can optimize across configurations later.

Select Editors, Multi-configuration from the main menu bar. The spreadsheet which appears is the multi-configuration editor (MCE). Using the menu bar on the MCE, both configurations (columns) and operands (rows) can be inserted and deleted. The insert and delete keys will also add or remove new operand rows. The data entered on the MCE will be saved automatically whenever you save your lens file.

To change the operand type, double click on the type column. A dialog box will appear where the type and number of the multi-configuration operand can be changed. The operands are also summarized in the following table.

SUMMARY OF MULTI-CONFIGURATION OPERANDS

Type	Number	Description
CRVT	Surface #	Curvature of surface.
THIC	Surface #	Thickness of surface.
GLSS	Surface #	Glass.
CONN	Surface #	Conic constant.
PAR1	Surface #	Parameter 1.
PAR2	Surface #	Parameter 2.
PAR3	Surface #	Parameter 3.
PAR4	Surface #	Parameter 4.
PAR5	Surface #	Parameter 5.
PAR6	Surface #	Parameter 6.
PAR7	Surface #	Parameter 7.
PAR8	Surface #	Parameter 8.
XFIE	Field #	X-field value.
YFIE	Field #	Y-field value.
FLWT	Field #	Field weight.
FVDX	Field #	Vignetting factor VDX.
FVDY	Field #	Vignetting factor VDY.
FVCX	Field #	Vignetting factor VCX.
FVCY	Field #	Vignetting factor VCY.
WAVE	Wave #	Wavelength.
WLWT	Wave #	Wavelength weight.

Type	Number	Description
PRWV	Ignored	Primary wavelength number.
APER	Ignored	System aperture value (whatever the current aperture definition is, such as entrance pupil diameter or F/#).
STPS	Ignored	Stop surface number. The stop can be moved to any valid surface number (excluding the object and image surfaces) by specifying an integer argument for each configuration.
SDIA	Surface #	Semi-diameter.
CSP1	Surface #	Curvature solve parameter 1.
CSP2	Surface #	Curvature solve parameter 2.
TSP1	Surface #	Thickness solve parameter 1.
TSP2	Surface #	Thickness solve parameter 2.
HOLD	Ignored	Holds data in the multi-configuration buffer, but has no other effect. Useful for temporarily turning off one operand without losing the associated data.
APMN	Surface #	Surface aperture minimum value. The surface must have a defined aperture (NOT semi-diameter).
APMX	Surface #	Surface aperture maximum value. The surface must have a defined aperture (NOT semi-diameter).
APDX	Surface #	Surface aperture X- decenter. The surface must have a defined aperture (NOT semi-diameter).
APDY	Surface #	Surface aperture Y- decenter. The surface must have a defined aperture (NOT semi-diameter).
TEMP	Ignored	Temperature in degrees Celsius.
PRES	Ignored	Air pressure in atmospheres. Zero means vacuum, 1 means normal air pressure.
EDVA	Surface/Extra Data Number	The EDVA operand is used to assign multiple values to the extra data values. This operand requires 2 numerical arguments: the surface number and the extra data value number.
PSP1	Surface #	Parameter solve parameter 1 (the pickup surface). This operand requires 2 numerical arguments: the surface number and the parameter number.
PSP2	Surface #	Parameter solve parameter 2 (the scale factor). This operand requires 2 numerical arguments: the surface number and the parameter number.
MIND	Surface #	Model glass index.
MABB	Surface #	Model glass Abbe.
MDPG	Surface #	Model glass dPgF.
CWGT	Ignored	The overall weight for the configuration. This number only has meaning relative to the weights in other configurations. The configuration weight is only used by the default merit function algorithm to set up a merit function which favors configurations via relative weighting. If the configuration weight is zero, then the configuration is ignored during construction of the default merit function.
FTYP	Ignored	Field type. Use 0 for angle in degrees, 1 for object height, 2 for image height.
RAIM	Ignored	Ray aiming. Use 0 for none, 1 for paraxial reference, and 2 for real reference.
COTN	Surface #	The name of the coating, if any, to be applied to the surface.

## Defining the number of configurations

The number of configurations (also called the number of zoom positions) is changed by inserting or deleting configurations using the Edit menu option.

## Defining each configuration

---

To define a multi-configuration operand, double click on the name of the operand you wish to change (the names are listed in the left most column). For example, suppose you want to enter in a multiple value for the thickness of surface 5. Assume you want three different configurations. Insert 2 new configurations using the Edit menu. Double click on the left most column in row 1. Select "THIC" from the drop-down list on the dialog box for the Operand Type. Select "5" for the surface number, then click on OK. Now in each configuration column, enter the desired thickness in that configuration.

## Adding and removing elements

---

One application for the multi-configuration feature is to design systems that can be operated with or without certain elements being present. This can easily be handled by making "phantom" elements in some of the configurations. The trick is to define the element (really the first surface of the element) to have a glass type which changes with the configuration. Suppose in configuration 1 the element is present and is made of BK7. In configuration 2, the element is absent. Using the GLSS operand, enter in "BK7" for configuration 1 and leave the field for configuration 2 blank. Although the surfaces which define the element are still present in the lens prescription, they have no effect because the glass has been removed (the element becomes a pair of dummy surfaces). Note that the number of surface must remain constant in all configurations.

## Changing configurations

---

To perform any analysis, proceed as usual. The program will use the current configuration for all calculations and graphics. To change configurations, double click on the column header at the top of the configuration you wish to change to on the MCE. The shortcut keys Ctrl-A and Shift-Ctrl-A also work.

## Optimization with multi configurations

---

ZEMAX will optimize multi-configuration data as readily as the conventional spreadsheet data. To make a multi-configuration parameter variable, place the cursor on the parameter, and press Ctrl-V. This is a toggle; pressing Ctrl-V again would eliminate the variability. When optimization is invoked, the optimization will automatically count the new variable. As many multi configuration variables as you like may be defined, up to the maximum number of variables supported by the program.

To optimize across configurations, simply select "Default Merit Function" from the Tools menu of the merit function editor. ZEMAX will build an appropriate merit function for you. Optimization across configurations is performed using the CONF operand. This special operand changes the current configuration during the evaluation of the merit function. This means all operands defined after CONF will be relevant to the new configuration. CONF may be used multiple times in the operand sequence to evaluate various parameters.

Note that if you change the field angles, heights, or weights, or the wavelength values or weights on the multi-configuration screen you should rebuild the default merit function. When the default merit function is constructed, it uses the data for each configuration to determine the rays traced, and the appropriate weighting.

Note that if you change the field angles, heights, or weights, or the wavelength values or weights on the multi-configuration screen you should rebuild the default merit function.

## Using solves with multi-configuration data

---

Only one solve is supported on the multi-configuration screen: a pick-up from another configuration. Suppose there are three configurations, and in two of the configurations the values for one of the rows (say a THIC or GLSS) need to be the same value. A pick-up solve on one of the configurations for that operand can be used to insure they are always the same.

To set a pickup solve, double click on the row and configuration on which the solve is to be placed. A dialog box will appear which lists all solves for that parameter. The solves include "Fixed", "Variable", and "Pick Up n". Neither "Fixed" or "Variable" are really solves, but they are displayed for completeness. If "Pick-Up n" is displayed, then the value of that cell will be picked up from configuration n.

See the chapter "Tutorial" for a tutorial on using multi-configuration.

ZEMAX-EE also supports a "TCE Pick Up n" solve which incorporates thermal effects; see the chapter "Thermal Analysis" for details.

This page intentionally left blank.



---

# USING GLASS CATALOGS

## Introduction

---

There are several standard catalogs supplied with ZEMAX, and you may create your own custom catalogs. The standard catalogs may be edited to suit your requirements. This chapter describes how to add glasses to the existing or new catalogs, and how to use the catalogs in your optical designs. There is no limit to the number of glass catalogs you may create.

ZEMAX computes indices of refraction from coefficients entered into the glass catalogs. When you specify a glass name such as "BK7" in the glass column of the LDE, ZEMAX looks for the name in each of the currently loaded glass catalogs. If the glass is found, ZEMAX uses the coefficients for that glass, and then using the formula for the glass selected in the catalog, computes the indices at each of the defined wavelengths.

This method may seem more complex than simply entering in the indices of refraction, but the advantages are numerous. For one, the formulas are generally more accurate than user-entered data. Catalog data is more convenient as well, requiring the user to only supply the glass name. This is a particular advantage during the glass selection phase of the design. Additionally, any wavelength may be used, even if no explicit index data at that wavelength is available. The primary disadvantage is that the coefficients must be calculated, although this data is either readily available in catalogs or easily calculated. If you have the index data for some material not in the catalog, or if you feel your data is better than the catalog data, ZEMAX will compute the coefficients for you automatically; see the section "Fitting index data".

## Specifying which glass catalogs to use

---

It is important to note that this chapter describes the loading, editing, and managing of the glass catalogs. However, to specify that a particular catalog be used for a particular lens, use the General Lens Data dialog box (select System, General). On this screen is a data field that reads "Glass Catalogs:". The default entry is "schott misc" which refers to the SCHOTT.AGF and MISC.AGF glass catalogs. You may specify an alternative, or additional catalogs on this line. To specify an alternative catalog, simply enter the name of the catalog (with no extension). You may also specify multiple simultaneous catalogs by simply listing them with spaces between the names. For example, to use the "schott" and the "hoya" catalogs, enter "schott hoyo" in the "Glass Catalogs" data field. To use the "ohara", "schott", and "infrared" catalogs, enter "ohara schott infrared" in the data field. It may be easier to simply click on the glass catalog names listed in the group of checkboxes below the list of names. You only need to type in the name of the catalog if it is not already listed in one of the available checkboxes.

All the listed catalogs must be in the "ZEMAX Path" listed on the environments screen (select System, Environment). The reason that glass catalog names are specified on this screen is that the catalog choice is stored with each lens separately. If you now save the lens, and reload it at a later time, the correct glass catalogs, and only those catalogs, will automatically be loaded. You can specify a different combination of catalogs for every lens you design, if you like.

## Loading a glass catalog

---

To load an existing glass catalog for editing, select Tools, Glass Catalogs. Select the catalog name from those listed in the drop-down list on the dialog box. Once the catalog is loaded, you may insert, delete, or modify data in the catalog, as is described in the following sections. You can save the newly modified catalog, but only back to a catalog of the same name. Therefore, be very careful when editing the supplied catalogs; you may inadvertently alter the data, which would then yield erroneous ray trace results if the catalog data is used later.

## Description of catalog data

---

The glass catalog dialog box displays a great deal of data about each individual glass. The data fields are described in the following table.

Item	Description
Catalog	Used to specify which of the .AGF format catalogs to display.
Glass	Used to specify which glass within the catalog to display data for.

Item	Description
New Name	In the event that a glass needs to be given a different name, the name of the currently selected glass can be modified in this field.
Formula	The dispersion of each glass is described by a formula. This control allows selection of which formula is used. If this setting is changed, then the dispersion data becomes invalid unless the appropriate coefficients are also entered. See the section "The glass dispersion formulas".
Index Nd	The index at d-light, or .587 microns. ZEMAX does not use this number when computing the index of refraction. It is displayed solely for reference. The entry may be meaningless for some materials which do not transmit well in the region around .587 microns.
Abbe Vd	The Abbe value at d-light. This number is not used by ZEMAX when computing the index of refraction. It is displayed solely for reference.
Exclude Substitution	If checked, then this glass will not be selected during global optimization, conversion from model to real glasses, or be considered by the RGLA optimization operand.
K1, L1... A0, A1..., A, B, C, etc.	The first six rows in the center column of the dialog box display the dispersion coefficient data. The names of these coefficients changes depending upon the glass formula.
Alpha 1, Alpha 2	Alpha 1 is the coefficient of expansion in the temperature range from -30 to +70 degrees celsius. Alpha 2 is the coefficient of expansion in the range of 20 to 300 degrees celsius. Both numbers are dimensionless. The values displayed or entered need to be multiplied by 1e-6 to yield the actual values. These values are also called the thermal coefficient of expansion, or TCE.
D0, D1, D2, E0, E1, Ltk	These are the thermal coefficients used by the thermal analysis model. See the chapter "Thermal Analysis" for details.
p	p is the material density in grams per cubic centimeter.
dPgF	The value is the deviation of the relative partial dispersion from the normal line.
Min Wave, Max Wave	The minimum and maximum wavelengths in microns over which the dispersion formula returns valid index data.

## Creating a new catalog

To create a new glass catalog, first exit ZEMAX. Using the DOS COPY command, copy one of the catalogs (for example, SCHOTT.AGF) to a new file name. You can then run ZEMAX, load the new catalog, delete the data you do not want, and add the data you do want.

## The glass dispersion formulas

The coefficients in the catalog are used in any one of several polynomial formulas that ZEMAX recognizes. Currently, there are five glass formulas for the behavior of the dispersion: the Schott constants of dispersion, the Sellmeier 1, the Sellmeier 2, the Herzberger, and the Conrady. The reason there are two Sellmeier formula is due to the variations on the form of the equation common in the literature on measured index values. There is also a dispersion formula described by the six-digit MIL number, but those indices are calculated directly from the MIL number entered on the spreadsheet. The MIL number formula is not part of the glass catalog, and so it will not appear. See the last section in this chapter for a discussion of the MIL number glass formula.

You may add new glasses to the currently loaded catalog if you have index data given in the form of one of the following equations. In all of the equations  $\lambda$  is in microns.

The Schott constants of dispersion formula is

$$n^2 = a_0 + a_1 \lambda^2 + a_2 \lambda^{-2} + a_3 \lambda^{-4} + a_4 \lambda^{-6} + a_5 \lambda^{-8} .$$

The required coefficients are available in most manufacturers glass catalogs. Schott no longer uses this formula, but it is widely used by other glass manufacturers.

The Sellmeier 1 formula is

$$n^2 - 1 = \frac{K_1 \lambda^2}{\lambda^2 - L_1} + \frac{K_2 \lambda^2}{\lambda^2 - L_2} + \frac{K_3 \lambda^2}{\lambda^2 - L_3}.$$

Coefficients for all three terms may be entered to describe the material, although fewer terms may be used. There is a variation of this expression common in the literature; the formula is

$$n^2 - 1 = A + \frac{B_1 \lambda^2}{\lambda^2 - \lambda_1^2} + \frac{B_2}{\lambda^2 - \lambda_2^2}.$$

The difference is that only two terms are used, there is no wavelength dependence in the numerator of the second term, and there is a constant term. This alternate expression is called the Sellmeier 2. The Herzberger expression is

$$n = A + BL + CL^2 + D\lambda^2 + E\lambda^4 + F\lambda^6,$$

$$L = \frac{1}{\lambda^2 - .028}.$$

The Herzberger is used mainly in the infrared spectrum.

These coefficients for infrared materials are often measured experimentally with considerable variance in the results; use caution in the interpretation of exact index values. The reason ZEMAX does not have a more extensive infrared catalog is because of the great variability of the data. If you have better data for a given material or sample, by all means use it. Often melt data for a specific piece of glass is available in terms of coefficients; this data may also be useful for some applications.

The Conrady formula is

$$n = n_0 + \frac{A}{\lambda} + \frac{B}{\lambda^{3.5}}.$$

The Conrady formula is extremely useful for fitting to sparse data. For example, if you have only three index-wavelength pairs of data, fitting to the six-term Schott formula would yield meaningless data at the intermediate wavelengths.

It is important to note that some publications use equations similar, but not identical to any of these expressions. It is often possible to rearrange the expressions to get them into the required form, and then recompute the required coefficients.

It is also a good idea to check the coefficients against a tabular listing of the index data available in many handbooks and publications. Use the dispersion plot or listing feature, or the prescription data report, which lists index data at each surface. If there are discrepancies, check the data you have entered carefully, and verify that the correct units and formula are being used.

## Fitting index data

---

It is often the case that the materials you are designing with are in the catalog already. If they are not, you can enter the coefficients for the formulas described previously. As an alternative, ZEMAX will compute either the Schott, Herzberger, or Conrady constants of dispersion formula coefficients for you. ZEMAX does not fit data to the Sellmeier formulas because they have non-linear coefficients. With the glass catalog dialog box displayed, click on Fit Data, and the "Fit Index Data" dialog box appears.

On the left side of this screen is a two-column spreadsheet editor. Using the mouse, enter in the wavelength (in microns) and index data you have. The more data you enter, the more accurate will be the fit.

If you have more data than will fit in the spreadsheet, use the data most closely representing the wavelength region of interest. At least three points are required to get a good fit if you are using the Conrady formula, six or more (and preferably twelve to fifteen) for the Schott or Herzberger formula. Select the formula to use by selecting the formula name from the drop-down list. You may want to try each of the models in turn, to see which gives the lowest residual, especially if you have many data points. The residual error is an RMS fit error between the given data, and the index data generated using the resulting fit coefficients.

Now move the cursor over to the "Name" field, and enter the material name for the catalog. Select "Fit" and ZEMAX will compute the optimal coefficients. The residual RMS error, and the maximum single-point error, are listed on the bottom of the display. To enter this data in the currently loaded catalog, select "Add". ZEMAX will issue a message verifying that the glass was saved. When a glass is added to the catalog, the transmission data, if any, needs to be added as described in the next section. Otherwise, the default internal transmission of 1.0 is used at all wavelengths.

The index and wavelength data may also be saved to an ASCII file for later use, and loaded for fitting again by selecting the appropriate buttons. The ASCII file may also be edited outside of ZEMAX, and then loaded for fitting.

## Defining Transmission Data

---

Selecting the "Transmission" button invokes the transmission data editor within the glass catalog. Transmission refers to the intensity transmittance of light that depends upon the thickness of the glass as well as the wavelength. ZEMAX models the transmitted intensity using Beer's law:

$$t = e^{-\alpha\tau},$$

where  $\alpha$  is the transmission per unit length, which generally depends upon wavelength, and  $\tau$  is the path length through the glass. See the Chapter "Polarization Analysis" for information on polarization ray tracing and transmission.

The transmission is defined in the glass catalog by a series of 3 numbers: the wavelength in microns, the intensity transmittance, and the reference thickness in mm. For example, the transmission of a glass at 0.35 microns may be 0.65 for a thickness of 25 mm. Multiple data points may be defined in the transmission data editor. Internally, ZEMAX converts the data to "per mm", and interpolates between defined wavelengths. If ray tracing is being performed at a wavelength outside of the defined wavelengths, then the data for the closest wavelength is used; otherwise, ZEMAX performs a linear interpolation.

## Other glass catalog options

---

Selecting "Sort Index" sorts the glasses by the d-light (.587 microns) index of refraction, while "Sort Name" sorts by glass name (alphabetically then numerically). Calc Nd/Vd computes the Nd and Vd values based upon the dispersion coefficients. Report provides a text listing for the selected glass.

## Using MIL number glasses

---

MIL number glasses are those described by a six-digit number, such as 517640 for BK7. The first three digits in the MIL number is the d-light index minus one, without the decimal place. The last three digits is the Abbe V-number times 10. You can enter a six-digit number directly in for the glass rather than the glass name if you desire.

ZEMAX uses a formula for computing the index at each defined wavelength based upon the index and Abbe number defined by the MIL number. The formula is based upon a least-squares fit of coefficient data of many typical glasses. Typically, the index data calculated is accurate to roughly .001. At wavelengths in the deep UV or infrared the index value becomes less reliable. MIL number glasses are generally inferior substitutes for the constants of dispersion (or other) models for the glass, however they are useful if no other data is available.

It is important to note that the indices calculated from the six-digit MIL number are not the same as those calculated from the glass catalog, even if the MIL number you are using corresponds to a glass in the catalog. Index data is calculated directly from the MIL number entered on the main screen; not from the glass catalog data, even if a glass with that MIL number is in the catalog.

## Glass catalog sources

---

The data for the glass catalogs included with ZEMAX are generally provided by the manufacturers. For example, the data in the Schott catalog was provided to Focus Software by Schott Glass Technologies. The data has been converted to the ZEMAX .AGF format.

Although the data contained in the glass catalogs is generally reliable, it is always possible for errors to be made during translation or editing of the data. *It is absolutely crucial that all index data be verified for accuracy by the*

end user! This is especially true where fabrication of the optics is being considered. The catalog data may be in error, and it needs to be verified before it can be trusted.

The data contained in the INFRARED.AGF catalog has been compiled from published sources as described in the following table. A few other materials may be included in the catalog which are not referenced in the table. These materials should be especially scrutinized before trusting the data accuracy.

### INFRARED CATALOG DATA SOURCES

Material	Source
AGCL	Handbook of Optics Vol. II
ALON	Handbook of Optics Vol. II
AMTIR1	Amtir Spec Sheet
AMTIR3	Amtir Spec Sheet
BAF2	Handbook of Optics Vol. II
CACO3-O	Handbook of Optics Vol. II
CACO3-E	Handbook of Optics Vol. II
CAF2	Handbook of Optics Vol. II
CDTE	Handbook of Optics Vol. II
CLEARTRAN	Spec Sheet (Morton)
CSBR	Handbook of Optics Vol. II
GAAS	Amtir Spec Sheet
GEO2	Handbook of Optics Vol. II
GERMANIUM	JOSA, 47, 244 (57): 48, 579 (58)
IRG2	Schott Spec Sheet
IRG3	Schott Spec Sheet
IRGN6	Schott Spec Sheet
IRG7	Schott Spec Sheet
IRG9	Schott Spec Sheet
IRG11	Schott Spec Sheet
IRG15	Schott Spec Sheet
IRG100	Schott Spec Sheet
KRS5	Handbook of Optics Vol. II
LIF	Handbook of Optics Vol. II
MGF2-O	Handbook of Optics Vol. II
MGO	Handbook of Optics Vol. II
NACL	The Infrared & Electro-Optical Systems Handbook V. III
PBF2	Handbook of Optics Vol. II
Sapphire	The Infrared & Electro-Optical Systems Handbook V. III
Silicon	Handbook of Optics Vol. II
F_SILICA	The Infrared & Electro-Optical Systems Handbook V. III
SRF2	Handbook of Optics Vol. II
SRTIO3	Handbook of Optics Vol. II
ZBLA	Handbook of Optics Vol. II
ZBLAN	Handbook of Optics Vol. II
ZNSE	Handbook of Optics Vol. II
ZNS	Morton International Spec Sheet

The data contained in the MISC.AGF catalog has been compiled from published sources as described in the following table.

## MISC CATALOG DATA SOURCES

Material	Source
ACRYLIC	Handbook of Optics Vol. II
KDP-EXT	Handbook of Optics Vol. II
KDP-ORD	Handbook of Optics Vol. II
POLYCARB	Handbook of Optics Vol. II
POLYSTYR	Handbook of Optics Vol. II
QUARTZ-EXT	Handbook of Optics Vol. II
QUARTZ-ORD	Handbook of Optics Vol. II
SAN	Handbook of Optics Vol. II
SAPPHIRE	The Infrared & Electro-Optical Systems Handbook V. III
SILICA	Handbook of Optics Vol. II
ZNSE	Handbook of Optics Vol. II

### Obsolete catalog data

---

Recently, changes in environmental protection laws have required the discontinuation of the manufacture of many optical glasses. Roughly 2 out of 3 glasses which were listed in earlier catalogs of the various glass manufacturers have been discontinued, and are therefore no longer listed in the ZEMAX catalogs either.

However, much of this glass exists in optical shops and may in fact be used for new designs, if a supply can be located. Also, many existing designs need to be modeled and ray traced, especially when designing new optics to work in harmony with the existing optical system. For these reasons, the older catalogs are still provided.

All of the obsolete catalogs are given names such as OLD\_SCHO.AGF for Schott, OLD\_HOYA.AGF for Hoya, etc. The catalogs contain data for discontinued as well as still available glasses, although glasses with identical names in the old and new catalogs may have different index data, because some glasses were reformulated to comply with new environmental restrictions.

Because some new glasses may have the same name as old glasses, although the exact composition may have changed, optical engineers need to be especially vigilant about checking the index data predicted by the software against the melt sheets of the glass which will actually be used.

Never blindly trust the accuracy of the index computed using catalog coefficients. There are numerous sources of potential error, such as the measurement of the original sample, the fitting of the data, the typing in of the data in the manufacturers catalog, and then retyping into ZEMAX, and finally, the ZEMAX code itself.

This page intentionally left blank.

## Introduction

---

This feature is only available in the EE edition of ZEMAX.

Environmental factors such as ambient temperature and air pressure can affect the performance of optical systems. There are three primary factors to consider. First, the index of refraction of glass depends upon both temperature and wavelength; relative indices which are measured with respect to air also change with pressure. Second, glass expands and contracts with temperature, which can change the radius and thickness of a lens. Third, the spacings between lens elements can change due to the expansion and contraction of the mounting material.

The thermal analysis features of ZEMAX can account for all these effects. By accounting for thermal effects, ZEMAX can be used to analyze and optimize a design for any specific temperature or for a range of temperatures.

The index of refraction data given by the dispersion formulas are normally referenced to standard temperature and pressure, which is 20 degrees Celsius and 1 atmosphere. Also, index of refraction data is by convention referenced to that of air, which means thereafter the air has an index of unity. The index referenced to air is called the relative index of refraction. When the index is referenced to vacuum (which truly has an index of unity) then the index is called the absolute index of refraction. The difference between these two references for any glass is a function of wavelength, temperature, and pressure.

## Defining temperature and pressure

---

Two user provided values define the environment: the ambient temperature in degrees Celsius and the ambient air pressure measured in atmospheres. These values are set on the dialog box which can be found under System, Advanced. There is also a checkbox on this dialog box with the caption "Use T, P". If this checkbox is unchecked, then the temperature and pressure data are ignored. Ignoring the temperature and pressure data speeds up the computation of the index of refraction, so leave the box unchecked if the temperature is 20 degrees Celsius and the pressure is 1 atmosphere; the default environment settings. This standard temperature and pressure condition is abbreviated STP. If "Use T, P" is checked, then the temperature and pressure values are considered during computation of the index of refraction.

## Index of refraction computation

---

ZEMAX always uses relative, not absolute index data. Therefore, several steps are required for ZEMAX to compute the index data. The basic steps for computing the index for each glass type at each wavelength are:

- Compute the relative index of refraction of the glass at STP.
- Compute the index of air at STP.
- Compute the absolute index of the glass (relative to vacuum) at STP.
- Compute the change in absolute index of refraction of the glass at the specified temperature.
- Compute the index of air at the specified temperature and pressure.
- Compute the index of the glass relative to the air at the specified temperature and pressure.

The end result, which is the index of the glass at the specified temperature and pressure relative to air at that temperature and pressure, is what is used by ZEMAX for ray tracing. The index of a surface with no glass specified is treated as air, with unity index. Since only the ratio of the indices matters when performing ray tracing there is no need to resort to absolute index data. Note also that ZEMAX can easily model systems used in a vacuum by simply changing the ambient air pressure to zero.

Usually, there is no need to define a "vacuum" index material which has an index less than unity. There are cases however where a vacuum index material does come in handy. For example, suppose an optical system contains an air-tight cell from which the air has been evacuated. If the rest of the optical system is not in vacuum, then use an ambient air pressure of 1, and specify the "glass" for the evacuated cell to be "vacuum" (the material "vacuum"



is defined in the MISC.GLC catalog). This technique will yield correct results for any ambient temperature and pressure settings.

For the index of refraction of the glass at STP, ZEMAX uses the dispersion formulas and numerical data stored in the glass catalog. For details, see the Chapter "Using Glass Catalogs". For the index of air, ZEMAX uses the following formula:

$$n_{air} = 1.0 + \frac{(n_{ref} - 1) P}{1.0 + (T - 15) * (3.4785E-3)}$$

where

$$n_{ref} = 1 + \left[ 6432.8 + \frac{2949810\lambda^2}{146\lambda^2 - 1} + \frac{25540\lambda^2}{41\lambda^2 - 1} \right] 1.0 E-8 ,$$

T is the temperature in Celsius, P is the relative air pressure (dimensionless) and  $\lambda$  is measured in microns. This formula for the index of air is from F. Kohlrausch, *Praktische Physik*, 1968, Vol 1, page 408.

The change in absolute index of the glass with temperature is given by the following expression:

$$\Delta n_{abs} = \frac{n^2 - 1}{2n} \left[ D_0 \Delta T + D_1 \Delta T^2 + D_2 \Delta T^3 + \frac{E_0 \Delta T + E_1 \Delta T^2}{\lambda^2 - \lambda_{tk}^2} \right],$$

where n is the relative index at STP,  $\Delta T$  is the change in temperature relative to 20 degrees Celsius ( $\Delta T$  is positive if the temperature is greater than 20 degrees), and the six other constants are provided by the glass manufacturer to describe the thermal behavior of the glass.

The six constants must all be provided in the glass catalog for the computation to be valid. The default values of zero for all six constants yields zero for the change in index; therefore if no thermal data has been added to the catalog, no thermal effects are considered and the index data reverts to that for STP. ZEMAX cannot compute thermal index variation for any glass type without these six constants provided. However, some approximations are available if the six coefficients are not available. See the section "Adding thermal index variation data" later in this chapter for details.

ZEMAX does not consider environmental effects on the index of refraction of gradient index surfaces, MIL number glasses, or model glasses which are described simply by index and Abbe number.

To analyze or optimize a lens at a specific temperature and pressure, all that is required is to define the relevant data on the general lens data dialog box. All the radii and thickness data are then assumed to be measured at that temperature, and ZEMAX computes the index data accordingly.

## Defining multiple temperature and pressure values

---

The real power of the thermal analysis feature comes into play when a lens must be analyzed at or optimized for multiple environments, such as a broad temperature range or varying altitudes (or both). There are several new issues these lens systems introduce:

- A means of specifying the nominal temperature at which radii and thickness are measured must be provided.

- The change in index, radii, and thickness must be accounted for as the environment changes.

- The thermal effects on mounting material must be considered.

ZEMAX is up to the task in all particulars. The basic approach to setting up a multiple environment lens is:

- Define the lens at some nominal temperature and pressure. This is usually STP, and should be the environment under which the lens will be fabricated. All radii and thickness data will be specified only at this temperature and pressure.

- Now define additional configurations using the multi-configuration feature (see the chapter Multi-configurations). In each additional configuration, the temperature and pressure will be specified, and special solves called "TCE pickup solves" will be used to adjust the radii and thickness data for each configuration.

TCE stands for thermal coefficient of expansion. When a glass element changes temperature, the linear change in size is given by the expression

$$L' = L (1 + \alpha \Delta T) ,$$

where L is a linear dimension,  $\alpha$  is the TCE, and  $\Delta T$  is the change in temperature. As the material expands, the radius of curvature also expands. Therefore, both the thickness and the radii of a glass surface change linearly with temperature. The assumption of linearity is only an approximation, but a reasonably good one for most materials and temperature ranges.

The TCE coefficient is defined along with the glass dispersion data in the glass catalog. See the chapter "Using Glass Catalogs" for details on the TCE data.

## **Defining which parameters consider thermal effects**

On the multi-configuration editor, there is a special solve called "TCE Pick Up n" where n is the nominal configuration number. This pick up solve is used to compute a new value for a multi-configuration parameter based up on the temperature and pressure of the new configuration. TCE pick up solves only affect data for certain types of multi-configuration values, as described below.

### **Radius of curvature (CRVT) values**

If the multi-configuration operand is a CRVT, then the curvature of that surface for that configuration is computed from the curvature in the nominal configuration, the difference in temperature between the two configurations, and the TCE for that material.

If the glass type for the surface being solved for is specified in the catalog (such as BK7 or F2) then the TCE from the catalog is used. If the glass type is "air" with a blank entry in the glass column then there are two possibilities: if the previous surface glass type is a catalog glass, then the TCE for that glass is used, Otherwise, thermal effects are ignored. These rules have one important ramification: if an element is a cemented doublet, the TCE for the first radius will be that of the first glass, while the TCE for the second and third radii will be that of the second glass. In other words, ZEMAX ignores the "stress" induced in a cemented doublet. This assumption may not be accurate for a large temperature range.

### **Thickness (THIC) values**

If the operand is a THIC, then there are two possibilities. If the surface is composed of a catalog glass, then the TCE for that glass is used. Otherwise, the TCE specified in the TCE column of the surface is used. The TCE column is used for entering user-defined TCE data for the material used to manufacture the mount. There is one other important point: if the surface is not composed of a catalog glass, then the thermal expansion is computed along a length of material which extends from the edge of the surface to the edge of the next surface. Since the material expands along the length of the edges, not at the center thickness, this is a more accurate computation.

For example, suppose two lenses were separated by aluminum with a center thickness spacing of 80 mm. If the sag of the first surface (the rear of the first lens) lens was -5 mm, and the sag of the second surface (the front face of the second lens) was 8 mm, the total edge thickness is then 93 mm. If the TCE for aluminum is 23.50 E-6, then under a temperature change of +20 degrees the edge thickness would change from 93 to 93.0437 mm. Ignoring the change in sag of the two lens faces (ZEMAX actually accounts for this effect, but for simplicity it is ignored here), the center thickness would then change to 80.0437. Note that this is a different amount of expansion than if the computation had only considered the center thickness.

Because the expansion along the edge accounts for the change in radii of the adjacent surfaces, even a TCE of 0.0 will result in a change of thickness if the radii change. To turn off thermal expansion of a thickness, do not use a TCE of 0.0, rather, simply delete the thermal pickup solve entirely.

### **Parameter values**

The TCE pick up solve properties for parameter values depends upon the parameter number and surface type. If the parameter is interpreted as having units of length, then the appropriate scaling is performed just like the radius of curvature described earlier. If the units are in powers of length, such as length squared or inverse length, then appropriate scaling is also performed. Otherwise, the TCE pick up solve ignores thermal effects and simply picks up the value from the nominal configuration.

## **Extra data values**

The TCE pick up solve properties for extra data values depends upon the surface type. Generally, only the "normalization radius" needs to be scaled. This convenience is due to the fact that thermal expansion can be considered simply a scaling of length. Since most extra data surfaces use unitless coefficients, only the normalization radius needs to be scaled. This works effectively for binary optic and polynomial aspheric surfaces equally well. For extra data surface types which do not use a normalization radius, the TCE pick up solve ignores thermal effects and simply picks up the value from the nominal configuration.

## **All other values**

All other values are straight pick ups, and the value will be identical to that in the nominal configuration. Thermal effects are ignored.

## **Automatic thermal setup**

A handy means for automatically setting up a lens for thermal analysis is described in the Chapter "Editors Menu" under "Multi-configuration" in the "Tools" section.

## **Adding TCE data**

There are two kinds of TCE data. For surfaces which use a glass type named in one of the catalogs (such as BK7 from the Schott catalog), ZEMAX uses the TCE data specified in the catalog. See the chapter "Using Glass Catalogs" for the description of  $\alpha$ , the thermal coefficient of expansion value.

If the surface does not use a catalog glass, then the TCE value is extracted directly from the TCE column in the Lens Data Editor. The TCE column is the last column in the spreadsheet, to the right of the "parameter" columns.

Note that the TCE is a dimensionless parameter, and is always entered as the coefficient times 1E-6. Therefore, a value of 23.50 E-06 would be entered as simply 23.5. ZEMAX automatically considers the 1E-6 factor when computing thermal effects.

## **Adding thermal index variation data**

The variation of index of refraction with temperature, ambient air pressure, and wavelength is modeled for any glass using the polynomial expression given earlier. The expression requires six coefficients to define the temperature and wavelength dependence on the change in absolute index. It is often the case that for materials added by the user, the six coefficients describing the variation are not available. However, most glass catalogs contain at least a single linear approximation of the ratio of index change with temperature change. This value is called  $dn/dt$ . If the only data available is a single  $dn/dt$ , then an approximation to the general expression can be made assuming all the coefficients except  $D_0$  are zero:

$$\Delta n_{abs} = \frac{n^2 - 1}{2n} [D_0 \Delta T],$$

which implies that a reasonable approximation to  $D_0$  is given by

$$D_0 = \frac{2n}{n^2 - 1} \frac{dn}{dt}$$

The  $D_0$  needs to be calculated and then entered into the glass catalog. The relative index at the center wavelength measured at STP is an adequate value for the index  $n$ . Extreme care should be taken to insure this is an adequate approximation by subsequently checking the computed index values at various wavelengths and temperatures. Note that the  $dn/dt$  data should be the absolute, not the relative  $dn/dt$ .

## **Optimizing athermal lenses**

To optimize an athermal lens, first define the multiple configurations required to simulate the lens at each temperature using the methods described in the previous sections. Then define the variables for optimization only in the nominal configuration. For example, suppose the nominal configuration is number 1, and configurations 2,

3, and 4 are defined using TCE Pick Up solves for every curvature and thickness. Make only the curvatures and thicknesses in the nominal configuration variable.

It is also possible to optimize the TCE of the spacing material between lens groups. To do so, simply set a variable on the value in the TCE column of the Lens Data Editor.

## **Limitations of thermal analysis**

---

There are several limitations to the accuracy of the thermal analysis capabilities of ZEMAX. First of all, the TCE data should always be checked for accuracy over the temperature range of use. The index data coefficients should also be verified with the manufacturer of the glass being used.

The thermal analysis currently does not necessarily work correctly with tilted, decentered, or otherwise unconventional optical systems. The difficulty arises when computing edge thickness displacements on components that are not symmetric; for example, between two lenses tilted with respect to one another. The thermal analysis model works fine within ranges of the optical system which are not tilted or decentered.

This page intentionally left blank.

---

# POLARIZATION ANALYSIS

## Introduction

---

This feature is only available in the EE edition of ZEMAX.

Ray tracing programs generally treat rays as purely geometric entities, which have only a position, orientation, and phase. For example, a ray is completely described at a surface by the ray intercept coordinates, the direction cosines which define the angles the ray makes with respect to the local coordinate axes, and the phase, which determines the optical path length or difference along the ray.

At the boundary between two media, such as glass and air, refraction occurs according to Snell's law. Usually, the effects at the interface which do not affect beam direction are ignored. These effects include amplitude and phase variations of the electric field which depend upon the angle of incidence, the incident polarization, and the properties of the two media and any optical coatings at the interface.

Polarization analysis is an extension to conventional ray tracing which considers the effects that optical coatings and reflection and absorption losses have on the propagation of light through an optical system.

## Review of polarization concepts

---

The ZEMAX User's Guide is not intended to be a tutorial on polarization theory. The subject is extensive and is treated well elsewhere. See for example the review by J. Bennett in *The Handbook of Optics Volume I*, McGraw Hill, 1995. The amplitude and polarization state of the electric field is conveniently described by the Jones vector:

$$E = \begin{bmatrix} E_x \\ E_y \end{bmatrix}$$

where  $E_x$  and  $E_y$  are complex valued. The  $x$  and  $y$  directions in which the electric field vector lies are orthogonal to the propagation vector of the ray. At a boundary between two media, the transmittance, reflectance, and phase of the electric field is different for the S and P components of the field. The S component of the field is the projection that lies along the axis orthogonal to the plane of incidence, while the P projection lies within the plane of incidence. The plane of incidence contains both the ray propagation vector and the surface normal vector at the intercept point. The electric field is then divided into  $E_s$  and  $E_p$  components, both of which are complex valued.

The electric field after the boundary is computed by the consideration of the S and P transmittance (if the surface is refractive) or reflectance (if the surface is reflective). ZEMAX computes the reflectance and transmittance coefficients, but only uses the one appropriate to propagation of the ray. For brevity, the rest of this discussion will use the terms transmittance and refraction to indicate the terms of interest. If the surface is a reflector (i.e. the glass type is MIRROR) then the reflectance coefficients will be used by the program automatically. The electric field after refraction is then

$$E_s' = E_s \tau_s, \quad E_p' = E_p \tau_p,$$

where the transmittance coefficients,  $\tau_s$  and  $\tau_p$ , are complex valued. After the  $E_s$  and  $E_p$  projections are computed, they are recomposed into the Jones electric field vector and propagation continues to the next surface. The computation of the transmittance coefficients depends upon the index of refraction of the incident media, the (possibly complex) index of refraction and thickness of each layer in the optical coating (if any), and the (possibly complex) index of refraction of the substrate. The details of this computation are provided in *The Handbook of Optics Volume I*, McGraw Hill, 1995, and will not be repeated here.

There are a number of ways to define parameters of interest for describing polarized light beams. All of the expressions generally depend upon angle of incidence, wavelength, and polarization orientation. Here are the terms used frequently by ZEMAX and their definitions:

### Reflectance Amplitude, $\rho$

The reflectance amplitude is the complex valued coefficient of reflectance for the electric field.

### Transmittance Amplitude, $\tau$

The transmittance amplitude is the complex valued coefficient of transmittance for the electric field.

### Reflectance Intensity, $R$

Reflectance intensity is measured normal to the surface. It is given the symbol  $R$ , and is always real valued between 0 and 1. Reflectance can be computed from the reflectance amplitude as follows:

$$R = \rho^* \rho .$$

### Transmittance Intensity, $T$

Transmittance intensity is measured normal to the surface. It is given the symbol  $T$ , and is always real valued between 0 and 1. Transmittance can be computed from the transmittance amplitude as follows:

$$T = \tau^* \tau .$$

### Absorption Intensity, $A$

Absorption intensity is simply the intensity of the electric field which is neither transmitted nor reflected:

$$A = 1.0 - T - R .$$

### Diattenuation, $D$

Diattenuation means "two attenuations", and is used to compare the loss of intensity of the S polarized light compared to the P polarized light. Diattenuation is defined as

$$D = \left| \frac{T_s - T_p}{T_s + T_p} \right| .$$

### Phase, $P$

The phase of the transmitted beam is generally different for the S and P polarizations. The phase is given by

$$P_s = \arg ( E_s ) , P_p = \arg ( E_p ) .$$

### Retardance, $S$

The retardance is the difference in phase between the S and P polarizations, or

$$S = \arg ( E_s ) - \arg ( E_p ) .$$

## Coating description shorthand

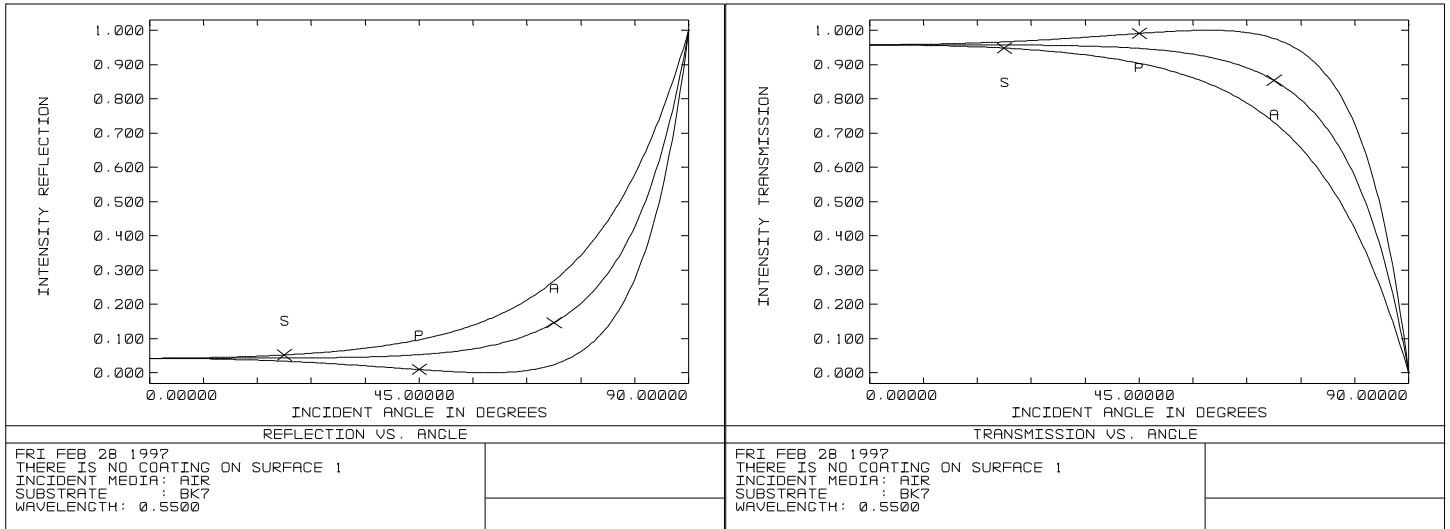
A very convenient, if incomplete, description of optical coatings is to use the abbreviations A, S, H, and L to indicate layers of optical coating materials and media. The incident media is usually given the symbol A for air, S is used for the substrate, and H and L are used for high and low index quarter wave layers. Therefore, a quarter wave coating on MgF2 on glass might be described as ALS. A "W" coating, which is a quarter wave of low index material on a half wave of high index material is described by ALHHS. What is H or L depends largely upon the index relative to the substrate.

## Properties of uncoated surfaces

Uncoated surfaces between dielectric media have transmittance and reflectance amplitudes which are described by the Fresnel expressions:

$$R = \left[ \frac{n_1 - n_2}{n_1 + n_2} \right]^2 , \text{ and } T = \frac{4n_1 n_2}{[n_1 + n_2]^2} .$$

For example, a surface between air and BK7 at a wavelength of .55 microns has reflectance and transmittance as a function of incident angle as shown in the following figures.



Note that the reflectance increases substantially with incident angle, until 100% of the light is reflected rather than transmitted at grazing incidence angles. Note also that the S and P polarization states have different transmittance. This leads to a polarization dependent apodization of the aperture.

## Defining coatings in ZEMAX

For dielectric materials, the index of refraction is purely real, and therefore the imaginary part of the index is zero. For metals, the index of refraction is complex. There are two sign conventions in common use for the imaginary part of the index of refraction. ZEMAX uses the following convention:

$$\eta = n + ik,$$

where  $n$  is the usual index of refraction and  $k$  is the extinction coefficient. The alternate sign convention in common use is  $\eta = n - ik$ , but ZEMAX uses the positive form. Therefore, materials such as aluminum have a complex index of refraction given by

$$\eta = 0.7 - 7.0i.$$

Note the extinction coefficient is negative using this convention for most materials.

ZEMAX uses a single ASCII file to define all coating data. Within the file, two types of data are provided: the name and index of individual materials, and the name and construction of individual coating types consisting of the defined materials. The file consists of the following format:

```

MATE <material name>
wavelength real imaginary
wavelength real imaginary
....
MATE <next material name>
wavelength real imaginary
wavelength real imaginary
....
COAT <coating name>
material thickness
material thickness
...
COAT <next coating name>
material thickness
material thickness
...

```



The coating and material names can be any user defined name up to 10 characters in length, with no spaces or special characters permitted.

The data is stored in a file called COATING.DAT, and a sample file is included with ZEMAX. The keywords MATE (for material) and COAT (for coating) are the only ones used. All of the material definitions come first, then all of the coating definitions. ZEMAX uses the material data in the following way:

The wavelength argument is always in microns. The wavelengths must be specified in ascending order.

The real value is the real index of refraction of the material at that wavelength.

The imaginary value is the extinction coefficient.

If a single wavelength value is provided for a material, then the real and imaginary parts of the index of refraction are used no matter what wavelength is being traced. Dispersion of the coating material is therefore ignored.

If two or more wavelengths are defined for a material, then for wavelengths shorter than the shortest defined wavelength, the shortest wavelength data is used. For wavelengths longer than the longest defined wavelength, the longest wavelength data is used. For wavelengths in between, linear interpolation is used.

ZEMAX uses the coating data in the following way:

When the COATING.DAT file is first read, ZEMAX verifies that each coating consists of materials that were defined in the material section. If referenced materials are not defined, an error is issued.

The coating thickness is measured in units of primary wavelength thickness in that medium. Therefore, if the same coating is used by two different lens files, with two different primary wavelengths, then the actual coating thicknesses will be different. The actual thickness of the coating is determined by:

$$d = \frac{\lambda_0}{n_0} T,$$

where  $\lambda_0$  is the primary wavelength in microns,  $n_0$  is the real part of the index of refraction of the coating at the primary wavelength, and T is the "optical thickness" of the coating specified in the COATING.DAT file. For example, in a coating material whose real part of the index of refraction is 1.4, a quarter wave coating (T = 0.25) at a primary wavelength of 0.550 microns would be 0.0982 microns thick. Note that only the real part of the index of refraction is used for determining the thickness of the layer.

The order of the coating layers is important. See the discussion in the next section.

There is room in the catalog for 50 materials, 50 wavelength-index points per material, 50 coating types, and up to 100 layers per coating.

There are a few ways of accessing the data in COATING.DAT. First, the file can be edited outside of ZEMAX using any ASCII text editor. Second, there is a "Edit Coating File" option under the "Tools" menu on the main menu bar. This invokes the Windows NOTEPAD editor. Also, there is a "Coating Listing" option under the "Reports" menu, which lists the coating and material definition data. If the file COATING.DAT is edited, the lens file must be saved and then opened again in order to update ZEMAX to use the new catalog data.

## Default materials and coatings supplied with ZEMAX

---

The default COATING.DAT file supplied with ZEMAX contains several materials and a few common optical coatings. The following table describes these materials and coatings.

DEFAULT MATERIALS AND COATINGS

Materials	Description
AL2O3	Aluminum Oxide, index 1.59
ALUM	Aluminum, index 0.7-7.0i
CEF3	Cerous Flouride, index 1.63
LA2O3	Lanthanum Oxide, index 1.95

Materials	Description
MGF2	Magnesium Fluoride, index 1.38
THF4	Thorium Fluoride, index 1.52
ZNS	Zinc Sulphide, index 2.35
ZRO2	Zirconium Oxide, index 2.1
Coatings	Description
AR	General anti reflection, defined as a quarter wave of MGF2.
WAR	Anti reflection "W" coat, defined as a half wave of LA2O3 followed by a quarter wave of MGF2.
HEAR1	High performance anti reflection coating
HEAR2	High performance anti reflection coating

These materials and coatings are meant to be used as examples, and may not be applicable in any particular situation. Always check with coating manufacturer or designer for details on coating material specifics. For more detailed information on how these materials and coatings are defined, use the Coating Listing report described in the chapter "Reports menu".

## Specifying coatings on surfaces

Once a coating has been defined, it can be applied to a surface by specifying the coating name in the coating column; located at the extreme right side of the lens data editor. ZEMAX interprets the coating definition using one of four rules:

If the surface specified is a boundary going from air to glass, the coating is interpreted exactly as specified in COATING.DAT, with the substrate being the glass type on the surface. The definition of the coating should not include the substrate index. The term glass here means the glass type is not "MIRROR", and not blank (which is treated as unity index for air). Gradient index lenses are considered glass.

If the surface specified is a boundary between air and air, or glass and glass, the coating is also interpreted exactly as for air to glass, with the appropriate calculations done for the incident and substrate media.

If the surface specified is a boundary going from glass to air, then the order of the layers is automatically reversed, so that the coating is the same as if it had been applied going from air to glass. Therefore, a coating defined by ALHHS, would be interpreted as SHHLA if the boundary goes from glass to air.

If the surface type is a mirror, then the coating definition must include the substrate index. The last layer in the coating definition is then assumed to be a semi-infinite thickness of substrate material.

ZEMAX may also model certain limiting cases, such as frustrated TIR; see the discussion later in this chapter.

## What ZEMAX does if no coating is specified

If the coating column is left blank, then the following is assumed:

If the surface is a mirror, then the surface is assumed to be uncoated aluminum, with an index of refraction  $0.7 - 7.0i$ .

If the surface is a dielectric interface, bare, uncoated surfaces are assumed.

## Defining the incident polarization

The individual analysis settings, such as those for the polarization ray trace (see the Analysis Menu chapter) allow for specification of the input polarization. The polarization is defined completely by the Jones vector:

$$E = \begin{bmatrix} E_x \\ E_y \end{bmatrix},$$

where  $E_x$  and  $E_y$  have both a magnitude and a phase. ZEMAX normalizes the specified  $E_x$  and  $E_y$  values to have unity magnitude, and then scales the intensity as appropriate if any pupil apodization has been specified. Note the  $E_x$  and  $E_y$  values are measured in terms of relative electric field amplitude. The  $E_x$  and  $E_y$  values describe the electric field in the local coordinate system of the ray, since  $E_z$  (in local ray coordinates) is identically zero.

The electric field is then converted from the local  $E_x$  and  $E_y$  description to the global  $E_x$ ,  $E_y$ , and  $E_z$ . Since in general the ray may not be parallel to the object space axis, the global  $E_z$  need not be zero. The conversion is performed by choosing the Jones vector component  $E_x$  to indicate the electric field along a vector  $S$ , which has no global  $Y$  component. The Jones vector component  $E_y$  indicates the electric field along a vector  $P$ , which is orthogonal to  $S$  and has no global  $X$  component. The vectors  $S$  and  $P$  are defined such that  $S \times P = R$ , where  $R$  is the ray vector. The global  $E_x$ ,  $E_y$ , and  $E_z$  are then computed from the local  $E_x$  and  $E_y$  projections along  $S$  and  $P$ . The details of this computation are handled automatically by ZEMAX, and no additional input is required. For finite object distances, the polarization vectors of rays with different pupil coordinates are not parallel, since the polarization vectors must be orthogonal to the ray direction.

## Defining polarizing components

Any boundary between two media can polarize a beam. However, ZEMAX supports an idealized model for a general polarizing device. The model is implemented as a special "Jones Matrix" surface type. The Jones matrix modifies a Jones vector (which describes the electric field) according to

$$\begin{bmatrix} E'_x \\ E'_y \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} E_x \\ E_y \end{bmatrix},$$

where  $A$ ,  $B$ ,  $C$ ,  $D$ ,  $E_x$ , and  $E_y$  are all complex numbers. In the lens data editor, ZEMAX provides cells for defining  $A$  real,  $A$  imag, etc. Note that the  $z$  component of the electric field is unaffected by the Jones matrix surface. This general Jones matrix can be used to define a wide variety of polarizing components. For example, see the sample Jones matrices in the following table. The format of the numbers is  $(a, b)$  where the  $a$  value is the real part and  $b$  is the imaginary part.

SAMPLE JONES MATRICES

Component	Matrix
Null matrix	$\begin{bmatrix} (1,0) & (0,0) \\ (0,0) & (1,0) \end{bmatrix}$
X analyzer	$\begin{bmatrix} (1,0) & (0,0) \\ (0,0) & (0,0) \end{bmatrix}$
Y analyzer	$\begin{bmatrix} (0,0) & (0,0) \\ (0,0) & (1,0) \end{bmatrix}$
Quarter-Wave plate in X direction	$\begin{bmatrix} (0,1) & (0,0) \\ (0,0) & (1,0) \end{bmatrix}$
Quarter-Wave plate in Y direction	$\begin{bmatrix} (1,0) & (0,0) \\ (0,0) & (0,1) \end{bmatrix}$
Half-Wave plate in X direction	$\begin{bmatrix} (-1,0) & (0,0) \\ (0,0) & (1,0) \end{bmatrix}$
Half-Wave plate in Y direction	$\begin{bmatrix} (1,0) & (0,0) \\ (0,0) & (-1,0) \end{bmatrix}$
Intensity filter, 50% transmission	$\begin{bmatrix} (0.5,0) & (0,0) \\ (0,0) & (0.5,0) \end{bmatrix}$

## What ZEMAX can compute using polarization analysis

ZEMAX can generate plots of R, T, A, D, P, and S as a function of either wavelength for a given incidence angle or as a function of angle given the wavelength. See the "Analysis" chapter for details.

Using the "Polarization Ray Trace" feature, ZEMAX can compute and tabulate the detailed calculations used to trace the polarization state through the system for any given ray. Other features directly compute polarization dependent results; see the "Analysis Menu" chapter for details.

ZEMAX also can accurately compute transmission of individual rays, or compute an average over the pupil. Transmission calculations are based upon surface effects, such as reflection losses, as well as bulk transmission through glass according to Beer's law:

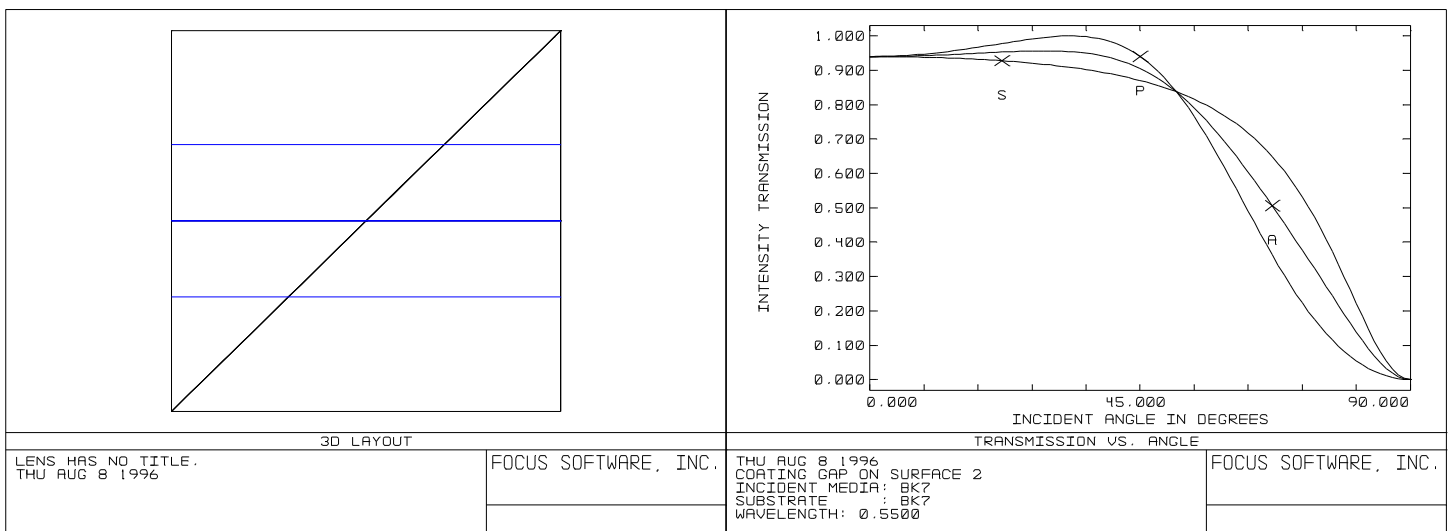
$$t = e^{-\alpha\tau},$$

where  $\alpha$  is the transmission per unit length, which generally depends upon wavelength, and  $\tau$  is the path length through the glass. See the Chapter "Using Glass Catalogs" for details on defining transmission data. Polarization effects are not considered in all ZEMAX computations.

## Modeling frustrated total internal reflection

Frustrated TIR (FTIR) occurs when a ray of light traveling through glass strikes an interface at an angle exceeding the critical angle. If another dielectric medium, such as another piece of glass, is placed close, but not touching the interface, some light will transmit through the thin gap and propagate, even though the refraction at the boundary cannot satisfy Snell's law. Both the reflected and the transmitted beams will be affected, depending upon the thickness of the gap. In the limit of the gap having zero thickness, the light will continue through as if there were no boundary. In the limit of a large gap, more than a fraction of a wavelength, then virtually all of the light is perfectly reflected.

Although this appears to be a special case, the very general model incorporated into the polarization routines of ZEMAX handle FTIR as any other coating problem. The trick is to define a coating that reflects the situation. One of the default coatings is GAP, defined to be 0.1 waves of AIR, with index unity. Using the tilted surface with a Y-Tangent of 1.0, and placing the coating GAP on the boundary yields the following system, with correctly modeled polarization properties. Note the air gap may be changed by editing the COATING.DAT file as described earlier in this chapter.



A FTIR beamsplitter in transmission. The tilted surface between the two glass blocks has a GAP coating.

Transmission vs. angle for a FTIR beamsplitter.

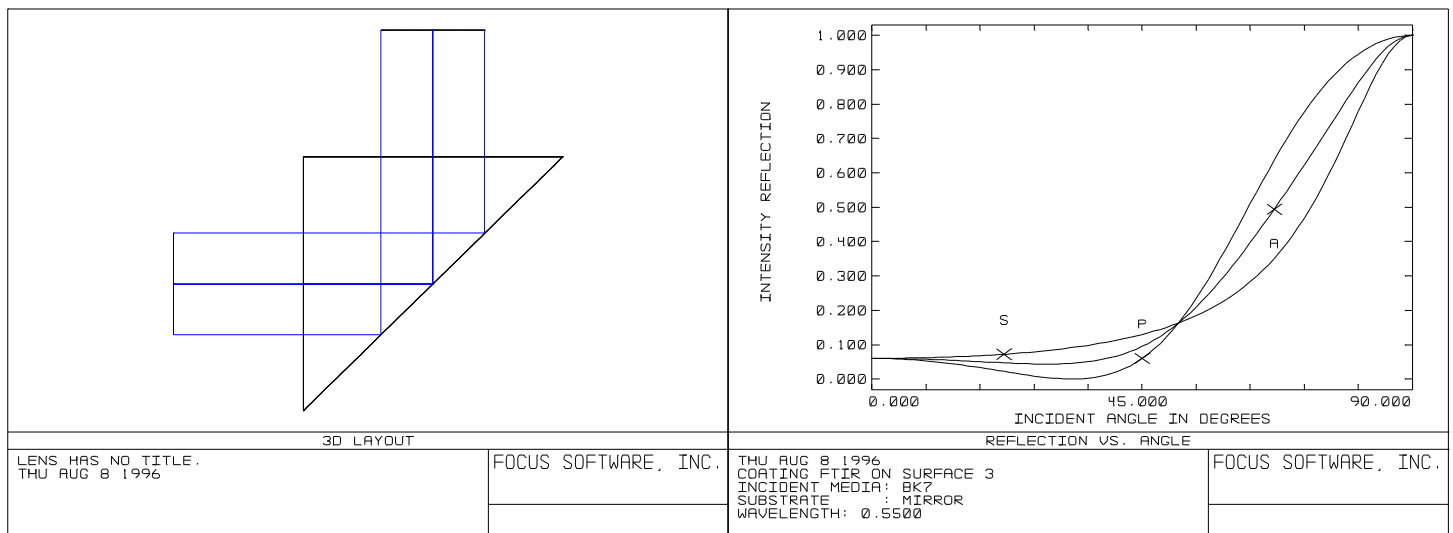
With the GAP coating in place, ZEMAX can now compute either transmitted or reflected polarization properties. However, the ray tracing for other purposes (such as computing system transmission, ray fans, optimization) will only proceed along the transmitted path. To model the reflected path, simply give the internal tilted surface a glass type of MIRROR, add a 90 degree coordinate break, and change the sign of the remaining thicknesses as usual.

The coating in this case now needs to be modified to indicate the presence of the second dielectric block, even though no ray tracing will be conducted in the transmitted path; just the reflected path. A new coating needs to be defined in COATING.DAT (called FTIR in this example) that looks like:

```
COAT FTIR
AIR .1
N15 1
```

The "coating", when applied to a mirror, treats the mirror as a gap of air followed by piece of glass. Note that detailed modeling of the index and dispersion of the substrate glass is certainly possible by definition of a glass in COATING.DAT; here N15 was used for simplicity. The thickness of the last material, N15, which defines the substrate, is irrelevant because ZEMAX always assumes the substrate is semi-infinite when applied to a MIRROR. The resulting lens figure and reflection plot is shown. Both the reflection and transmission plots are of course identical for either system model.

Finally, note that conventional TIR can also be modeled by simply increasing the gap thickness. Distances much greater than a fraction of a wavelength are essentially equivalent to an infinite gap. However, very large thicknesses for evanescent wave propagation should not be defined in COATING.DAT because of "underflow" numerical errors which arise because the transmission is exceedingly low. One wave, or an optical thickness of 1.0, is usually sufficient.



A FTIR beamsplitter in reflection. The tilted "mirror" surface has a FTIR coating.

Reflection vs. angle for a FTIR beamsplitter.

## Limitations of polarization analysis

There are a few limitations to the polarization analysis capabilities of ZEMAX. The data in the COATING.DAT file should always be carefully checked for accuracy whenever fabrication decisions are being made. Also, the reflectivity and transmission curves generated by ZEMAX should always be presented to the coating manufacturer as a check to insure the coatings will behave as intended.

The polarization ray tracing algorithm generally does not return correct results for certain types of surfaces, such as diffraction gratings, binary optics, or Fresnel surfaces. Generally, only surfaces which have refraction or reflection described by Snell's law have valid polarization data computed. The polarization effects of propagation through gradient index or birefringent media is also not considered.

This page intentionally left blank.

---

# ZEMAX PROGRAMMING LANGUAGE

## Introduction

---

This feature is only available in the XE and EE editions of ZEMAX.

The ZEMAX Programming Language (ZPL) is a simple to use macro language specifically designed for use with ZEMAX. ZPL offers the power of user-extensibility. This means that if you need a particular calculation or graphical display which is not "built in", you can write your own ZPL program to do the job. ZPL programs can be stored on disk and called from within ZEMAX. You can build up a library of ZPL programs, and share them with other ZEMAX users.

ZPL is similar to the BASIC programming language, except not all BASIC constructs and keywords are supported, and some new capabilities and functions unique to ray tracing have been added. If you are familiar with BASIC, you will learn ZPL quickly. If you are not, or if you have never programmed before, don't panic! ZPL is easy to use, and this chapter will give you simple instructions and examples to get you started.

## Creating ZPL programs

---

To create a ZPL program, it is probably easiest to start with an existing program that performs a task similar to the one you want to achieve. If you are attempting to write your first ZPL program, you may want to read the example sections at the end of this chapter. Some example ZPL macros can also be found in the ZEMAX directory, they are described in the chapter "ZEMAX example files".

Use any ASCII text editor to create the ZPL file (such as the NOTEPAD.EXE editor). Name the file any descriptive title you wish, but the file must end in the .ZPL extension.

## Running ZPL programs

---

To run your ZPL program, from the main menu, select Tools, ZPL Macros. The ZPL control dialog box will appear with the following options:

**Active File:** A drop-down list of macros available. All the listed macros are ASCII files ending in the extension .ZPL. The files must be in the default directory for ZPL Macros; see the Chapter "File Menu" in the section on Environment, Directories.

**Close After Execution:** If checked, the ZPL control dialog box will automatically close after the macro execution.

**Status:** During execution of the macro, ZEMAX will use this area to print a status message stating the line number of the macro being executed. The line number is computed after any blank lines or comments are removed. The status messages is updated every quarter second.

**Terminate:** The terminate button will stop execution of the macro currently running.

**Cancel:** The cancel button terminates the current macro if one is running. If no macro is running, cancel closes the ZPL control dialog box.

**Edit:** The edit button invokes the Windows NOTEPAD.EXE editor, which must be in the main Windows directory. The editor can be used to modify or rename the macro.

The view button will display the contents of the macro file in a text window which can be scrolled or printed. No editing is allowed in the view window.

Select the macro to run from the "Active File" list, and then click on Execute.

ZEMAX will begin running the program. The entire program will run, and any text output from PRINT statements or error messages will be placed in a text file. When the macro has terminated, the text file will be displayed in a window. The display of this output window can be suppressed using the CLOSEWINDOW keyword.

## An overview of ZPL

---

A ZPL program consists of a series of commands (called statements) which are stored in an ASCII text file. You can create a ZPL program with any text editor outside of ZEMAX (you can also edit the program from within ZEMAX; this will be discussed in a subsequent section). The ZPL language uses a syntax similar to BASIC for most (but not all) commands. For example, the statement

```
x=5
```

is one type of valid ZPL statement. There are several important things to note in this simple statement. First, no declaration of variables is required. This means "X", which is called a variable, did not need to exist before the value of 5 was assigned to it. If "X" had already been assigned a value, it would now be reassigned. There is no need to declare the variable before using it. Second, no special symbol is required to terminate a statement, such as ";" in C. Because of this, each ZPL statement must be on a line by itself.

ZPL supports "inline" function calls. This allows statements of the following forms to be used:

```
x=SQRT( 5 )
y=SINE( x )
z=SQRT( x+5*( 7-x ) )
```

The functions SQRT (square root) and SINE (sine) are built in to ZPL. There are many of these single-operand functions, all of which are defined in a subsequent section. Note that ZPL is not case-sensitive; SQRT() and sqrt() are the same function. This documentation will use the convention of capital letters for functions and keywords, and lower case for everything else.

All of the previously shown statements have one thing in common: they are all assignments. This means the expression on the right side of the equals sign is evaluated, and the result is assigned to the variable designated on the left. There is one other kind of statement in ZPL, and it is called a keyword.

One example of a keyword is PRINT. The PRINT keyword is followed by a list of items, separated by commas, to be printed. For example, the ZPL statements

```
x = 3
y = 4
z = SQRT( x * x + y * y )
PRINT "The hypotenuse is ",z
```

will print the following to the screen:

```
The hypotenuse is 5.0000
```

Note that ZPL enforces operator precedence. Within any pair of parentheses, operations are carried out in a particular order. ZPL uses the following precedence from highest to lowest: functions (such as SQRT), logical operators (such as ==), multiplication and division, and then addition and subtraction. Parentheses always override precedence, in this sense they have the highest priority.

There are many other keywords which will be described in a subsequent section.

## ZPL variables

---

Variables provide temporary storage for numerical quantities whose exact value may not be known when the program is written, but is defined when the program is run. ZEMAX performs most of the work for you when you need a new variable. For example, the simple statement

```
x=5
```

presented earlier will cause ZEMAX to allocate memory for the new variable "x" and keep track of the value associated with it. Once the variable is defined, it may be used in any subsequent expression. There are however a few rules regarding ZPL variables.

First, variable names must not contain any "special" characters that ZPL uses for logical operations or delimiting such as (, ), =, +, -, \*, /, !, >, <, ^, &, |, #, ", and the space character. Second, a variable cannot take on the same name as a keyword or function, such as THIC or RAYX. Since ZPL is not case-sensitive, you cannot use rayX or Thic to avoid this rule. Third, each variable name is limited to 28 characters. Violating these rules may generate



a syntax error, however the program may appear to run without errors and will simply not compute what you really want. The maximum number of local ZPL variables is 200. All ZPL variables are stored internally as 64-bit double precision numbers.

## ZPL functions

ZPL functions can be used on the right hand side of an assignment, and in expressions which are arguments to keywords. These functions may require no arguments, one argument, or multiple arguments. All functions return a single value. Certain functions, such as PWAV() (primary wavelength), return a value which does not depend upon the argument, and therefore it is not required to provide one. The parentheses however, are still required.

Many functions have corresponding keywords. For example, RADI() is a function which returns the radius of the surface corresponding to the argument (RADI(3) returns the radius of surface 3). There is also a keyword called RADI which is used to set the radius of that surface. See the keyword descriptions for more information.

In the following table, all ZPL functions are listed. If the syntax is given as FUNC(), then no arguments are required. FUNC(x) indicates one argument is required, FUNC(x,y) indicates two arguments, etc.

ZPL FUNCTION LISTING

Function	Argument	Return value
ABSO(x)	Numeric expression	The absolute value of the expression.
ACOS(x)	Numeric expression	Arc cosine in radians.
APMN(x)	Valid surface number	The minimum radius value. For spider apertures, this is the width of the arms. For rectangular and elliptical apertures, it is the x-half width of the aperture.
APMX(x)	Valid surface number	The maximum radius value. For spider apertures, this is the number of arms. For rectangular and elliptical apertures, it is the y-half width of the aperture.
APXD(x)	Valid surface number	The aperture x-decenter value.
APYD(x)	Valid surface number	The aperture y-decenter value.
APTP(x)	Valid surface number	An integer code describing the aperture type on the specified surface.
ASIN(x)	Numeric expression	Arc sine in radians.
ASPR()	(null)	Aspect ratio of the current graphics device.
ATAN(x)	Numeric expression	Arc tangent in radians.
ATYP()	(null)	System aperture type code: 0 for EPD, 1 for F/#, 2 for N.A. and 3 for float by stop size.
AVAL()	(null)	System aperture value.
CONF()	(null)	Returns the current configuration number, which is between 1 and the number of configurations, inclusive.
CONI(x)	Valid surface number	Conic constant of the surface.
COSI(x)	Numeric expression in radians	Cosine of the expression.
CURV(x)	Valid surface number	Curvature of the surface.
EDGE(x)	Valid surface number	Edge thickness at the semi-diameter of that surface.
EOFF()	(null)	End of file flag. Returns 1 if the end of file has been reached, otherwise returns 0. Only valid after execution of READ keyword.

Function	Argument	Return value
ETIM()	(null)	The elapsed time in seconds since last TIMER.
EXPE(x)	Numeric expression	e to the power of the expression.
EXPT(x)	Numeric expression	10 to the power of the expression.
FLDX(x)	Valid field number	X-angle or height of specified field.
FLDY(x)	Valid field number	Y-angle or height of specified field.
FTYP()	(null)	Returns 0 if the current field type is angles in degrees, 1 if object height, and 2 if image height. Heights are always in lens units (see UNIT).
FVCX(x), FVCY(x)	Valid field number	Vignetting compression x or y of specified field.
FVDX(x), FVDY(x)	Valid field number	Vignetting decenter x or y of specified field.
FWGT(x)	Valid field number	Field weighting of specified field.
GABB(x)	Valid surface number	The glass catalog Abbe number of the glass for the specified surface.
GIND(x)	Valid surface number	The glass catalog d-light index of the glass for the specified surface.
GLCA(x)	Valid surface number	Global vertex x vector of the specified surface.
GLCB(x)	Valid surface number	Global vertex y vector of the specified surface.
GLCC(x)	Valid surface number	Global vertex z vector of the specified surface.
GLCX(x)	Valid surface number	Global vertex x-coordinate of the specified surface.
GLCY(x)	Valid surface number	Global vertex y-coordinate of the specified surface.
GLCZ(x)	Valid surface number	Global vertex z-coordinate of the specified surface.
GNUM(A\$)	Any string variable name	If the string A\$ is the name of a valid glass, such as BK7, then GNUM returns the number of the glass in the glass catalog. The glass number can subsequently be used by GLAS to set the glass type on a surface. If A\$ does not correspond to any glass in the catalog, GNUM returns 0. GNUM returns -1 if the string is "MIRROR".
GPAR(x)	Valid surface number	The glass catalog partial dispersion coefficient of the glass for the specified surface.
INDX(x)	Valid surface number	Index of refraction at the primary wavelength.
INTE(x)	Numeric expression	Returns the largest integer not greater than the argument.
LOGE(x)	Positive numeric expression	Log base e of the expression.
LOGT(x)	Positive numeric expression	Log base ten of the expression.
MAGN(x,y)	x and y are any real numbers	Computes the square root of x squared plus y squared.
MAXF()	(null)	The maximum radial field angle in degrees, or radial object height in lens units. Which value is returned depends upon if the fields are defined by angle or height.
MAXG()	(null)	The number of glasses currently loaded.

Function	Argument	Return value
MFCN()	(null)	Returns the current merit function value.
NCON()	(null)	Returns the number of configurations.
NFLD()	(null)	The number of defined fields.
NSUR()	(null)	The number of defined surfaces.
NWAV()	(null)	The number of defined wavelengths.
OPDC()	(null)	The optical path difference. Valid only after RAYTRACE call.
OPTH(x)	Valid surface number	The optical path length along the ray to the specified surface. Unlike RAYT and RAYO, OPTH considers the phase added by diffractive surfaces such as gratings, holograms, and binary optics. Valid only after RAYTRACE call.
PARn(x)	Valid surface number	Parameter "n" of surface.
PMOD()	(null)	0 if paraxial mode is off, else 1.
POWR(x,y)	x is positive, y is any number	Computes x to the power of y. X must be positive, or the absolute value is used.
PVHX()	(null)	"hx" parameter from the ZPLM optimization operand.
PVHY()	(null)	"hy" parameter from the ZPLM optimization operand.
PVPX()	(null)	"px" parameter from the ZPLM optimization operand.
PVPY()	(null)	"py" parameter from the ZPLM optimization operand.
PWAV()	(null)	The primary wavelength number.
RADI(x)	Valid surface number	Radius of curvature of surface.
RAGX(x)	Valid surface number	The global X-coordinate of the ray intercept. Valid only after RAYTRACE call.
RAGY(x)	Valid surface number	The global Y-coordinate of the ray intercept. Valid only after RAYTRACE call.
RAGZ(x)	Valid surface number	The global Z-coordinate of the ray intercept. Valid only after RAYTRACE call.
RAND(x)	Positive numeric expression	Random floating point number uniformly distributed between 0 and the expression.
RANX(x)	Valid surface number	The X-direction cosine of the surface normal. Valid only after RAYTRACE call.
RANY(x)	Valid surface number	The Y-direction cosine of the surface normal. Valid only after RAYTRACE call.
RANZ(x)	Valid surface number	The Z-direction cosine of the surface normal. Valid only after RAYTRACE call.
RAYE()	(null)	The ray-trace error flag, 0 if no error. Valid only after RAYTRACE call. See the RAYTRACE keyword for details.
RAYL(x)	Valid surface number	The X-direction cosine of the ray following the surface. Valid only after RAYTRACE call.
RAYM(x)	Valid surface number	The Y-direction cosine of the ray following the surface. Valid only after RAYTRACE call.

Function	Argument	Return value
RAYN(x)	Valid surface number	The Z-direction cosine of the ray following the surface. Valid only after RAYTRACE call.
RAYO(x)	Valid surface number	The ray optical path length from the previous surface to the specified surface. The optical path length is the path length times the index of refraction, either or both of which may be negative. Valid only after RAYTRACE call. See also OPTH and RAYT.
RAYT(x)	Valid surface number	The ray path length from the previous surface to the specified surface. The path length may be negative. Valid only after RAYTRACE call. See also OPTH and RAYO.
RAYV()	(null)	0 if ray was not vignetted, else vignetted surface number. Valid only after RAYTRACE call.
RAYX(x)	Valid surface number	The X-coordinate of the ray intercept. Valid only after RAYTRACE call.
RAYY(x)	Valid surface number	The Y-coordinate of the ray intercept. Valid only after RAYTRACE call.
RAYZ(x)	Valid surface number	The Z-coordinate of the ray intercept. Valid only after RAYTRACE call.
SAGG(x,y,z)	x,y are the coordinates in lens units at the surface number z	Computes the sag in lens units at the specified point on the surface.
SCOD(A\$)	Any string variable or literal that is the name of a ZEMAX surface type.	The string name must be a valid surface name, such as STANDARD, EVENASPH, PARAXIAL, etc. The valid names are displayed on the prescription data report for each surface type in use.
SDIA(x)	Valid surface number	Semi-diameter of surface.
SIGN(x)	Numeric expression	Returns -1 if the argument is less than zero, 0 if the argument is zero, and +1 if the argument is positive.
SINE(x)	Numeric expression in radians	Sine of the expression.
SLEN(A\$)	Any string variable name	The number of characters in the string variable A\$
SQRT(x)	Positive numeric expression	Square root of the expression.
STYP(n)	Surface number n	The surface type code of the surface. See SCOD.
TANG(x)	Numeric expression in radians	Tangent of the expression.
THIC(x)	Valid surface number	Thickness of the surface.
UNIT()	(null)	Returns 0, 1, 2, or 3, if the current unit type is millimeters, centimeters, inches, or meters, respectively.
VEC1(x)	Positive subscript value	Returns the value of the array variable at the specified subscript.
VEC2(x)	Positive subscript value	Returns the value of the array variable at the specified subscript.
VEC3(x)	Positive subscript value	Returns the value of the array variable at the specified subscript.
VEC4(x)	Positive subscript value	Returns the value of the array variable at the specified subscript.

Function	Argument	Return value
VERS()	(null)	Returns the four digit version code. A typical version number is 5500. Subsequent versions of ZEMAX will return larger version codes.
WAVL(x)	Valid wavelength number	Wavelength in microns.
WWGT(x)	Valid wavelength number	Wavelength weighting.
XMIN()	(null)	The minimum x-coordinate in the graphics window.
XMAX()	(null)	The maximum x-coordinate in the graphics window.
YMIN()	(null)	The minimum y-coordinate in the graphics window.
YMAX()	(null)	The maximum y-coordinate in the graphics window.

## ZPL math operations

ZPL macros can include basic math operations such as add, subtract, multiply, and divide. The syntax for each is shown below.

```
x = y + z
x = y - z
x = y * z
x = y / z
```

All other operations are supported only through the use of ZPL functions or ZPL logical operators, both described in subsequent sections.

## ZPL logical operators

Logical operators are used to construct complex statements which ultimately evaluate to one or zero. Most logical operations take the form (left\_expression) (operator) (right\_expression), similar to mathematical statements such as  $1 + 2$ . The exception is the not operator "!" which takes only a single argument, of the form !(right\_expression). The logical operators use the convention that zero is "false" and any non-zero value is "true". The not operator returns 1 (true) if the (right\_expression) is 0 (false) and returns 0 (false) if (right\_expression) is non-zero (true). One common use of the not operator is in IF statements such as:

```
IF !x THEN PRINT "x is zero."
```

The other logical operators can be also be used as part of the argument in IF statements. For example, an IF statement may contain two conditions which must both be true for the THEN statement to be executed:

```
IF ( x > 1 ) & ( y < 2 ) THEN PRINT "Both conditions are true."
```

These two conditions are related by an "and" expression denoted by &. Note the parentheses are used to force precedence. ZPL supports the logical operators described in the following table.

### ZPL LOGICAL OPERATORS

Logical	Description
&	And, returns 1 only if both expressions are non-zero.
	Or, returns 1 if at least one expression is non-zero.
^	Xor, returns 1 if only one expression is non-zero.
!	Not, returns 0 if (right_expression) is non-zero, else returns 1.
==	Equality, returns 1 if expressions are equal.

Logical	Description
>	Greater than, returns 1 if left_expression is greater than right_expression.
<	Less than, returns 1 if left_expression is less than right_expression.
>=	Greater than or equal to, returns 1 if left_expression is greater than or equal to right_expression.
<=	Less than or equal to, returns 1 if left_expression is less than or equal to right_expression.
!=	Inequality, returns 1 if expressions are unequal.

## ZPL string variables and operations

ZPL supports string variables and basic string operations. String variables do not need to be declared, but can be created at any time using a defining statement such as:

```
newstring$ = "Here is the new string"
```

Note that string variables are distinguished from numeric variables by the presence of the \$ character at the end of the string. String variables can be concatenated using the + operator. The syntax is:

```
C$ = A$ + B$
```

Concatenation can also include constant strings:

```
total$ = "A$ is " + A$ + " and B$ is " + B$
```

String variables are printed just like other strings:

```
PRINT "Here is A$: ", A$
```

Note that the PRINT function can only print single string variables; there is no support for the concatenation operand inside print statements. The correct procedure is to concatenate the strings into a new string and then print the new string:

```
A$ = B$ + C$
PRINT A$
```

Alternatively, the comma acts as a concatenation operand:

```
PRINT A$, B$, C$
```

There are also string functions which can be used to extract text data, such as

```
title$ = $LENSNAME()
```

Note the function \$LENSTITLE() starts with the \$ character. This identifies the function as returning a string result. The string functions can be used in a defining statement such as

```
this$ = "Here is the lens title: " + $LENSNAME()
```

One very important function is \$STR(). This function accepts as an argument any expression which evaluates to a number. \$STR is useful for formatting numeric data into strings:

```
A$ = "The expression evaluates to " + $STR( SQRT(3*3 + 4*4 + z*y) )
```

The following table lists the available string functions.

### ZPL STRING FUNCTIONS

Function	Description
\$COMMENT(i)	Returns the comment string for the ith surface.
\$DATE()	Returns the current date and time string. The formatting is specified on the environment dialog box.

Function	Description
\$FILENAME()	Returns the current lens file name, without the path.
\$FILEPATH()	Returns the current lens file name, with the complete path.
\$GLASS(i)	Returns the glass name of surface number i.
\$LENSNAME()	Returns the lens title defined in the General System dialog box.
\$PATHNAME()	Returns the path name only for the current lens file. This is useful for determining the directory where the lens file is stored.
\$STR(expression)	Returns a string formatted using the format defined by the FORMAT keyword. The numeric expression may be any equation, including combinations of constants, variables, and functions.
\$UNITS()	Returns either MM, CM, IN, or M, depending upon the current lens units.

## ZPL keywords

ZPL keywords provide the capability to direct program flow, to generate output, and to perform certain crucial tasks, such as ray tracing and modifying the lens prescription. Each keyword is described in detail in the following sections. Certain keywords with similar or related functions are grouped together.

### **APMN, APMX**

#### *Purpose:*

Used to set the aperture minimum/maximum radius of a surface.

#### *Syntax:*

APMN(surface) = (new\_value)

APMX(surface) = (new\_value)

#### *Discussion:*

These keywords requires two expressions, one to specify the surface number and the other to define the new value. The expression (surface) is evaluated and then rounded down to an integer to yield the surface number. If the surface number is less than zero or greater than the number of surfaces, the command is ignored. Otherwise, the expression (new\_value) is evaluated, and assigned to the aperture minimum (APMN) or aperture maximum (APMX) radius.

The aperture minimum/maximum radius data is used by all of the aperture types, although the meaning of the data depends upon which aperture is being used, as described in the following table.

Aperture type	APMN is used for:
None.	Nothing
Circular aperture	Setting the minimum aperture radius
Circular obscuration	Setting the minimum aperture radius
Spider	Setting the width of each spider arm
Rectangular aperture	Setting the x-half width
Rectangular obscuration	Setting the x-half width
Elliptical aperture	Setting the x-half width
Elliptical obscuration	Setting the x-half width

#### *Example:*

APMN(3) = 1.75

APMX(3) = 3.50

## **APXD, APYD**

### *Purpose:*

Used to set the aperture x/y decenter of a surface.

### *Syntax:*

```
APXD(surface) = (new_value)
```

```
APYD(surface) = (new_value)
```

### *Discussion:*

These keywords requires two expressions, one to specify the surface number and the other to define the new value. The expression (surface) is evaluated and then rounded down to an integer to yield the surface number. If the surface number is less than zero or greater then the number of surfaces, the command is ignored. Otherwise, the expression (new\_value) is evaluated, and assigned to the aperture x-decenter (APXD) or aperture y-decenter (APYD).

### *Example:*

```
APXD(3) = -3.6
```

```
APYD(5) = -1 * APYD(3)
```

## **ATP**

### *Purpose:*

Used to set the aperture type of a surface.

### *Syntax:*

```
ATP(surface) = integer_code
```

### *Discussion:*

This keyword requires two expressions, one to specify the surface number and the other to define the new value. The expression (surface) is evaluated and then rounded down to an integer to yield the surface number. If the surface number is less than zero or greater then the number of surfaces, the command is ignored. Otherwise, the expression (new\_value) is evaluated, rounded to an integer and assigned to the aperture type.

The aperture type is stored as an integer code described in the following table.

Integer code	Aperture type
0	None.
1	Circular aperture
2	Circular obscuration
3	Spider
4	Rectangular aperture
5	Rectangular obscuration
6	Elliptical aperture
7	Elliptical obscuration

### *Example:*

```
ATP(3) = 1
```



## **ATYP, AVAL**

### *Purpose:*

Used to set the system aperture type and value.

### *Syntax:*

```
ATYP = (code)
AVAL = (new_value)
```

### *Discussion:*

These keywords are used to define the system aperture type and value. The aperture type is defined using one of the following codes:

Aperture type	Code
Entrance Pupil Diameter	0
Image Space F/#	1
Object Space Numerical Aperture	2
Float By Stop Size	3

### *Example:*

```
! Set the EPD to be 35
ATYP = 0
AVAL = 35.0
```

## **CLOSE**

### *Purpose:*

Closes an ASCII file previously opened by the OPEN command.

### *Syntax:*

```
CLOSE
```

### *Discussion:*

See the description for OPEN.

## **CLOSEWINDOW**

### *Purpose:*

Suppresses the display of the default output window.

### *Syntax:*

```
CLOSEWINDOW
```

### *Discussion:*

CLOSEWINDOW is used to run the ZPL macro in "quiet" mode. The text window normally displayed at the end of the macro execution will not be displayed if the CLOSEWINDOW keyword is included at any line in the macro. CLOSEWINDOW has no other effect on macro execution.

## **COLOR**

### *Purpose:*

Sets the current pen color to use for graphics line and text functions.

### *Syntax:*

```
COLOR n
```

*Discussion:*

If the argument *n* is zero, the color will be set to black. Otherwise, the color defined by the integer value will be used for all subsequent line and text commands in graphics mode.

**CONI**

*Purpose:*

CONI is used to set the conic constant of a surface.

*Syntax:*

```
CONI (surface) = (new_value)
```

*Discussion:*

This keyword requires two expressions, one to specify the surface number and the other to define the new value. The expression (surface) is evaluated and then rounded down to an integer to yield the surface number. If the surface number is less than zero or greater than the number of surfaces, the command is ignored. Otherwise, the expression (new\_value) is evaluated and assigned.

*Example:*

```
CONI 1 = -1
```

*Related keywords:*

UPDATE

**CURV**

*Purpose:*

CURV is used to set the curvature of a surface.

*Syntax:*

```
CURV (surface) = (new_value)
```

*Discussion:*

This keyword requires two expressions, one to specify the surface number and the other to define the new value. The expression (surface) is evaluated and then rounded down to an integer to yield the surface number. If the surface number is less than zero or greater than the number of surfaces, the command is ignored. Otherwise, the expression (new\_value) is evaluated and assigned.

*Example:*

```
CURV(3) = - CURV(4) + .001
```

*Related keywords:*

UPDATE

**DELETE**

*Purpose:*

Deletes a surface from the spreadsheet.

*Syntax:*

```
DELETE n
```

*Discussion:*

The value *n* must evaluate to an integer expression. See also INSERT.

*Example:*

```
DELETE 5  
DELETE i+2*j
```

**EDVA**

*Purpose:*

Sets the extra data values for a surface.

*Syntax:*

```
EDVA surf, ed_value, new_value
```

*Discussion:*

The surf expression indicates which surface is to be modified. The ed\_value expression must evaluate to one of the extra data value positions, between 1 and 99. See the chapter "Surface types" for information on the extra data values. The new\_value expression is evaluated and is used for the new value of the extra data value on that surface.

*Example:*

```
EDVA 5, 6, x+y
```

**END**

See GOSUB.

**FINDFILE**

*Purpose:*

Used to find the names of files stored on disk.

*Syntax:*

```
FINDFILE TEMPNAME$, FILTER$
```

*Discussion:*

This keyword requires two expressions, one to specify the string variable name to store the file name in, and another string variable which contains a "filter" string. The filter string usually specifies a path name and wildcards appropriate to the desired file type. See the example below.

FINDFILE is useful for listing all files of a certain type in a directory, or for analyzing large numbers of similar lens files.

*Example:*

```
FILTER$ = "C:\ZEMAX-EE\*.ZMX"  
PRINT "Listing of all ZEMAX files in ", FILTER$  
FINDFILE TEMPFILE$, FILTER$  
LABEL 1  
if (SLEN(TEMPFILE$))  
    PRINT TEMPFILE$  
    FINDFILE TEMPFILE$, FILTER$  
    GOTO 1  
ENDIF  
PRINT "No more files."
```

## **FLDX, FLDY, FWGT, FVDX, FVDY, FVCX, FVCY**

### *Purpose:*

FLDX and FLDY are used to change the field point definitions. FWGT is used to change the field weight. FVDX and FVDY change the x and y vignetting decenter factors. FVCX and FVCY change the x and y vignetting compression factors.

### *Syntax:*

```
FLDX (field number) = (new_value)
FLDY (field number) = (new_value)
FWGT (field number) = (new_value)
FVDX (field number) = (new_value)
FVDY (field number) = (new_value)
FVCX (field number) = (new_value)
FVCY (field number) = (new_value)
```

### *Discussion:*

These keywords require two expressions, one to specify the field number and the other to define the new value. The expression (field number) is evaluated and then rounded down to an integer to yield the field number. If the field number is less than 1 or greater than the number of fields, the command is ignored. Otherwise, the expression (new\_value) is evaluated and assigned. Because ray coordinates and surface semi-diameters are effected by the change in field values, it is recommended that the UPDATE command be used after changing any field settings.

### *Example:*

```
FLDX 1 = 0.0
FLDY 1 = 12.5
FWGT 1 = 1.0
```

### *Related functions:*

FLDX, FLDY, FWGT, MAXF, FTYP, FVDX, FVDY, FVCX, FVCY

### *Related keywords:*

UPDATE, FTYP

## **FOR, NEXT**

### *Purpose:*

The FOR and NEXT keywords define a program block which is executed a specific number of times in a loop.

### *Syntax:*

```
FOR variable = start_value, stop_value, increment
    statements...
NEXT
```

### *Discussion:*

The keyword FOR marks the beginning of a group of statements to be executed a multiple number of times. FOR requires a variable to be specified which acts as a counter (it need not be an integer), a starting value for the counter, a stop value, and an increment. The NEXT keyword marks the end of the group of statements. FOR-NEXT loops may be nested. The number of FOR and NEXT statements must be the same.

Upon reaching a FOR statement, the expressions for the start, stop, and increment values are evaluated and saved. The stop and increment values are not evaluated again, even if the expressions defining the values consist of variables whose values change within the program block. Only the values valid at the beginning of the FOR loop are used.

If the start value and stop value are the same, the loop executes exactly once. If the start value is less than the stop value, then the loop continues until the counter variable is greater than the stop value. If the start value is greater than stop value, then the loop continues until the counter variable is less than the stop value.

*Example:*

```
FOR i = 1, 25, 1
    PRINT i
NEXT

j = 5
k = 0
FOR i = j, j + 5, 2
    k = i + j + k
NEXT
```

**FORMAT**

*Purpose:*

Specifies the numerical precision format for subsequent PRINT statements.

*Syntax:*

```
FORMAT m.n [EXP]
```

*Discussion:*

The integers m and n are separated by a decimal point. The value m refers to the total number of characters to be printed, even though some of them may be blank. The value n refers to the number of places to display after the decimal point. Therefore, FORMAT 8.4 will cause subsequent PRINT statements to print 8 characters, with 4 numbers after the decimal point. FORMAT .5 will cause PRINT to show 5 decimal places, and as many total places as needed. FORMAT only affects numeric output from PRINT. If a number is too large to fit within the m decimal places, then the m portion of the FORMAT statement will be ignored. The optional keyword EXP after the m.n expression indicates exponential notation should be used.

*Example:*

```
x = 1.123
FORMAT 12.0
PRINT "An integer portion=", x
FORMAT 12.8
PRINT "A decimal portion = ", x
FORMAT 12.8 EXP
PRINT "Exponential notation = ", x
```

**FTYP**

*Purpose:*

FTYP is used to change the type of field definition between angle in degrees, object height, and image height.

*Syntax:*

```
FTYP = (new_value)
```

*Discussion:*

The (new value) must be 0, 1, or 2. Use 0 to set the field type to angle in degrees, 1 for object height in lens units, and 2 for image height in lens units. Setting the field type does not change the field values, nor is the system updated. Usually, the field values must be adjusted with the FLDX and FLDY keywords whenever the field type is changed.

*Related functions:*

FLDX, FLDY, FWGT, MAXF

*Related keywords:*

UPDATE, FLDX, FLDY, FWGT

### **GDATE**

*Purpose:*

GDATE will place the current date under the lens name in the text box on user defined graphics screens.

*Syntax:*

GDATE

*Discussion:*

GTITLE is primarily used for making your graphics look like other ZEMAX graphics.

*Example:*

See the section GRAPHICS.

### **GETEXTRADATA**

*Purpose:*

Retrieves the extra data values from the extra data editor. The data is placed in one of the vector array variables (either VEC1, VEC2, VEC3, or VEC4).

*Syntax:*

GETEXTRADATA vector\_expression, surface\_expression

*Discussion:*

The data is stored in the specified VECn array variable. For example, if the command GETEXTRADATA 1, 5 is issued, the extra data for surface 5 will be placed in VEC1. The data is stored in the following format, where the first number in each line refers to the array position:

```
0:      The number of extra data values in the vector
1:      The first extra data value
n:      The nth extra data value
```

See the chapter "Surface Types" for descriptions of the extra data values.

### **GETGLASSDATA**

*Purpose:*

Retrieves the data for any glass in the current catalogs. The data is placed in one of the vector array variables (either VEC1, VEC2, VEC3, or VEC4).

*Syntax:*

GETGLASSDATA vector\_expression, glass\_number

*Discussion:*

The data is stored in the specified VECn array variable. For example, if the command GETGLASSDATA 1, 32 is issued, the data for glass number 32 will be placed in VEC1. The glass number is returned by the GNUM function. The data is stored in the following format, where the first number in each line refers to the array position:

```
0:      The number of data values in the vector
1:      Formula number: 1 for Schott, 2 for Sellmeier 1,
        3 for Herzberger, 4 for Sellmeier 2, 5 for Conrady.
2:      MILNUM
3:      Nd
4:      Vd
```

5: Thermal coefficient of expansion -30 to +70 c  
6: Thermal coefficient of expansion +20 to 300 c  
7: Density in g/cm<sup>3</sup>  
8: Deviation from normal line P gf  
9: Lambda min  
10: Lambda max  
11-16: Constants of dispersion (meaning depends upon formula)  
17-22: Thermal constants of dispersion

## **GETMTF**

### *Purpose:*

Calculates tangential and sagittal MTF, real part, imaginary part, phase, or square wave response data for the currently loaded lens file, and places the data in one of the vector arrays (either VEC1, VEC2, VEC3, or VEC4).

### *Syntax:*

GETMTF freq, wave, field, sampling, vector, type

### *Discussion:*

The freq argument is the desired spatial frequency in line pairs per millimeter. If the frequency is less than zero, or greater than the cutoff frequency, GETMTF returns zero. Wave is an integer corresponding to the wavelength number to use for the calculation. A value of zero indicates a polychromatic calculation. Field must be an integer between 1 and the maximum number of fields. The value indicates which field position to use. Sampling may be 1 (8x8), 2 (16x16), 3 (32x32) or 4 (64x64). The vector argument must be an integer value between 1 and 4, and specifies which vector array the data should be placed in. The type argument refers to the data type: 1 for MTF, 2 for real part, 3 for imaginary part, 4 for phase in radians, 5 for square wave MTF. If any of the arguments fall outside the valid ranges, then the nearest acceptable value is used instead.

The data is returned in one of the vector arrays with the following format: Vector position 0: tangential response; Vector position 1: sagittal response.

### *Example:*

```
! This macro computes the T & S response at 30 lp/mm
! for the currently loaded lens, polychromatic,
! at the maximum defined field,
! and a 16x16 grid density (sampling = 2).
! Data will be placed in vector 1.
! This is all it takes to get the data:
GETMTF 30, 0, NFLD(), 2, 1, 1
PRINT "Tangential response:", vec1(0)
PRINT "Sagittal response :", vec1(1)
```

## **GETSYSTEMDATA**

### *Purpose:*

Retrieves most system specific data, such as effective focal length, working F/#, apodization factors, and other data not associated with any particular surface. The data is placed in one of the vector array variables (either VEC1, VEC2, VEC3, or VEC4).

### *Syntax:*

GETSYSTEMDATA vector\_expression

### *Discussion:*

The data is stored in the specified VECn array variable. For example, if the command GETSYSTEMDATA 1 is issued, the system data will be placed in VEC1. The data is stored in the following format, where the first number in each line refers to the array position:

0: The number of system data values in the vector

- 1: Aperture Value
- 2: Apodization Factor
- 3: Apodization Type (0:none, 1:gaussian, 2:tangent)
- 4: Use Env Data (1 if true, 0 if false)
- 5: Temperature in degrees c (valid only if Use Env Data true)
- 6: Pressure in ATM (valid only if Use Env Data true)
- 7: Effective Focal Length
- 8: Image Space F/#
- 9: Object Space Numerical Aperture
- 10: Working F/#
- 11: Entrance Pupil Diameter
- 12: Entrance Pupil Position
- 13: Exit Pupil Diameter
- 14: Exit Pupil Position
- 15: Paraxial Image Height
- 16: Paraxial Magnification
- 17: Angular Magnification
- 18: Total Track
- 19: Use Ray Aiming (1 if true, 0 if false)
- 20: X Pupil Shift
- 21: Y Pupil Shift
- 22: Z Pupil Shift
- 23: Ray Aiming Tolerance

**GETVARDATA**

*Purpose:*

Retrieves the current number, type, and value of all optimization variables. The data is placed in one of the vector array variables (either VEC1, VEC2, VEC3, or VEC4).

*Syntax:*

GETVARDATA vector

*Discussion:*

The data is stored in the specified VECn array variable. For example, if the command GETVARDATA 1 is issued, the data will be stored in VEC1. The data is stored in the following format, where the first number in each line refers to the array position:

- 0: n, the number of variables
- 1: The type code for the first variable
- 2: ID number for the first variable
- 3: The value of the first variable
- 3\*q-2: The type code for the qth variable
- 3\*q-1: ID number for the qth variable
- 3\*q: The value of the qth variable

The integer q goes from 1 to n, where n is the number of variables. If n is zero, then no valid data is returned. The value of the number in array position zero, n, is always valid. The type codes for variables is as described in the following table.

GETVARDATA TYPE AND ID CODES

Variable type	Type Code	ID number
Curvature	1	surface
Thickness	2	surface
Conic	3	surface



Variable type	Type Code	ID number
Parameter 1-8	4-11	surface
Index Nd	12	surface
Abbe Vd	13	surface
Partial Dispersion $\Delta P_{g,F}$	14	surface
Extra Data 1-100	15-114	surface
TCE	115	surface
Multiconfig Operand i in Config j	300+i	300+j

## **GETZERNIKE**

### *Purpose:*

Calculates Zernike coefficients for the currently loaded lens file, and places them in one of the vector arrays (either VEC1, VEC2, VEC3, or VEC4).

### *Syntax:*

GETZERNIKE maxorder, wave, field, sampling, vector

### *Discussion:*

The maxorder argument is any number between 4 and 36, and corresponds to the highest Zernike term desired. Wave and field are the integer values for the wavelength and field number respectively. The value for sampling determines the size of the grid used to fit the coefficients. Sampling may be 1 (8x8), 2 (16x16), 3 (32x32) or 4 (64x64). The vector argument must be an integer value between 1 and 4, and specifies which vector array the data should be placed in. If any of the arguments fall outside the valid ranges, then the nearest acceptable value is used instead.

The data is returned in one of the vector arrays with the following format: Vector position 1: Peak to valley in waves; Vector position 2: RMS (to zero) in waves; Vector position 3: RMS (to mean) in waves; Vector position 4: RMS (to centroid) in waves; Vector position 5: Variance in waves; Vector position 6: Strehl ratio; Vector position 7: RMS fit error in waves; Vector position 8: Maximum fit error (at any one point) in waves. The remaining vector positions contain the actual Zernike coefficient data. For example, Zernike term number 0 is in vector position 9, Zernike term 1 is in position 10, and so on.

### *Example:*

```
! This macro computes the first 8 Zernike coefficients
! for the currently loaded lens, at wave 1, field 3
! and a 16x16 grid density (sampling = 2). The coefficients
! will be placed in vector 1. First get the data:
GETZERNIKE 8,1,3,2,1
! Now print it out:
FORMAT 16.6
PRINT "Peak to Valley      : ", vec1(1)
PRINT "RMS to zero        : ", vec1(2)
PRINT "RMS to mean        : ", vec1(3)
PRINT "RMS to centroid    : ", vec1(4)
PRINT "Variance           : ", vec1(5)
PRINT "Strehl ratio       : ", vec1(6)
PRINT "RMS Fit Error      : ", vec1(7)
PRINT "Maximum Fit Error  : ", vec1(8)
i = 0
label 1
      FORMAT 2.0
      PRINT "Zernike #", i, " = ",
```

```
        FORMAT 16.6
        PRINT vec1(9+i)
        i = i + 1
if (i < 9) THEN GOTO 1
PRINT "All Done!"
```

## **GLAS**

### *Purpose:*

GLAS is used to set the glass type of a surface.

### *Syntax:*

```
GLAS (surface) = (glass_number)
GLAS (surface) = glass$
```

### *Discussion:*

This keyword requires two expressions, one to specify the surface number and the other to define the number of the glass. If the glass number is -1, the glass type is set to "mirror". If the glass number is 0, then the glass type is blank, or unity index. Otherwise, the glass number must be between 1 and the number returned by MAXG. GLAS also accepts a string variable which contains the name of the desired glass, such as "BK7".

### *Example:*

```
GLAS 3 = MAXG()
A$ = "BK7"
GLAS 4 = GNUM(A$)
GLAS 4 = A$
```

### *Related keywords:*

UPDATE

## **GLENSNAME**

### *Purpose:*

GLENSNAME will place the current lens name at the top left corner of the text box on user defined graphics screens.

### *Syntax:*

```
GLENSNAME
```

### *Discussion:*

GLENSNAME is primarily used for making your graphics look like other ZEMAX graphics.

### *Example:*

See the section GRAPHICS.

## **GOSUB, SUB, RETURN, and END**

### *Purpose:*

These four keywords are used together to define and call subprograms within the ZPL macro file. Each keyword has a special purpose. GOSUB is used to direct the program flow to a defined subroutine. SUB is used to define the subroutine name, as well as indicate the beginning of the subroutine body. RETURN indicates the program execution should continue at the point where the most recent GOSUB call was placed. END indicates that the program should terminate immediately.

### *Syntax:*

See the Example section for sample syntax.

### *Discussion:*

There can be no more than 50 defined subroutines per ZPL macro file. Each subroutine must be terminated by a RETURN statement. More than one return statement may be placed in the body of a subroutine. If subroutines are defined, at least one END statement must be used to indicate the end of the main program body. The main program body must be at the top of the file.

There can be no more than 50 "nesting levels" used in the ZPL macro. For example, if subroutine ABC calls subroutine XYZ, then the nesting level is 2. If subroutine then calls subroutine DEF, the nesting level is 3.

All variables in ZPL are global. Any variables used or defined within a subroutine exist within the main program as well.

### *Example:*

```
x = 1
y = 2
GOSUB add
print "the sum of ", x, " and ", y, " is ", z
END
```

```
SUB add
z = x + y
RETURN
```

## **GOTO**

### *Purpose:*

Normally, each program line is executed in turn. GOTO allows execution to resume at an arbitrary point in the program. GOTO is always used in conjunction with the LABEL command.

### *Syntax:*

```
GOTO label_number
```

### *Discussion:*

There must be a LABEL command with the corresponding label\_number somewhere in the program, or an error will result.

### *Example:*

```
LABEL 1
x=RAND(10)
if x <= 5 THEN GOTO 1
PRINT " X is greater than 5 "
```

## **GRAPHICS**

### *Purpose:*

Creates a standard ZEMAX graphics frame with ruling lines for the plot title.

### *Syntax:*

```
GRAPHICS
GRAPHICS OFF
```

### *Discussion:*

If GRAPHICS is specified alone, then a standard ZEMAX graphics window will be created. All subsequent graphics commands will be sent to this newly created window. GRAPHICS OFF will close any existing open graphics windows, and then display the closed window.

### Example:

```
graphics
xmx = xmax()
xmn = xmin()
ymx = ymax()
ymn = ymin()
xwidth = xmx-xmn
ywidth = ymx-ymn
xleft = xmn + ( .1 * xwidth )
xright = xmn + ( .9 * xwidth )
ytop = ymn + ( .1 * ywidth )
ybott = ymn + ( .7 * ywidth )
line xleft,ytop,xright,ytop
line xright,ytop,xright,ybott
line xright,ybott,xleft,ybott
line xleft,ybott,xleft,ytop
gtitle " the rain in spain falls mainly on the plain"
glensname
gdate
gtext xmx/2,ymx/2,0, " start this text in the center."
gtextcent ymx*.05, " center this text near the top."
gtext xmx*.05,ymx*.75,90, " place me vertically near left edge."
gtext xmx*.15,ymx*.68,30, " orient me at 30 degrees."
graphics off
```

The resulting graphic is shown below.

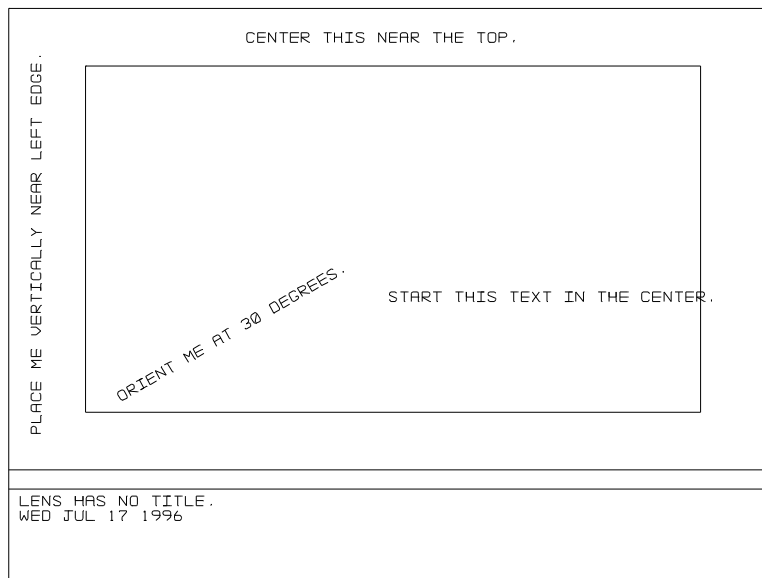


Figure 1.

## GTEXT

### Purpose:

GTEXT is used for labeling graphics plots with user defined text.

### Syntax:

```
GTEXT x, y, angle, user_text
GTEXT x, y, angle, A$
```

*Discussion:*

The coordinates x and y refer to the left edge of where the text string user\_text will appear. "user\_text" may be either a constant string in quotes or a string variable name. Angle specifies how the text is rotated with respect to the graphics frame, and defaults to 0 degrees (horizontal). See also SETTEXTSIZE.

*Example:*

See the section GRAPHICS.

**GTEXTCENT**

*Purpose:*

GTEXT is used for centering labels on graphics plots with user defined text.

*Syntax:*

```
GTEXTCENT y, user_text
```

*Discussion:*

The coordinate y refers to the vertical position of the text string user\_text. See also SETTEXTSIZE.

*Example:*

See the section GRAPHICS.

**GTITLE**

*Purpose:*

GTITLE is similar to GTEXT except the only the text needs to be specified, and the text will appear centered in the title bar on the graphics display. GTITLE is useful for making your ZPL graphics functions look like standard ZEMAX graphics.

*Syntax:*

```
GTITLE user_title
```

*Discussion:*

GTITLE is primarily used for making your graphics look like other ZEMAX graphics.

*Example:*

See the section GRAPHICS.

**IF-THEN-ELSE-ENDIF**

*Purpose:*

IF provides conditional program execution and branching.

*Syntax:*

```
IF (expression)
(statement)
ELSE
(statement)
ENDIF
```

or

```
IF (expression) THEN (statement)
```

### *Discussion:*

The IF-ELSE-ENDIF construction is used for conditional execution of either the group of statements following the IF statement or the statements following the ELSE statement, but not both. The value of expression is considered false if it is zero, otherwise it is considered true. The expression can be any valid ZPL expression, composed of functions, variables, operands, and constants. The IF statement must be paired with an ENDIF, although the ELSE is optional. IF-ENDIF pairs may be nested to any level.

The IF-THEN construction is handy for conditional execution of a single instruction. If the THEN keyword is specified, the IF statement is terminated, and there is no need for an ENDIF. The ELSE keyword is not supported in the IF-THEN construction.

### *Example:*

```
x = 1
y = 2
if (x < y)
    PRINT "x is less than y"
ELSE
    if (x==y) THEN PRINT "x equals y"
    if (x > y) THEN PRINT "x is greater than y"
ENDIF
```

## **INPUT**

### *Purpose:*

INPUT provides a means for prompting the user for numeric or text data when the macro is run.

### *Syntax:*

```
INPUT "Prompt String" , variable
INPUT variable
INPUT "Prompt String" , string_variable$
INPUT string_variable$
```

### *Discussion:*

The variable may be any valid variable name. If the variable name is a string variable, then the input will be interpreted as a literal string; otherwise, as a numeric. The INPUT command will use a "?" prompt if no prompt string is supplied. The prompt is always displayed on the screen, and the input always is accepted from the keyboard only.

### *Example:*

```
INPUT "Enter value for x:", x
PRINT "X = ", x
INPUT "Enter a value for A$:", A$
PRINT A$
```

## **INSERT**

### *Purpose:*

INSERT insets a new surface in the spreadsheet.

### *Syntax:*

```
INSERT n
```

### *Discussion:*

The value n must evaluate to an integer expression. See also DELETE and SURFTYPE.

*Example:*

```
INSERT 5  
INSERT i+2*j
```

**LABEL**

*Purpose:*

LABEL provides a destination for the GOTO command, see the section "GOTO" for details.

*Syntax:*

```
LABEL label_number
```

*Discussion:*

The label\_number must be an integer value, such as 1 or 7. LABEL has no effect on program flow by itself.

**LINE**

*Purpose:*

LINE is the primitive line drawing function for graphical displays.

*Syntax:*

```
LINE oldx, oldy, newx, newy
```

*Discussion:*

LINE will evaluate the four expressions and draw a line connecting the points defined. The coordinates refer to the current graphics frame, and must be contained within the bounds defined by XMIN, YMIN, XMAX, and YMAX. Although only integer pixel values can actually be plotted, LINE will accept real values as arguments and round the coordinates to the nearest integer equivalents. LINE is only valid in graphics mode.

*Example:*

See the section GRAPHICS.

**LOADLENS**

*Purpose:*

Loads a new lens file from disk, replacing any lens file currently in memory.

*Syntax:*

```
LOADLENS "filename" [appendflag]  
LOADLENS file$ [appendflag]
```

*Discussion:*

LOADLENS will load a new lens file from disk. If the filename contains the complete path, such as C:\MY-DIR\MYLENS.ZMX, then the specified file will be loaded. If the path is left off, then the default directory for lenses will be used (see the File menu chapter under Environment).

If the appendflag is zero or absent, then LOADLENS simply loads the file. If the appendflag is greater than zero, then the file is appended to the current lens starting at the surface specified by the value of the appendflag.

*Example:*

```
LOADLENS "COOKE.ZMX"
```

*Related Keywords:*

SAVELENS

## **LOADMERIT**

### *Purpose:*

Loads a merit function file from disk, replacing the merit function in the current lens.

### *Syntax:*

```
LOADMERIT "filename"  
LOADMERIT file$
```

### *Discussion:*

LOADMERIT will load a new merit function from disk. If the filename contains the complete path, such as C:\MYDIR\MYLENS.MF, then the specified file will be loaded. If the path is left off, then the default directory for lenses will be used (see the File menu chapter under Environment).

## **NEXT**

See FOR.

## **NUMFIELD**

### *Purpose:*

Sets the total number of defined fields.

### *Syntax:*

```
NUMFIELD 5
```

## **NUMWAVE**

### *Purpose:*

Sets the total number of defined wavelengths.

### *Syntax:*

```
NUMWAVE 3
```

## **OPEN**

### *Purpose:*

Opens an existing ASCII numeric text file for reading by the READ command.

### *Syntax:*

```
OPEN "filename"  
OPEN A$
```

### *Discussion:*

The filename provided must be a valid file within quotes, or a string variable name containing the file name. See the keywords READ and CLOSE. Always CLOSE a file after all the data has been read.

### *Example:*

```
PRINT "Reading the double-column file TEST.DAT!"  
OPEN "TEST.DAT"  
READ x1, y1  
READ x2, y2  
READ x3, y3  
CLOSE
```



## **OPTIMIZE**

### *Purpose:*

Invokes the optimization algorithm to optimize the current lens with the current merit function.

### *Syntax:*

```
OPTIMIZE  
OPTIMIZE number_of_cycles
```

### *Discussion:*

If no argument is provided, then the optimization runs in "Automatic" mode, stopping when the algorithm detects the process has converged. If an argument is provided, it must evaluate to an integer value between 0 and 99, and the optimization algorithm will run the specified number of cycles.

### *Related Functions:*

MFCN

### *Example:*

```
PRINT "Starting merit function:", MFCN()  
OPTIMIZE  
PRINT "Ending merit function :", MFCN()
```

## **OPTRETURN**

### *Purpose:*

Used to return numerical values back to the optimization algorithm through the use of the ZPLM optimization operand.

### *Syntax:*

```
OPTRETURN (datafield) = (result)
```

### *Discussion:*

OPTRETURN takes two arguments, separated by an equal sign. The (datafield) expression must evaluate to an integer between 0 and 25. The datafield refers to a position in an array where the value of the expression (result) may be stored. The sole purpose of OPTRETURN is to be able to optimize values computed within a ZPL macro.

The optimization operand ZPLM must be used in the merit function to call the ZPL macro and retrieve the value returned by OPTRETURN. See the chapter "Optimization" for details.

### *Example:*

```
x = sqrt(thic(3) + radi(5))  
OPTRETURN j = x+5
```

## **OUTPUT**

### *Purpose:*

Specifies destination for text output. Output is either to the screen, or to a file.

### *Syntax:*

```
OUTPUT SCREEN  
OUTPUT "filename"  
OUTPUT "filename" APPEND  
OUTPUT A$  
OUTPUT A$ APPEND
```

### *Discussion:*

If OUTPUT SCREEN is specified alone, then all subsequently executed PRINT statements will be directed to the screen. If a valid filename is provided, then subsequent PRINT statements will output to the filename specified. To close the file created earlier, use OUTPUT SCREEN which will direct subsequent PRINT outputs to the screen. SHOWFILE will close the file and send it to the text viewer program for screen display. PRINTFILE will close the file and print it on the currently defined printer.

If the keyword APPEND follows the file name, then subsequent output will be appended to the file. Otherwise, the contents of the file will be overwritten.

### *Example:*

```
OUTPUT "x.txt"  
PRINT "This will not appear on the screen, but in the file x.txt."  
OUTPUT SCREEN  
PRINT "This will appear on the screen."  
OUTPUT "x.txt" APPEND  
PRINT "This will appear after the first line in the file x.txt."
```

### *Related Keywords:*

CLOSE, OPEN, SHOWFILE, PRINTFILE

## **PARn**

### *Purpose:*

PARn is used to set the  $n^{\text{th}}$  parameter of a surface. The parameter values are defined in chapter 2, "Using ZEMAX". Each surface model uses from zero to eight of the parameter values.

### *Syntax:*

```
PAR1 (surface) = (new_value)  
PAR8 (surface) = (new_value)
```

### *Discussion:*

This keyword requires two expressions, one to specify the surface number and the other to define the new value. The expression (surface) is evaluated and then rounded down to an integer to yield the surface number. If the surface number is less than zero or greater than the number of surfaces, the command is ignored. Otherwise, the expression (new\_value) is evaluated and assigned. Note that PARn refers to eight different commands: PAR1, PAR2, ..., PAR8.

### *Example:*

```
PAR1 (1) = 250
```

### *Related keywords:*

UPDATE

## **PARAXIAL**

### *Purpose:*

Used to control whether ray tracing is done with paraxial or real rays.

### *Syntax:*

```
PARAXIAL ON  
PARAXIAL OFF
```

### *Discussion:*

The current paraxial mode can be established by a call to the function PMOD, which returns 0 if paraxial mode is off, 1 if it is on. This feature is used for switching between real and paraxial ray traces. Certain calculations,

such as measuring distortion and computing first-order properties such as effective focal length, require the tracing of paraxial rays.

*Example:*

```
mode = PMODE()  
IF mode THEN PRINT "Paraxial mode is on!"  
IF !mode THEN PRINT "Paraxial mode is not on!"  
PARAXIAL ON  
PRINT "Now paraxial mode is on!"  
PRINT "Restoring original mode..."  
if !mode THEN PARAXIAL OFF
```

**PAUSE**

*Purpose:*

Halts program execution while displaying a status message.

*Syntax:*

```
PAUSE  
PAUSE x  
PAUSE "Ready to continue..."  
PAUSE x+sqrt(5)
```

*Discussion:*

This feature is automatically called when closing a graphics screen.

**PIXEL**

*Purpose:*

Turns on a single pixel on the current graphics screen.

*Syntax:*

```
PIXEL xcoord, ycoord
```

*Discussion:*

This feature is useful for making spot diagrams.

**POLDEFINE**

*Purpose:*

Defines the input polarization state for subsequent POLTRACE calls.

*Syntax:*

```
POLDEFINE Ex, Ey, PhaX, PhaY
```

*Discussion:*

The POLDEFINE keyword is used to define the input polarization state for subsequent polarization ray tracing. POLDEFINE requires the normalized Ex and Ey electric field magnitudes, as well as the X and Y phase angles in degrees. The default values are 0, 1, 0, and 0, respectively. Once the polarization state is defined, it remains the same until changed.

*Example:*

```
POLDEFINE 2.0 2.0 45.0 -66.0
```

*Related keywords:*

POLTRACE

## **POLTRACE**

### *Purpose:*

Calls the ZEMAX polarization ray tracing routines to trace a particular ray through the current system.

### *Syntax:*

```
POLTRACE (hx), (hy), (px), (py), (wavelength), (vec), (surf)
```

### *Discussion:*

The expressions hx and hy must evaluate to values between -1 and 1, and represent the normalized object coordinates. The pupil coordinates are specified by the expressions px and py, which also must be between -1 and 1. For more information about normalized coordinates, see the chapter "Using ZEMAX" under "Using normalized field and pupil coordinates". The wavelength expression must evaluate to an integer between 1 and the maximum number of defined wavelengths. The vec expression must evaluate to a number between 1 and 4, inclusive. The surf expression must evaluate to an integer between 1 and the number of surfaces, inclusive.

The input polarization state of the ray is defined by the POLDEFINE keyword.

Once the ray is traced, the polarization data for the ray is placed in the vector variable specified by the vec expression. For example, if the command "POLTRACE hx, hy, px, py, w, 2, n" is issued, the data will be stored in VEC2. The data is stored in the following format, where the first number in each line refers to the array position:

```
0:          n, the number of data entries in the vector
1:          The ray intensity after the surface
2:          E-Field X component, real
3:          E-Field Y component, real
4:          E-Field Z component, real
5:          E-Field X component, imaginary
6:          E-Field Y component, imaginary
7:          E-Field Z component, imaginary
8:          S-Polarization amplitude reflection, real
9:          S-Polarization amplitude reflection, imaginary
10:         S-Polarization amplitude transmission, real
11:         S-Polarization amplitude transmission, imaginary
12:         P-Polarization amplitude reflection, real
13:         P-Polarization amplitude reflection, imaginary
14:         P-Polarization amplitude transmission, real
15:         P-Polarization amplitude transmission, imaginary
```

If the value in array position 0 is 0, then an error occurred and the polarization data is invalid. This may occur if the specified ray cannot be traced. See the RAYTRACE command to extract extended error information.

### *Example:*

```
POLDEFINE 0, 1, 0, 0
POLTRACE 0, 1, 0, 0, pwav(), 1, nsur()
PRINT "Transmission of chief ray at primary wavelength is ", vec1(1)
```

### *Related keywords:*

POLDEFINE, RAYTRACE

## **PRINT**

### *Purpose:*

Print is used to output constant text and variable data to either the screen or a file, depending upon the current status of the keyword OUTPUT.

### *Syntax:*

```
PRINT
```

```
PRINT X
PRINT "The value of x is ", x
PRINT " x = ", x, " x + y = ", x + y
```

*Discussion:*

PRINT alone will print a blank line. PRINT with a list of text arguments and expressions will print each text string (enclosed in double quotes) and the numeric value of each expression. PRINT uses the numerical output format specified by FORMAT. If the last item in the list is followed by a comma, PRINT will not end the line with a carriage return.

*Example:*

```
X=3
PRINT "X equals ",x
```

## **PRINTFILE**

*Purpose:*

Prints a text file.

*Syntax:*

```
PRINTFILE "filename"
PRINTFILE NAME$
```

*Discussion:*

The filename must be a valid filename within quotes or the name of a string variable which contains a valid file name. The file must be an ASCII file (as would be created by OUTPUT and PRINT statements in ZPL) and must be in the current directory. PRINTFILE also closes the file if no CLOSE statement has been executed.

*Example:*

```
OUTPUT "test.txt"
PRINT "Print this to the printer."
PRINTFILE "test.txt"
```

*Related Keywords:*

OPEN, OUTPUT, CLOSE, PRINT, PRINTFILE

## **PRINTWINDOW**

*Purpose:*

Prints any open graphic or text window.

*Syntax:*

```
PRINTWINDOW winnum
```

*Discussion:*

The winnum value may be either an integer or an expression that evaluates to an integer. The integer winnum corresponds to the window number that should be printed. ZEMAX numbers windows sequentially as they are opened, starting with 1. Any closed windows are deleted from the window list, without renumbering the windows which remain. Any windows opened after another window has been closed will use the lowest window number available.

*Example:*

```
PRINTWINDOW 5
```

## **RADI**

### *Purpose:*

RADI is used to set the radius of curvature of a surface.

### *Syntax:*

```
RADI (surface) = (new_value)
```

### *Discussion:*

This keyword requires two expressions, one to specify the surface number and the other to define the new value. The expression (surface) is evaluated and then rounded down to an integer to yield the surface number. If the surface number is less than zero or greater than the number of surfaces, the command is ignored. Otherwise, the expression (new\_value) is evaluated and assigned.

### *Example:*

```
RADI i+m = -1200
```

### *Related keywords:*

UPDATE

## **RAYTRACE**

### *Purpose:*

Calls the ZEMAX ray tracing routines to trace a particular ray through the current system.

### *Syntax:*

```
RAYTRACE (hx), (hy), (px), (py), (wavelength)
```

### *Discussion:*

The expressions hx and hy must evaluate to values between -1 and 1, and represent the normalized object coordinates. The pupil coordinates are specified by the expressions px and py, which also must be between -1 and 1. For more information about normalized coordinates, see the chapter "Using ZEMAX" under "Using normalized field and pupil coordinates". The wavelength expression is optional, defaulting to the primary wavelength, but if supplied must evaluate to an integer between 1 and the maximum number of defined wavelengths.

Once the ray is traced, the ray intercept coordinates and direction cosines may be determined using the ZPL functions RAYX, RAYY, RAYZ, RAYL, RAYM, and RAYN. If an error occurred during ray tracing, the function RAYE (for RAY Error) will return a value other than zero. If RAYE is negative, it indicates that total internal reflection occurred at the surface whose number is the absolute value of the value returned. If RAYE is greater than zero, then the ray missed the surface number returned. Checking RAYE is optional, however, the RAYX, RAYY, ... functions may return invalid data if RAYE is not zero. The functions RANX, RANY, and RANZ return the intercept surface normal direction cosines, and OPDC returns the optical path difference for the ray. The function RAYV returns the surface number at which the ray was vignetted, or it returns zero if the ray was not vignetted. Values returned for surfaces past the surface of vignetting may not be accurate.

### *Example:*

```
PRINT "Tracing the marginal ray at primary wavelength!"
n=NSUR()
RAYTRACE 0,0,0,1
y = RAYY(n)
PRINT "The ray intercept is ", y
PRINT "Tracing the chief ray at maximum wavelength!"
RAYTRACE 0,1,0,0,NWAV()
y = RAYY(n)
PRINT "The ray intercept is ", y
```

## **READ**

### *Purpose:*

Reads data from an existing ASCII numeric text file opened for reading by the OPEN command.

### *Syntax:*

```
READ x
READ x, y
READ x,y,z,a,b,c,q
```

### *Discussion:*

The ASCII file must be already open, see the keyword OPEN for details. Each READ command reads a single line from the file. The first valid data field from this line is placed in the variable first listed. The data from the second field is placed in the second variable listed, if any. Therefore, the number of variables listed in the read statement should match the number of columns in the text file. Numeric data in the file should be delimited by spaces. The data may be in free-form, and is internally promoted to double precision. A maximum of 199 characters can be read in on any single line. The variables listed must be valid ZPL variable names.

Always CLOSE a file after all the data has been read. See the function EOFF.

### *Example:*

```
PRINT "Reading the double-column file TEST.DAT!"
OPEN TEST.DAT
READ x1, y1
READ x2, y2
READ x3, y3
CLOSE
```

## **READSTRING**

### *Purpose:*

Reads data from an existing ASCII text file opened for reading by the OPEN command.

### *Syntax:*

```
READSTRING A$
```

### *Discussion:*

The ASCII file must be already open, see the keyword OPEN for details. Each READ command reads a single line from the file. The entire line read is placed in the variable listed. The variable listed must be a valid ZPL string variable name, although it does not need to be previously referenced. Always CLOSE a file after all the data has been read. See the function EOFF.

### *Example:*

```
PRINT "Reading the contents of file TEST.DAT!"
OPEN TEST.DAT
READSTRING A$
PRINT A$
CLOSE
```

## **REM, !**

### *Purpose:*

REM is used to indicate the rest of the line is a remark.

### *Syntax:*

```
REM text
! text
```

### *Discussion:*

The exclamation symbol may also be used to indicate a remark. Both the REM command and the "!" symbol are only recognized as remark indicators if they appear at the very beginning of the line, prior to any spaces, tabs, or other characters. Any other use will cause a syntax error at run time.

### *Example:*

```
REM any text can be placed after the REM command.  
! any text can also be placed  
! after the exclamation symbol.
```

## **RETURN**

See GOSUB.

## **SAVELENS**

### *Purpose:*

Saves the current lens file.

### *Syntax:*

```
SAVELENS ["filename"]  
SAVELENS NEW$
```

### *Discussion:*

SAVELENS will save the current lens file to disk. The name of the current lens in memory will also be changed. If the file name is absent, then the lens data is stored in the current file name.

### *Example:*

```
SAVELENS  
SAVELENS "NEWCOPY.ZMX"  
SAVELENS NEW$
```

### *Related Keywords:*

LOADLENS

## **SDIA**

### *Purpose:*

SDIA is used to set the semi-diameter of a surface.

### *Syntax:*

```
SDIA (surface) = (new_value)
```

### *Discussion:*

This keyword requires two expressions, one to specify the surface number and the other to define the new value. The expression (surface) is evaluated and then rounded down to an integer to yield the surface number. If the surface number is less than zero or greater than the number of surfaces, the command is ignored. Otherwise, the expression (new\_value) is evaluated and assigned.

If new\_value evaluates to a positive number, then the semi-diameter status will be changed to "Fixed", no matter what the initial status was. If new\_value is negative, then the status is set to "Automatic" which will be updated the next time "UPDATE" is called.

### *Example:*

```
SDIA 5 = SDIA(4)
```



*Related keywords:*

UPDATE

### **SETAIM**

*Purpose:*

Sets the state of the ray aiming function.

*Syntax:*

```
SETAIM state
```

*Discussion:*

This keyword requires one numeric expression that must evaluate to an integer between 0 and 2. The state is a code which is 0 for no ray aiming (off), 1 for paraxial reference, and 2 for real reference.

*Example:*

```
SETAIM 1
```

### **SETCONFIG**

*Purpose:*

Sets the current configuration for multi-configuration (zoom) systems.

*Syntax:*

```
SETCONFIG (confignumber)
```

*Discussion:*

This keyword requires one numeric expression that must evaluate to an integer between 1 and the maximum number of configurations. The expression is evaluated and then rounded down to an integer to yield the configuration number.

*Example:*

```
SETCONFIG 4
```

*Related Functions:*

CONF, NCON

### **SETTEXTSIZE**

*Purpose:*

Changes the size of the characters drawn by the GTEXT command.

*Syntax:*

```
SETTEXTSIZE xsize ysize
```

*Discussion:*

The arguments refer to the fraction of the graphic screen width that each character represents. For example, the default text size is 70 40. This means each character is 1/70 of the graphic screen width, and 1/40 of the screen height. An argument of zero restores the text size to the default.

*Example:*

```
! Make text twice default size  
SETTEXTSIZE 35 20  
! Restore text size to default  
SETTEXTSIZE
```

## **SETTITLE**

### *Purpose:*

Sets the title of the lens that normally appears on all plots.

### *Syntax:*

```
SETTITLE A$  
SETTITLE "Here is the lens title"
```

## **SETUNITS**

### *Purpose:*

Sets the current lens units.

### *Syntax:*

```
SETUNITS code
```

### *Discussion:*

This keyword requires one numeric expression that must evaluate to an integer between 0 and 3. The code is 0 for mm, 1 for cm, 2 for inches, and 3 for meters. This function does not scale or convert the lens data in any way, it only changes how the lens prescription data is interpreted.

### *Example:*

```
SETUNITS 0
```

## **SETVAR**

### *Purpose:*

Changes the state of variables for optimization.

### *Syntax:*

```
SETVAR (surface_expression), VARCODE, (status_expression)
```

### *Discussion:*

The surface\_expression must evaluate to an integer argument between 0 and the maximum number of surfaces, or an error will result. The surface\_expression determines which surface will be made variable. The VARCODE must be one of the following ASCII mnemonics:

R for radius of curvature

T for thickness

C for conic

Pn for parameter n, where n must be one digit.

Enn for extra data value nn, where nn must be two digits.

If the status\_expression evaluates to 0, then the variable status is removed. Otherwise, the value is made variable.

### *Examples:*

```
SETVAR j+3, R, 1
```

```
SETVAR 5, P6, 0
```

```
SETVAR surfk+2, E06, status
```

## **SETVECSIZE**

### *Purpose:*

Changes the maximum size of the VEC1, VEC2, VEC3, and VEC4 arrays.

### *Syntax:*

```
SETVECSIZE (expression)
```

### *Discussion:*

The expression must evaluate to an integer argument between 1 and 100,000. All four vector variables are always the same size.

## **SETVIG**

### *Purpose:*

Sets the vignetting factors for the lens.

### *Syntax:*

```
SETVIG
```

### *Discussion:*

See the Chapter "Conventions and Definitions" for a description of vignetting factors.

## **SHOWFILE**

### *Purpose:*

Displays a text file to the screen using the ZEMAX file viewer.

### *Syntax:*

```
SHOWFILE "filename" saveflag  
SHOWFILE NAME$ saveflag
```

### *Discussion:*

The filename must be a valid filename within quotes or a string variable containing a valid file name. The file must be an ASCII file (as would be created by OUTPUT and PRINT statements in ZPL) and must be in the current directory. Once the file is displayed, it may be scrolled up and down and printed like any other text file. The ability to scroll and print the data is the primary advantage of using OUTPUT and SHOWFILE instead of simple PRINT statements. SHOWFILE also closes the file if no CLOSE statement has been executed. If the saveflag is zero or omitted, then the file is erased when the window is closed. If saveflag is any value other than zero, then the file remains even after the window is closed.

### *Example:*

```
OUTPUT "test.txt"  
PRINT "Print this to a file."  
SHOWFILE "test.txt"
```

### *Related Keywords:*

OPEN, OUTPUT, CLOSE, PRINT, PRINTFILE

## **SOLVETYPE**

### *Purpose:*

Changes the solve status on a given surface and value. Only some solve types are supported; contact FSI for information on setting other types of solves.

### *Syntax:*

```
SOLVETYPE (surf_expression), CODE, (arg1), (arg2)
```

*Discussion:*

The surf\_expression must evaluate to an integer argument between 0 and the maximum number of surfaces, or an error will result. The CODE must be an ASCII mnemonic as listed in the following table. The arg1 and arg2 expressions evaluate to the first and second solve parameters as specified in the "Solves" chapter.

CODES FOR SOLVETYPE KEYWORD

Solve Type	CODE
Curvature Pickup	CP
Thickness Pickup	TP
Thickness Marginal Ray Height	TM
Glass Pickup	GP
Conic Pickup	KP

*Example:*

```
! The following line will add a glass pickup solve
! on surface 7, picking up from surface 5:
SOLVETYPE 7, GP, 5
! Add a thickness pickup with a scale factor of -1:
SOLVETYPE 7, TP, 5, -1
```

**STOPSURF**

*Purpose:*

STOPSURF sets the current stop surface location by number.

*Syntax:*

```
STOPSURF (surface)
```

*Discussion:*

This keyword requires one expression which evaluates to an integer between 1 and the number of surfaces minus one.

*Example:*

```
STOPSURF n+2
```

*Related keywords:*

UPDATE

**SUB**

See GOSUB.

**SURFTYPE**

*Purpose:*

Changes the surface type.

*Syntax:*

```
SURFTYPE (surf_expression), CODE
```

*Discussion:*

The surf\_expression must evaluate to an integer between 0 and the number of surfaces. The CODE is an integer value which indicates the surface type. The integer codes may be determined at run time by the SCOD function.

*Example:*

```
SURFTYPE j+1, SCOD("STANDARD")
B$ = "EVENASPH"
SURFTYPE 5, SCOD(B$)
```

**THIC**

*Purpose:*

THIC is used to set the thickness of a surface.

*Syntax:*

```
THIC (surface) = (new_value)
```

*Discussion:*

This keyword requires two expressions, one to specify the surface number and the other to define the new value. The expression (surface) is evaluated and then rounded down to an integer to yield the surface number. If the surface number is less than zero or greater than the number of surfaces, the command is ignored. Otherwise, the expression (new\_value) is evaluated and assigned.

*Example:*

```
THIC i+m = THIC(i+n)-1
```

*Related keywords:*

UPDATE

**TIMER**

*Purpose:*

Resets the internal clock. This feature is used in conjunction with the ZPL function ETIM() for measuring the elapsed time since the last TIMER command.

*Syntax:*

```
TIMER
```

*Discussion:*

TIMER and ETIM() are used primarily for testing execution efficiency of the ZPL interpreter and various program architectures.

*Example:*

```
i=0
TIMER
LABEL 1
x=RAND(1000)
i=i+1
if i 10000 THEN GOTO 1
FORMAT .1
PRINT "Elapsed time:", ETIM(), "Seconds"
```

## **UPDATE**

### *Purpose:*

Updates pupil positions, index data, paraxial constants, semi-diameters, maximum field normalization values, and solves. The UPDATE keyword MUST be used before tracing or evaluating a system if the prescription data (such as radii and thicknesses) have been altered since the last UPDATE. If the UPDATE command is followed by any word, such as "ALL", then all open windows will be updated as well.

### *Syntax:*

```
UPDATE
UPDATE ALL
```

### *Example:*

```
THIC 5 = THIC (5)-1
UPDATE
```

## **VEC1, VEC2, VEC3, VEC4**

### *Purpose:*

These keywords are used to set the vector variables VEC1, VEC2, VEC3, and VEC4. Each vector can store an array of double-precision floating point numbers.

### *Syntax:*

```
VEC1 (array_subscript) = (new_value)
VEC2 (array_subscript) = (new_value)
```

### *Discussion:*

VEC1..4 are used to store any user data in an array. The array\_subscript value can be any expression which is rounded down to an integer. The resulting integer expression must be between 0 and the current maximum vector size, which is initially 100, or an error will be reported. The ZPL functions VEC1..4 can be used to extract the data.

To change the default size to a value different from 100, use the SETVECSIZE keyword.

### *Example:*

```
i = 0
LABEL 1
i = i + 1
VEC1(i)=i
IF i < 10 THEN GOTO 1
j = 0
LABEL 2
j = j + 1
VEC2(j) = VEC1(j) * VEC1(j)
IF j < 10 THEN GOTO 2
i = 0
LABEL 3
i = i + 1
PRINT "x = ", VEC1(i), " x*x = ", VEC2(i)
IF i < 10 THEN GOTO 3
PRINT
PRINT "All done!"
```

## **WAVL, WWGT**

### *Purpose:*

WAVL and WWGT are used to change the wavelength value and weight.

### *Syntax:*

```
WAVL (wavelength number) = (new_value)
WWGT (wavelength) = (new_value)
```

### *Discussion:*

This keyword requires two expressions, one to specify the wavelength number and the other to define the new value. The expression (wavelength number) is evaluated and then rounded down to an integer to yield the wavelength number. If the wavelength number is less than 1 or greater than the number of wavelengths, the command is ignored. Otherwise, the expression (new\_value) is evaluated and assigned.

### *Example:*

```
WAVL 1 = 0.486
WAVL 2 = 0.587
WWGT 1 = 1.0
WWGT 2 = 0.65
```

### *Related functions:*

WAVL, WWGT

## **Example 1**

---

Suppose you needed a program to print out the image plane intercept coordinates of the chief ray at every defined field angle. This is more commonly required than you may think, for example, ZEMAX spot diagrams are all plotted relative to the chief ray coordinates, and so the intercept coordinates are needed to properly offset the data. The same is true for transverse ray aberrations; they are referenced to the chief ray.

The best way to learn ZPL is to study previously written programs. Look closely at the following program and try to follow the program logic.

```
nfield = NFLD()
maxfield = MAXF()
n = NSUR()
FOR i=1,nfield,1
    hx = FLDX(i)/maxfield
    hy = FLDY(i)/maxfield
    PRINT "Field number ", i
    RAYTRACE hx,hy,0,0,PWAV()
    PRINT "X-field angle : ",FLDX(i)," Y-field angle : ", FLDY(i)
    PRINT "X-chief ray   : ",RAYX(n), " Y-chief ray   : ", RAYY(n)
    PRINT
NEXT
PRINT "All Done!"
```

The first line of the program calls the NFLD() function which returns the number of defined field angles, and assigns it to the variable "numfield". The second line calls MAXF(), which returns the maximum radial field angle in degrees, and stores the value in the variable "maxfield". The number of surfaces is then stored in the variable "n" by calling the function NSUR().

The FOR loop is then defined with i being the counter for the field position, starting at 1, with a maximum value of nfield, and an increment of 1 on each loop. The two program lines which define "hx" and "hy" use the functions FLDX() and FLDY(), which return the x- and y-angles for the current field position number "i". The chief ray is then traced by the keyword RAYTRACE. Note the chief ray intersects the center of the pupil, which is why the two pupil coordinates are both zero. The PWAV() function returns the primary wavelength number, which is usually used for the chief ray.

The various PRINT statements will output the chief ray coordinates on the image plane to the screen. If you like, try typing the program into a file ending in ZPL (for example, CHIEF.ZPL) and then run it from within ZEMAX to test it. The coordinates reported should be the same as those listed on the spot diagram when the chief ray is the reference point.

## Example 2

---

Here is an example ZPL program to estimate the RMS spot size (on axis) of the current optical system. The program traces many random rays through the system, and records the radial offset from the primary wavelength chief ray. The current wavelength weighting is applied to estimate the RMS spot size.

```
PRINT "Primary wavelength is number ",
FORMAT .0
PRINT PWAV(),
FORMAT .4
PRINT " which is ", WAVL(PWAV()), " microns."
PRINT "Estimating RMS spot size for each wavelength."

! How many random rays to trace to make estimate?
n = 100

! Initialize the timer
TIMER

! Store the number of surfaces for later use
ns = NSUR()

! Start at wavelength 1
weightsum=0
wvrms = 0
FOR w = 1, NWAU(), 1
    rms = 0
    FOR i = 1, n, 1
        hx=0
        hy=0
        angle = 6.283185 * RAND(1)
        ! SQRT yields uniform distribution in pupil
        radius = SQRT(RAND(1))
        px = radius * COSI(angle)
        py = radius * SINE(angle)
        RAYTRACE hx,hy,px,py,w
        x=RAYX(ns)
        y=RAYY(ns)
        rms = rms + (x*x) + (y*y)
    NEXT
    rms = SQRT(rms/n)
    wvrms = wvrms + ( WWGT (w) * rms )
    weightsum = weightsum + WWGT(w)
    FORMAT .4
    PRINT "RMS spot size for ", WAVL(w),
    FORMAT .6
    PRINT " is ", rms
NEXT
wvrms = wvrms / weightsum
PRINT "Wavelength weighted rms is ", wvrms
FORMAT .2
t = ETIM()
PRINT "Elapsed time was ",t," seconds."
```

Note the use of the trailing commas in the first two PRINT statements. These allow the data printed from the first three PRINT statements to appear on the same line. The intervening FORMAT statements changes the way the numerical output is printed, even on the same line. The "!" character is used to indicate a comment, which is ignored when the program is run.



This page intentionally left blank.

## Introduction

---

ZEMAX is capable of modeling many types of optical systems and components. However, it is sometimes difficult to see how to set up certain kinds of optical systems. There are optical systems that the program does not appear to support, but that can in fact be modeled if certain tricks are employed. This chapter provides a discussion of some of these tricks and techniques. They are presented in no particular order.

## Simulating laser diode sources

---

ZEMAX models sources on the object plane as a collection of points. Each point radiates a "cone" of light which fills the entrance pupil. It therefore appears impossible to model a source with significant astigmatism, such as a laser diode. Diodes often have different divergence angles in the X- and Y- directions, and have different apparent source points in the XZ and YZ planes. Note that a diode that radiates from many points along a line with different divergence in the XZ and YZ planes can be thought of in this fashion.

It is however possible to model a diode by using a point source on axis, and then image it with a paraxial cylinder lens to a dummy surface, which is subsequently used as the diode source. The paraxial XY surface can act like a paraxial cylinder lens, with the flexibility to specify different focal powers in the X and Y directions. We will exploit this feature to introduce the proper astigmatism.

Suppose the diode has a half angular divergence in the YZ plane of  $\theta_y$ , a half angular divergence in the XZ plane of  $\theta_x$ , and the rays in the XZ plane appear to come from a point a distance  $t$  behind (away from the lens) the point at which the YZ rays appear to diverge from. The trick is to re-image a point source to a dummy plane that will act like the astigmatic diode source with these properties. It is assumed that the divergence angles are measured in air with unity index. The model we will develop is sufficiently general to model diodes which have a non-unity index internally.

To accomplish this re-imaging, we will use a single conventional point source on the object surface, which is surface 0. Surface 1 will be a paraxial XY surface (note this is a different surface type than a paraxial lens surface), and surface 2 will be a standard surface. Surface 1 should also be the stop surface. Surface 2 will act as the astigmatic diode source surface, so all of the optics that follow the diode source should be placed after, and positioned with respect to, surface 2 instead of the usual surface 0. Let  $t_1$  be the thickness of the object surface (the distance from the point source to the paraxial XY surface) and let  $t_2$  be the thickness of surface 1 (the distance from the paraxial XY surface to surface 2).

The "Glass" type for both surface 0 and surface 1 should be set the same. If the aberrations introduced by the index of refraction change at the face of the diode are unimportant, then the internal index of the diode may be left at unity, and no entry need be made in the glass column. If you need to add a special high index material to the glass catalog, see the chapter "Using glass catalogs". Be sure the same material name is entered on both surfaces 0 and 1. You can verify the index is correct using the surface data feature, described in the Chapter "Reports Menu".

Use the system aperture type selection "Object Space N.A." on the System, General dialog box, and set the system aperture value to the divergence in the XZ plane. For example, if the half angular divergence in the XZ plane is 10 degrees, set the N.A. to 0.17365 (the sine of the half-angle). This is the correct value to use even if the diode itself will have non-unity index.

Given  $\theta_x$ ,  $\theta_y$ ,  $t$ , and the index of the diode medium  $n$ ; the values  $t_1$ ,  $t_2$ , and  $\phi_y$  may all be computed. Define the magnification in the YZ plane to be

$$M = \frac{\tan\theta'_x}{\tan\theta'_y},$$

where

$$n \sin\theta'_y = \sin\theta_y, \quad n \sin\theta'_x = \sin\theta_x.$$

Then

$$t = t_1 + t_2 ,$$

$$M = \frac{t_2}{t_1} ,$$

$$t_1 = \frac{t}{M + 1} ,$$

and

$$t_2 = \frac{M t}{M + 1} .$$

The focal power required in the YZ plane is given by

$$\phi_y = \frac{n}{t_1} + \frac{n}{t_2} = \frac{n(M+1)^2}{M t} .$$

Parameter 1 on the paraxial XY surface is the X power, and it should be set to zero. Parameter 2 is the Y power, and it should be set to  $\phi_y$ .

Example: Suppose the diode medium has index unity, the XZ half angle divergence is 10 degrees, the YZ half angle divergence is 30 degrees, and the effective source location in the XZ plane is 0.70 millimeters behind the YZ source point. From the preceding equations, N.A. = 0.17365,  $M = .30541$ ,  $t_1 = 0.53623$  mm,  $t_2 = 0.16377$  mm, and  $\phi_y = 7.97104$  inverse mm.

An example lens with these source parameters can be found in the file DIODE.ZMX, and a drawing of it is shown in Figure A-1. The drawing has been rotated 45 degrees around the Z axis to show both the XZ and YZ plane rays. Note the source forms a line focus on the dummy source plane (surface 2). The length of the source line is given by

$$L = 2 t \tan\theta'_x$$

or 0.2469 mm for the example given. Note that  $t$  can be calculated from the previous expression if the divergence angle and source length is known.

If the index of the diode was 1.5, then the results are N.A. = 0.17365,  $M = .32965$ ,  $t_1 = 0.52645$  mm,  $t_2 = 0.17354$  mm, and  $\phi_y = 11.4925$  inverse mm.

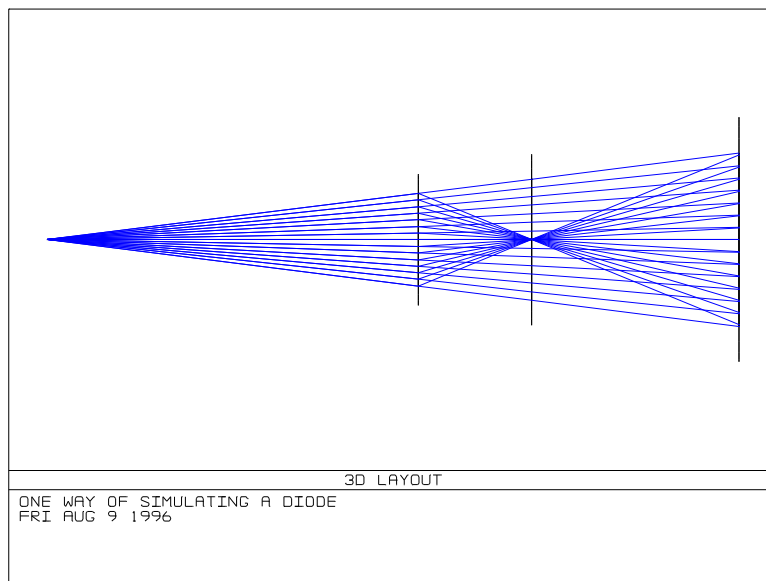


Figure 1

## Simulating an elliptical source

---

An elliptical source is similar to a laser diode source in that the divergence angles are different in the XZ and YZ planes. However, the apparent source points are the same in either plane, that is, there is no astigmatism. By far the easiest way to set up such a source is to use the vignetting factors available on the field dialog box. If the VCX (vignetting compression x) is greater than 0, then the beam will be narrower in x than in y. For a beam that is narrower in y than x, use a VCY value greater than 0.

See the chapter "Conventions and Definitions" for a discussion of vignetting factors.

## Modeling axicons

---

An axicon surface is like a cone, with the tip of the cone at the vertex of the surface. Axicons are rotationally symmetric about the z axis, and are described by a single parameter  $\theta$ . The angle  $\theta$  is measured between the plane normal to the z axis at the vertex of the cone and the axicon surface. For  $\theta = 0.0$ , the axicon is identical to a plane. Axicons are used to create a "ring" focus in certain types of laser systems, although they have other uses. The surface sag of an axicon is given by

$$z = \tan\theta r,$$

where z is the surface sag and r is the radial coordinate in lens units.

There are two ways to model an axicon: 1) use the "Odd Asphere" surface model, or 2) use the "Standard" surface model. Using the standard surface model is preferred for several reasons. The primary reason is that ray tracing is much faster and generally more robust when using the standard surface. This is because a closed form solution exists for ray tracing to a standard surface, while iteration is required when using the odd asphere. Another reason for using the standard surface is the avoidance of a "cusp" at the surface vertex which causes a singularity in the ray tracing equations (ZEMAX traps this special case, but it should still be avoided).

The drawback to using the standard surface is precision; it is only an approximation to a true axicon. The deviation is only significant near the cusp of the axicon, which is a region rarely used in real axicon applications anyway. The error can be made arbitrarily small by the technique discussed in the following paragraphs.

The standard surface behaves like an axicon when the radius of curvature is set to a very small value (the curvature is then a large value) and the conic constant is less than -1.0. This can be seen by taking the limit of the standard surface sag as the curvature goes to infinity:

$$\lim_{c \rightarrow \infty} \frac{cr^2}{1 + \sqrt{1 - (1+k)c^2r^2}} = \frac{1}{\sqrt{-(1+k)}} r.$$

Then the conic constant should be set according to

$$k = -\left(\frac{1}{\tan^2\theta} + 1\right).$$

The exact value of the radius of curvature does not matter, but it should be several times less than the smallest radial aperture you intend to use. For example, if the axicon has a diameter of 100 mm, the axial portion to be excluded is 5 mm in diameter, and the cone angle is 10 degrees, use  $k = -33.16$  and a radius of curvature on the order of .1 mm or less. A smaller value will also be satisfactory, but do not use zero.

The other approach is to use the odd asphere surface model. Simply set the radius of curvature to infinity, and the parameter 1 value to the tangent of the cone angle. See the chapter "Surface Types" for details on the odd asphere surface. Although it sounds simpler, this approach should not be used unless the standard surface model proves unsatisfactory.

## Eliminating variables in zoom systems

---

Imagine a simple zoom lens with 3 elements. The first and third elements are fixed in position with respect to each other, but the center element can shift back and forth to change the overall focal length. The distance between the first and third element must be the same in all zoom positions, but the actual distance is a free variable which can be optimized. Suppose there are five zoom positions.

Use the multi-configuration capability to define five configurations. Let the surface numbers of the first element be 1 and 2; the second element is 3 and 4, and the third element is 5 and 6. All that is left to do is to define the variables, multi-configuration operands, and constraints so that the distance between the first and third elements is constant.

The easy, but inefficient way of setting this problem up is:

Use the THIC multi-configuration operand twice; once for surface 2 and again for surface 4. Make the values of the THIC 2 and THIC 4 operands variable in each of the five configurations. To constrain the distance between the first and third elements to be constant, use the ZTHI optimization operand. There are a total of ten variables for optimization.

The more efficient method is only slightly harder to set up:

Use the THIC multi-configuration operand once for surface 2 only. Make the values of the THIC 2 operand variable in each configuration. Instead of making the values of surface 4 variable with THIC 4, instead use a thickness position solve on the thickness of surface 4 (solves are set on the Lens Data Editor). There are two "parameters" that define a position solve: the reference surface number (parameter 1) and the total length (parameter 2). For the reference surface in this example, use "2". For the length, use whatever the distance is between surface 2 and surface 5, which is the distance between the first and third elements.

The solve will be active in all configurations, and so the first to third element spacing will always be fixed. Only five variables were used (instead of ten) and no ZTHI operand was required. However, the solve requires that the first to third element spacing be fixed, which is not what we wanted. Ideally, the spacing can be made a variable for optimization.

The trick is to add a new multi-configuration operand, TSP2 on surface 4. This operand is "Thickness Solve Parameter 2", which means you can make the total length value specified in the solve on surface 4 a multi-configuration variable. Enter in the desired starting thickness for the TSP2 in configuration 1. Then, in the other four configurations, place a pick-up solve on the multi-configuration screen to simply copy the configuration 1 value to the other configurations. Now, make just the configuration 1 TSP2 value a variable. This will allow the total length solve to be a single variable for optimization. Therefore, there are only six variables instead of 10, greatly improving the optimization efficiency. There is a sample file with three zoom positions called ZOOM-DEM1.ZMX which illustrates this technique.

This page intentionally left blank.

---

# COMMON QUESTIONS ABOUT ZEMAX

## Introduction

---

This chapter provides answers to commonly asked questions about installing and using ZEMAX. Where appropriate, references are provided to other locations in the manual, or elsewhere. If you have a question that is not addressed here, also check the Table of Contents and the Index.

## Questions about installation

---

- Q** How do I install ZEMAX?
- A** *Check Chapter 1, "Installation procedure".*
- Q** When I run ZEMAX, a message appears that says "Please attach ZEMAX hardware lock to parallel port!" and then the program terminates.
- A** *The problem is that ZEMAX cannot find the hardware lock. Make certain you have installed the lock correctly, and make sure it is attached on the parallel port of your computer. If you have a printer attached to the lock, make sure it is turned on. If the lock still fails, please call for assistance.*

## Questions about setting up lenses

---

- Q** ZEMAX does not seem to draw the rays correctly; the rays are not hitting the aperture stop.
- A** *The most common cause for rays to miss the stop, or to behave strangely, is that the stop surface occurs after a coordinate break. Avoid placing the stop after any coordinate breaks! If the stop follows a coordinate break, especially one with a decenter x or decenter y value, then the paraxial rays required to place the entrance pupil cannot be traced. ZEMAX uses the paraxial entrance pupil to reference the chief and marginal rays. Try moving the stop to a dummy surface before any coordinate breaks. If you have a rotationally symmetric system with an imbedded stop, then try turning "ray aiming" on.*
- Q** Why can't I set a thickness on the image surface?
- A** *ZEMAX uses the convention that the thickness of a surface is the distance to the next surface. Since the image plane is the last surface, there is no next surface. Although some programs use this thickness as a defocus term, ZEMAX does not. The reason is primarily to keep the interface consistent, and of course because ZEMAX does not force the image plane to be at paraxial focus in the first place. The overhead of using a dummy surface before the image plane for the defocus term is insignificant.*
- Q** Why can't I set a semi-diameter on the object surface?
- A** *ZEMAX sets the object size based solely on the field definitions. To achieve a particular object size, simply set it on the field data screen.*

## Questions about lens analysis

---

- Q** When I do a layout plot, there are rays drawn which are obviously beyond the clear apertures of the lens surfaces. Why are these rays not vignetted?
- A** *ZEMAX has two independent controls for setting surface semi-diameters and surface apertures. The values defined in the semi-diameter column are only used to determine how the lens is drawn, not whether or not rays are vignetted. Vignetting is determined only by surface apertures or vignetting factors. To vignette rays,*

*you must add a circular aperture (or whatever is appropriate) to that surface explicitly, or define an appropriate vignetting factor for that field point. See the chapter "Conventions and Definitions".*

**Q** On 3D layout plots, the rays seem to move behind a fold mirror, then reflect off a surface that isn't there, and then bounce off the mirror in the correct direction.

**A** *This is caused by a dummy surface with a zero thickness being placed immediately prior to the coordinate break that tilts the fold mirror. ZEMAX traces the rays to the dummy surface, and then back to the mirror surface. Coordinate break surfaces are always ignored when drawing surfaces and rays. The solution is to set the semi-diameter of the dummy surface to zero. Alternatively, do not use the dummy surface at all.*

**Q** When I perform a point-spread function (PSF) plot, the drawing looks like a bunch of scribbled lines, not a diffraction image.

**A** *Whenever you perform any diffraction calculation, including MTF, PSF, and diffraction encircled energy, you must verify that the wavefront amplitude and slope is within the sampling limit of the grid size used to perform the computation. To check this, perform an OPD fan and verify that the OPD is on the order of a few waves. Typically, a 8x8 sampling density is acceptable for less than one wave peak-to-valley OPD, 16x16 is acceptable for up to 2 waves, 32x32 is acceptable for up to 4 waves, and so on. Note that the peak-to-valley OPD is available on the surface wavefront plot and on the Zernike coefficient screen. Be sure to check each wavelength separately (this is why the OPD fan is easiest, it shows all wavelength data at once). If the OPD exceeds these approximate limits, then the diffraction calculation suffers from aliasing, and computed data can not be trusted. Fortunately, if the OPD is large, then the diffraction calculations are not required, use the geometric equivalents instead (RMS spot size and geometric MTF).*

**Q** How can I model an extended source, or a source with a complex structure?

**A** *Strictly speaking, ZEMAX only models point sources. The usual approach is to select several representative points on the extended source, and then design for maximum performance at those points. The variation in imaging properties over the field of view is generally slow enough to be adequately modeled with a few discrete points. You can however test the extended object imaging properties of a system in a very general way using the image analysis capability. See the chapter "Analysis menu" for a complete discussion of image analysis.*

**Q** What do the field curvature and distortion plots measure? Why does the field curvature data not start at zero?

**A** *See the definitions of field curvature and distortion in the chapter "Analysis Menu".*

## **Questions about optimization**

---

**Q** Why don't the Lagrange multipliers seem to work?

**A** *ZEMAX enforces a strictly decreasing merit function during optimization. In other words, ZEMAX should never return a system after optimization with a higher merit function than when it started. This means that Lagrange multipliers are not enforced if the result is a system with a higher merit function. To circumvent this problem, use weighted operands instead. Weighted operands usually are faster and more numerically stable than Lagrange operands. A large weight will usually drive the operand to the desired value if it is free to move.*

**Q** When using the global search algorithm, ZEMAX returns systems which have non-physical thicknesses and curvatures. How do I enforce boundary conditions during global search?

**A** *The way to define boundary conditions in global search are therefore exactly the same as in the conventional optimizer-by boundary condition operands in the merit function.*



- Q** How do you require a system to be telecentric in image space?
- A** *One way is to use the RANG operand. Require the RANG of the chief ray ( $hx=0$ ,  $hy=1$ ,  $px=0$ ,  $py=0$ ) to have a value of 0 in image space. This implies the chief ray is parallel to the optical axis, which means the exit pupil is "at infinity". The optimization algorithm will have to be run before the system actually becomes telecentric, and there must be sufficient variables (radii, thicknesses) to allow the design to reach the telecentric condition. The same trick works in object space.*
- Q** How else can a system be made telecentric in object space?
- A** *The RANG operand described above usually works fine. However, to avoid having to use optimization, simply make the thickness of the object some large number, 1E5 for example, and then place the stop on surface 1. Use a pickup solve on the thickness of surface 1 with a -1 scale factor so that the thickness is -1e5, which returns surface 2 to the object. Now, place the thickness of surface 2 to whatever the actual object distance is. Note that if the system is telecentric in object space, the object distance must be finite. The only drawback to this method is that the layout drawings will look strange unless the starting surface is set to "2" instead of the default "0". Note also that the aperture type should be "Object space N.A." or "Image Space F/#" for systems telecentric in object space. Don't use the other aperture types.*
- Q** How is the default merit function constructed?
- A** *See the chapter "Optimization", "Defining the default merit function".*
- Q** Why can't I optimize wavelengths, field values, or the system aperture value?
- A** *ZEMAX does not support direct optimization of these system-specification parameters. However, you can easily circumvent this limitation by using the multi-configuration feature (see the chapter "Multi-Configurations"). Since on the multi-configuration screen you can set and vary fields, wavelengths, weights, and system aperture values, you can easily optimize these values.*
- Q** Why does the optimization feature seem to ignore some of the field points, or why does the optimized system perform so much better at some wavelengths than others?
- A** *The most common reasons are 1) You have modified the field values or weights or the wavelength values or weights and then forgot to reconstruct the default merit function, and 2) The variables you have defined do not have sufficient influence on the merit function. See the chapter "Optimization".*
- Q** Why do the lenses that are optimized have negative edge or center thicknesses?
- A** *Unless you specifically constrain the optimization to exclude designs with negative edge or center thicknesses, the optimizer is free to set any value on any variable. To control edge and center thicknesses, use MNET and MNCT described in the chapter "Optimization".*
- Q** Why does the focal length keep increasing while the optimization proceeds?
- A** *You must either constrain the effective focal length using the EFFL operand, or by using a marginal ray angle curvature solve on the last lens surface. Alternatively, you can simply fix the back focal distance (rather than vary it). The reason the effective focal length increases is because slow lenses perform better, and unless specifically constrained, the optimizer will gravitate to slower F/# systems. This problem may also be circumvented by using image space F/# instead of entrance pupil diameter for the system aperture.*

This page intentionally left blank.

## Introduction

There are many files distributed along with the ZEMAX program. This chapter describes some of the various file types, and lists additional information about them. The following table describes the most common ZEMAX files according to the file extension. Not every file type used by ZEMAX is listed.

ZEMAX FILE TYPES BY EXTENSION

Extension	Description
AGF	ASCII Glass Format. These are ASCII files which contain the data for the glass catalogs. An ASCII format is used so that additional data may be incorporated in future releases of ZEMAX.
ANN	Annotation files. These binary files store user defined annotations for ZEMAX graphics.
BGF	Binary Glass Format. To accelerate the loading of AGF files, ZEMAX converts AGF files to BGF, which are the version specific binary images of the glass catalogs.
C	C language source code files.
CFG	Configuration file. ZEMAX.CFG is the main configuration file which describes the user-selected options on the environment dialog box. There are numerous other files that may be present which end in CFG, such as RAY.CFG, which contains the user-defined defaults for the ray fan plot control screen.
DLL	Dynamic Link Library files. These files are externally compiled programs linked into ZEMAX at run time. Typically, DLL's are used to implement user-defined surfaces.
EXE	Executable file. This includes the main program, WZEMAX.EXE and other executables and utilities used by ZEMAX.
IMA	Image file, used by the image analysis feature.
INF	Information file. Contains vendor name and address associated with test plate data.
NOT	Element drawing note files.
TPF	Test Plate Fit files. These contain test plate lists supplied by various optical fabricators, and are used in the test plate fit feature.
TXT	The default extension for files generated by the hardcopy feature.
ZMF	ZMF are compilations of ZMX format files which are used to define lenses in the stock lens catalogs.
ZMX	ZMX files are the files used to store lens data. The ZMX file is an ASCII file which contains the complete description of the lens, including apertures, wavelengths, prescription data, and the merit function.
ZPL	Files ending in the extension ZPL are ZEMAX Programming Language macros.

Some of the files with the extensions listed in the preceding table are described in the following sections.

## AGF glass catalog files

Glass catalog files are ASCII data files which end in the extension .AGF. These files are usually named after the glass vendor, such as SCHOTT.AGF for the Schott Optical Glass catalog. The catalogs supplied with ZEMAX are listed in the following table, along with the a description of the catalog.

GLASS CATALOG FILES

File Name	Description of catalog
CORNING.AGF	Corning France Optical Division, 44, avenue de Valvins - 77210 AVON France
INFRARED.AGF	Infrared material data. Due to unreliable data in the IR, verify all data prior to use.
HOYA.AGF	Hoya Optics, Inc., 3400 Edison Way, Fremont, CA 94538, USA
MISC.AGF	Miscellaneous materials such as silica.
OHARA.AGF	Ohara Corporation, 215 Ave Del Mar, Suite L, San Clemente, CA 92672 USA

File Name	Description of catalog
SCHOTT.AGF	Schott Glass Technologies Inc. 400 York Avenue, Duryea, PA 18642 USA
SCHOTTV.AGF	A subset of SCHOTT.AGF for preferred Schott glass types.
OLD_CORN.AGF	Older, obsolete Corning glasses.
OLD_HOYA.AGF	Older, obsolete Hoya glasses.
OLD_OHAR.AGF	Older, obsolete Ohara glasses.
OLD_SCHO.AGF	Older, obsolete Schott glasses.

## C language files

Sample C language programs are used to show correct file format generation. These C programs must be compiled before they can be run.

### C LANGUAGE FILES

File Name	Description of catalog
GRIDSAMP.C	Generates a file format used by the grid sag and grid phase surfaces. The program shows correct methods for generating numerical partial derivatives using only the sag function. Can be modified for any user defined surface type.
SAMPLE.C	This program will create a binary format image file called SAMPLE.IMA for use by the image analysis feature. The file shows parallel bars, one at each wavelength. The program can be used as a sample to demonstrate the correct way of creating a binary format image file.

## Image files

Image files are n x n character files which indicate the intensity, pixel by pixel, of a simulated object. The object can then be imaged by any optical system to be used as a test of resolution, or to represent the appearance of a particular type of source. See the "Analysis" chapter under "Image analysis" for more details. You can add as many image files as you like. The files listed here are samples which can be used as is or modified to suit your requirements.

### IMAGE FILES

File Name	Description of image
ALPHABET.IMA	The first 25 letters of the alphabet in a simple font arranged in a 5 by 5 grid. This is a useful for testing the resolution of systems imaging text or other complex shapes.
CIRCLE.IMA	A 99 x99 pixel approximation to a circular source. An example C program used to generate the file is also included as CIRCLE.C. You can recompile the C program to make circular IMA files of any size. Note this can be used as "point" source simply by setting the object scale to zero.
GRID.IMA	This image file is useful for showing the effects of large distortion. It consists of a five hollow boxes stacked in a grid.
LETTERF.IMA	This 7 x7 image file shows a block representation of the capital letter F.
LINEPAIR.IMA	A 20 x 20 image file which has ten pairs of alternating "black" and "white" bars. This pattern is used for testing geometric MTF. For example, if you would like to test the MTF at 50 line pairs per mm, then choose a field size of 10 divided by 50 or 0.2 mm. To test 100 line pairs per mm, choose 10 divided by 100, or 0.1 mm. If you are using units other than mm, then scale these numbers accordingly. This example assumes you are using field heights (not field angles) and that the line pairs per mm are measured in the object (not image) space. The object scale needs to be modified by the magnification of the system for image space line pairs per mm.
SQUARE.IMA	This is just a plain square object shape with uniform intensity.

## Note files

---

Note files are used on the element drawing feature. These are ASCII files which can be edited to contain any text. You may want to have a separate note file for each kind of lens drawing you make.

### NOTE FILES

File Name	Description of image
DEFAULT.NOT	The default ZEMAX element drawing note file. Do not modify this file; instead make your own note file. This way, upgrades will not overwrite your modifications.

## Test plate fit files

---

Test plate fit files end in the extension TPF. These files contain data supplied by optical fabricators which describes the test plates they have available. Usually optics which match existing test plates are cheaper to fabricate than optics which require new test plates. These files are used by the test plate fit feature available in ZEMAX-EE and XE.

### TEST PLATE FIT FILES

File Name	Description of vendor
APPLIOPT.TPF	Applied Optics, Inc., 3349 Vincent Road, Pleasant Hill, CA 94523, tel. (510) 932-5686.
COASTAL.TPF	Coastal Optical Systems, Inc., 3555 Fiscal Court #3&4, Riviera Beach, FL 33404, tel. (407) 881-7400
HJOL.TPF	Harold Johnson Optical Laboratories, Inc. 1826 W. 169th St. Gardena, CA 90247, tel (310) 327-3051.
JLWOOD.TPF	J. L. Wood Optical Systems, 1361 E. Edinger Avenue, Santa Ana, CA 92705, tel. (714) 835-1888.
KODAK.TPF	Eastman Kodak Company, 901 Elmgrove Road, Rochester, NY 14653-5718, tel. (716) 726-6478
KREISCHR.TPF	Kreischer Optics, Ltd., 906 N. Draper Rd., McHenry Illinois 60050, tel. (815) 344-4220
LGHTNING.TPF	Lightning Optical Corp., 431 E. Spruce Street, Tarpon Springs, FL 34689, tel. (813) 938-0092.
OFR.TPF	Optics For Research, P. O. Box 82, Caldwell NJ 07006, (201) 228-4480
OGF.TPF	Optico Glass Fabrication, 3160-A El Camino Real, Atascadero, CA 93422, tel. (805) 461-9402.
OPTIMAX.TPF	Optimax Systems, Inc., P. O. Box 611, Ontario, NY 14519-0611, tel. (315) 524-2231.
TORC.TPF	Tucson Optical Research Corporation, 210 S. Plumer, Tucson, AZ 85719, tel. (602) 882-0300
TOWER.TPF	Tower Optical Corporation, 130 Ryerson Avenue, Wayne, NJ 07470, tel. (800) 335-5582.

## ZMF stock lens files

---

ZEMAX is distributed with stock lens catalogs from many vendors, as listed in the following table.

## STOCK LENS VENDORS

Vendor Stock Name	Description of Vendor
EDMUND.ZMF	Edmund Scientific Co., 101 E. Gloucester Pike, Barrington, NJ 08007, tel. (609) 573-6250.
ESCO.ZMF	ESCO Products, Inc., 171 Oak Ridge Road, Oak Ridge, NJ 07438-0155, tel. (201) 697-3700.
GELTECH.ZMF	GELTECH, Inc., 109 Canada Road, Painted Post, NY 14870, tel. (607) 936-1494.
JML.ZMF	JML Direct Optics, 609 Portland Avenue, Rochester, NY 14621-5196, tel. (800) 456-5462.
LPT.ZMF	LightPath Technologies, 6820 Academy Parkway E. NE, Albuquerque, NM 87109, tel. (505) 342-1100.
MGRIOT.ZMF	Melles Griot, 1770 Kettering Street, Irvine, CA 92714, tel. (800) 835-2626.
NEWPORT.ZMF	Newport, 1791 Deere Ave., Irvine, CA 92714, tel. (800) 222-6440.
NSG.ZMF	NSG America, Inc., 19200 Von Karman, Suite 400, Irvine, CA 92612, tel. (714) 261-6212.
OFR.ZMF	Optics For Research, P. O. Box 82, Caldwell, NJ 07006, tel. (201) 228-4480.
OPTOSIG.ZMF	OptoSigma Corporation, 2001 Deere Avenue, Santa Ana, CA 92705, tel. (714) 851-5881.
ROLYN.ZMF	Rolyn Optics, 706 Arrowgrand Circle, Covina, CA 91722, tel. (818) 915-5707.
SPINHOYR.ZMF	Spindler & Hoyer, 459 Fortune Blvd., Milford, MA 01757, tel. (800) 334-5678.
USPL.ZMF	U. S. Precision Lens, 3997 McMann Road, Cincinnati, OH 45245, tel. (513) 943-5600.

Not every lens offered by these vendors is listed in the catalogs; only those lenses for which the vendor provided machine readable data. If you need data for a lens from a particular manufacturer which is not included in the catalog, please contact the manufacturer, and not Focus Software. We do not have any data on any lens which is not supplied in the catalogs.

### ZEMAX example lens files

ZEMAX is distributed with many example lens files. You may load these files to study how certain kinds of lenses are setup in ZEMAX, or to use as starting points for your own designs. The files are described in the following table.

#### EXAMPLE LENS FILES

File Name	Description
APLANAT.ZMX	Demonstrates use of aplanatic, ray normal, and pickup solves.
APOCH.ZMX	A three glass apochromat.
APOCH2.ZMX	Another three glass apochromat.
APOCH3.ZMX	And yet another three glass apochromat, with glasses suggested by D. Buchroeder.
AXICON.ZMX	Shows the simulation of an axicon shape using the standard surface.
CASSRC.ZMX	A simple Ritchey-Chretien cassegrain telescope.
COLLIMAT.ZMX	A collimating lens contributed by J. Treptau. This file demonstrates the use of a paraxial lens. allowing the image plane to curve is equivalent to tolerating some vergence in the afocal beam.
COOKE.ZMX	A simple Cooke type triplet, F/5, 20 degree semi-field, 50 mm EFL.
CYLINDER.ZMX	A simple cylindrical singlet using the toroid surface.
DBGAUSS.ZMX	This is a typical double Gauss configuration. F/3, 14 degree semi-field, 100 mm EFL.
DBL_TELC.ZMX	A doubly telecentric lens. This lens was optimized using the ENPP and EXPP operands to force the telecentric condition in both image and object space. Some lens design programs cannot correctly model such a system, but it poses no problem for ZEMAX.

File Name	Description
DIODE.ZMX	An example of simulating a laser diode source using a relay lens. See the chapter "Advanced Topics" for a complete discussion.
DOUBLET.ZMX	A doublet made of BK7 and F2.
ELLIPSE.ZMX	Illustrates a wide angle source using "object space cone angle".
GRATING.ZMX	A simple example of a diffraction grating. Note the paraxial lens used to bring the diverging beams to focus. Note ZEMAX can compute the OPD correctly for such a system.
HOE_01.ZMX	A simple example of a hologram constructed and used in transmission.
HOE_02.ZMX	A simple example of a hologram constructed and used in reflection.
HOUGHTON.ZMX	A Houghton type telescope designed and contributed by W. Cook.
HUBBLE.ZMX	The (correct) prescription for the Hubble Space Telescope. The HST is a Ritchey-Chretien cassegrain design.
INVTELE.ZMX	An inverse telephoto lens from Cox, contributed by D. Griffin. F/3.9, 31 degree semi-field, 1 mm EFL. Like all lenses, this one can be scaled within ZEMAX to the EFL you require.
LAIKIN.ZMX	A lens from M. Laikin. F/1.0, 2.5 degree semi-field, 4 mm EFL.
MAKSUTOV.ZMX	A reflecting telephoto design. F/10, 1.5 degree semi-field, 100 mm EFL.
MANGIN.ZMX	A Mangin mirror.
MILDOUB.ZMX	A doublet example using MIL number six-digit glasses. F/10, 0 degree semi-field, 100 mm EFL.
MIRRTILT.ZMX	The minimum configuration required for introducing a folding flat in the beam. Note the tilt pick-up solve on surface 4. Try changing the tilt on surface 2, then type Ctl-U to watch it update. Use Ctl-K to monitor the effects of the tilting.
MONOBEST.ZMX	The best form ever found for the monochromatic quartet (International Lens Design Conference, 1990). Whether this exact design is the global optimum or not depends upon how the lens is evaluated, however this is the best known design form.
MONOTEST.ZMX	The starting point for the ZEMAX global search algorithm that eventually led to the design presented as MONOBEST.ZMX.
NEWTON.ZMX	A simple newtonian telescope, similar to the example design session telescope.
NULLTEST.ZMX	A double-pass null test configuration contributed by W. Anderson.
OFFAXIS.ZMX	This example shows the correct way of setting up an off-axis segment of a parabola. Note the telescope is unobscured.
OFFNER15.ZMX	An Offner clamshell type null lens to test an F/1.5 parabola.
OFFNER17.ZMX	Another Offner design for a F/1.75 parabola.
PARAB90.ZMX	illustrates 90 degree rotated parabolas.
PARABOLA.ZMX	An F/5 parabola. ZEMAX can handle parabolas with apertures larger than their focal lengths. Some programs cannot handle anything faster than F/1.
PETZVAL.ZMX	A Petzval lens, F/2, 4 degree semi-field, 100 mm EFL.
PRISM.ZMX	The minimum configuration required to model a simple wedge prism. Note the coordinate break takes on the value of the material before it.
PRISM2.ZMX	This is a more complex 45-45-90 prism with a corrector lens. Note the use of the mirror, coordinate breaks, and square apertures.
PRISM3.ZMX	A cube beamsplitter prism example.
RUMAK.ZMX	Rumak designed by H. Rutten, from Telescope Optics, H. Rutten and M. Van Venrooij.
SC_ files	Any file starting with SC_ is a "short course" file used in teaching the Optical Design Using ZEMAX short course series. Contact Focus Software for more information.
SCANDEMO.ZMX	A very simple scanning system using multi-configurations.
SCHMIDT.ZMX	A flat-field Schmidt, F/1.33, 100 mm EFL.
SCHMIDTC.ZMX	A Schmidt cassegrain F/4, 1 degree semi-field, 200 mm EFL.
SELFOC.ZMX	A sample NSG Selfoc lens.

File Name	Description
SPIDER.ZMX	This is a cassegrain type telescope designed by W. Anderson, with a spider aperture added to demonstrate the use of the feature. F/6, .1 degree semi-field, 145 inch EFL.
TELECEN.ZMX	A system using paraxial lenses that is telecentric in image space.
TEMPDEMO.ZMX	A sample double with two configurations, one at 20 degrees c, the other at 100.
TESSR100.ZMX	A Tessar lens from the patent literature, contributed by L. Owen. F/5.5, 25 degree semi-field, 100 mm EFL.
TEST3D1.ZMX	This example shows the use of coordinate breaks and pickups to implement fold mirrors at various angles.
TEST3D2.ZMX	Another tilted component example.
TEST3D3.ZMX	Another tilted component example.
TEST3D4.ZMX	Another tilted component example.
TILT_OBJ.ZMX	Shows how to tilt the object plane using the ray aiming feature.
TILTDemo.ZMX	This example shows how to tilt an element about an arbitrary point. Note the use of the pickup solves.
TOROID.ZMX	A simple singlet toroidal lens.
WIDEANG1.ZMX	A wide angle lens from M. Laikin. Note the use of the ray aiming feature to insure the rays go through the stop.
WIDEANG2.ZMX	Another wide angle design.
WIDEANG3.ZMX	Another wide angle design.
YOLO.ZMX	A Yolo design from an Optical Sciences Center technical report, contributed by D. Nicholson.
ZOOMDEM1.ZMX	A simple three position zoom lens which illustrates basic zoom lens design techniques.

## ZPL example files

There are several example ZPL macros included with ZEMAX; they are described in the following table.

### EXAMPLE ZPL MACROS

File Name	Description
CENTROID.ZPL	This macro computes the location of the chief ray, centroid, and middle of the spot diagram at the primary wavelength for each field position.
EXAM1.ZPL	This is essentially identical to the example 1 ZPL macro given in the chapter "ZEMAX programming language". It prints out the chief ray positions for each field position.
EXAM2.ZPL	This is essentially identical to the example 2 ZPL macro given in the chapter "ZEMAX programming language". It estimates the wavelength-weighted RMS spot size on axis.
FF_TEST.ZPL	Illustrates the use of the FINDFILE keyword.
FOOSCRN.ZPL	Used for performing a footprint analysis.
GLOBAL.ZPL	Computes and displays global vertex coordinates and direction cosines.
GRAFTTEST.ZPL	A sample graphics macro similar to the one in the chapter "ZEMAX programming language".
GXYZ.ZPL	Sample macro which prints out global coordinate data.
PHASES.ZPL	Computes the radial distance between 2 pi phase changes for the binary 2 surface type.
POLDEMO2.ZPL	Computes an average polarization state.
RT.ZPL	A macro which traces rays and lists out surface normal vector components.
SYSTEM.ZPL	Illustrates the use of the GETSYSTEMDATA keyword.
ZOOMEDGE.ZPL	This macro computes the smallest semi-diameter required to pass all rays in all zoom positions for each surface. It then sets the semi-diameter of each surface to that semi-diameter, and prints out the new edge thickness in each zoom position.



This page intentionally left blank.

# ERROR MESSAGES

## Introduction

ZEMAX issues error messages to inform the user that some condition has arisen which requires special attention. Error messages may be issued when results that have been computed are known to be in error, as may be the case when some assumption in the algorithm has been violated. Error messages may also appear when invalid data has been supplied, such as a wavelength of zero.

Most error messages are reasonably self explanatory. The most common ones are numbered starting at 900, and are tabulated and elaborated upon in the following table. Not all possible error messages are listed in the table, just the most common ones.

### ZEMAX ERROR MESSAGES

Number	Description
900	"Unable to create edit window!". This error is usually caused by insufficient free memory or resources, which prevents ZEMAX from creating an edit window in a spreadsheet.
901	"Not enough memory to load program!". This is a fatal error caused by insufficient memory.
902	"Please attach ZEMAX hardware lock to parallel port!". If the hardware lock cannot be detected, this error is issued. Usually caused by a printer being attached to the lock, but turned off.
903	"Can't find glass *xxx* in catalog!". Caused by a glass name being listed in the glass column of the lens data editor which is not listed in one of the loaded catalogs. Check the System, General dialog box to see which glass catalogs are loaded. The glass names in each catalog can be checked using Tools, Glass Catalogs.
904	"Can't open File!". Either the file does not exist, or the directory path is invalid.
905	"Cannot perform 2D Layout on non-axial system!". This error is issued if a 2D Layout is requested when the system is non-axially symmetric. This may be due to coordinate breaks, x-field positions, or the presence of non-symmetric surfaces such as holograms.
906	"Too many windows open!". ZEMAX has a maximum number of windows which may be simultaneously open, depending upon available memory. Close one or more ZEMAX display windows to circumvent the error.
907	"Unable to open window!". This error is generated by the operating system, and is usually caused by insufficient free memory or resources.
908	"Too many clocks or timers!". Some versions of Windows have a limited amount of internal clock timers, and ZEMAX requires at least one free timer be available to run. The way to avoid this error is to close other applications which use clocks, then try ZEMAX again.
909	"Cannot make surface x concentric to/with surface y!". Concentric surface solves can only be for surfaces which follow the reference surface.
910	"Can't open File (filename)!". This is the same as error 904, except the file name is provided.
911	"Test plate file exceeds 64Kb, List is being truncated!". ZEMAX can only load test plate files which have less than 64Kb of data in them.
912	"At least one radii must be variable for test plate fit!". Test plate fitting assumes that the radii to be fit are variables for optimization. If there are no variable radii, no fitting can be done.
913	"There are no targets defined!". This error message indicates that the merit function is empty or undefined. The error only appears if an optimization is requested and no merit function has been defined yet.
914	"System cannot be updated!". This error message is generated by the optimizer when the initial optical system cannot be updated without errors. The errors are usually related to aperture rays not passing the system, or errors in the glass catalog data. Optimization cannot proceed until the system can be updated without errors. See the "Update" command in the "System Menu" chapter.
915	"Some of the operands cannot be computed!". If any of the operands in the merit function cannot be computed, this error is issued prior to optimization beginning. Usually the errors are that certain rays in the merit function cannot be traced as requested, due to total internal reflection or other errors. The errors must be corrected prior to optimization beginning.
916	"There are no variables defined!". At least one variable must be defined to optimize the lens.

Number	Description
917	"Error in target (n). TIR at surface (m)!". The n and m values will indicate which operand in the merit function failed, and at which surface. This error usually follows error 915. The requested ray at line n in the merit function editor cannot be traced, due to a TIR condition. TIR is an abbreviation for Total Internal Reflection, which means Snell's law cannot be satisfied for the ray in question at surface m.
918	"Error in target (n). Missed surface (m)!". Similar to error 917, except the ray misses the surface. This means the ray and the surface never intersect. Usually caused by the radius of the surface being too small.
919	"Out of parse space!". This error indicates ZEMAX ran out of memory reading a text file.
920	"Multi-config glasses cannot be variable!". Glasses which change under the Multi-config editor "GLSS" operand cannot be made into model glasses, and only model glasses may be made variable.
921	"Sampling too low, data inaccurate!". ZEMAX automatically detects when the sampling used to compute the diffraction encircled energy or MTF is too low. This is generally caused by too many waves of OPD error. Increase the grid size, or decrease the amount of aberration to get more accurate results. For encircled energy calculations, make sure the "Maximum Scale" box is checked on the settings dialog. Alternatively, instead use the geometric equivalents to get accurate results. For MTF, use the Geometric MTF. For Encircled Energy, use Geometric Encircled Energy.
922	"WARNING: Some field positions have zero weight!". Each wavelength has a weight associated with it. If the weight is zero, most computations ignore the wavelength.
923	"WARNING: Some wavelengths have zero weight!". See error 922.
924	"File too long for complete display, truncating!". ZEMAX will display text files in their entirety only if the file is less than about 10,000 lines long. Longer files are stored on disk, but only the first 10,000 lines are displayed. The exact number of lines depends upon the available memory.
925	"Cannot close window during processing!". This error indicates that a window is being closed while a computation is still incomplete. Early termination may cause severe problems with the stability of ZEMAX. To request early termination, try using the Escape key. If this fails, wait for the computation to complete before closing the window.
926	"Invalid surface choices!". Indicates that the surface or surfaces are not valid choices for the requested operation.
927	"Error loading catalog!". Usually caused by a glass or stock lens catalog having the incorrect file format.
928	"At least one entry must be in the catalog!". At least one glass type or one stock lens must remain in the catalog. The last entry cannot be deleted.
929	"Not enough room in catalog!". The catalog is full, and new entries cannot be added. Another catalog must be selected.
930	"Incompatible key type". This error is issued when the ZEMAX hardware key needs reprogramming.
931	"Maximum frequency cannot be adjusted when computing square wave response!". The MTF square wave response uses harmonics to compute the data, which requires that the full MTF be computed all the way to cutoff.
932	"Unable to find entrance pupil! Try moving stop to a dummy surface 1!". This error occurs when the entrance pupil cannot be located. This happens when many of the usual reference rays, such as the chief and marginal rays, cannot be traced. The general approach to working around the problem is to move the stop surface closer to the object surface, with fewer surfaces in between.
933	"No valid configuration file!". ZEMAX stores the default setting for most analysis features in a configuration file. If "Load" is pressed, and no configuration file exists, this error is generated.
934	"Invalid wavelength!". This is generated if a physically meaningless wavelength is entered, such as zero or -20.
935	"Can't use obj. spa. N.A. if object thickness is infinity!". If numerical aperture is used to define the system aperture, then the object distance must be finite.
936	"Non-zero semi-diameter required on Cubic Spline surface!". The cubic spline surface semi-diameter must be non-zero.
937	"Both Z1 and Z2 must be non-zero!". Certain surfaces, such as the hologram and conjugate surfaces, are undefined if the Z1 and Z2 values are zero. Non-zero values must be selected.

Number	Description
938	"No construction wavelength defined!". Caused when the hologram surface construction wavelength is zero.
939	"Non-zero step sized required on GRIN surface!". Gradient index surfaces require non-zero step sizes.
940	"Coordinate breaks cannot be mirrors!". Since coordinate breaks are always dummy surfaces, they cannot have an index change across their boundary. This means they cannot be mirrors.
941	"The thickness of the image surface is not used. For defocus, use a dummy surface prior to the image surface." ZEMAX does not use the thickness of the image surface for any purpose.
942	"Set object size on the field screen using field height!". The semi-diameter of the object surface is always computed automatically by ZEMAX. To maintain a particular object height, use the field dialog box. See the chapter "System Menu" under "Fields".
943	"System does not support OPD calculations!". Systems using ABCD surfaces and non-plane substrate Fresnel lenses cannot compute anything based upon optical path differences. This error appears if a OPD computation is requested and cannot be performed.
944	"Cannot print file!". An error generated by the operating system indicating some printer error has occurred.
945	"Insufficient Memory Available!". This catch-all error occurs when ZEMAX cannot allocate enough free memory to perform a computation. Try closing other programs or windows, or better yet, add more RAM to the computer.
946	"Surface n must be a supported type! Only supported surfaces may follow GRIN surfaces!". This error occurs when a surface type which is not supported for GRIN tracing follows a GRIN surface type. See the Chapter "Surface Types" under "The Gradient Index 1" surface type for a list of supported surfaces.
947	"Can't find GRIN 5 glass *xxx* in GRIN.DAT!". Occurs when a gradient index 5 surface specifies a glass name which is not properly defined in the file GRIN.DAT. See the "Surface Types" chapter for information on the gradient index 5 surface type.
948	"Cannot determine object coordinates!". Occurs when the field angles or heights are defined in such a way that the starting coordinates on the object surface cannot be determined.
949	"Pupil ray TIR at surface n". This error occurs when the rays being traced to define the entrance pupil cannot be traced through the system.
950	"Pupil ray missed surface n". See error 949.
951	"Entrance pupil cannot be located at object. Check object thickness and stop location!". This error occurs when the distance between the object surface and the entrance pupil is zero. These two planes cannot be at the same place.
952	"Cannot pick up curvature n from surface n!". This error is caused by a pick up solve on a surface attempting to pick up a value from the same surface, or from a surface which follows the solve surface. All pick up values must be from a previous surface. This is done to prevent circular definitions for values (i.e. 5 picks up from 7, 7 picks up from 5).
953	"Cannot pick up thickness n from surface n!". See error 952.
954	"Unable to converge OPD solve!". Caused when an OPD solve on a thickness cannot find any thickness which yields the specified OPD. This may be because the OPD is not a function of the thickness, such as for collimated light.
955	"Cannot pick up glass n from surface n!" See error 952.
956	"Cannot pick up conic n from surface n!" See error 952.
957	"Cannot pick up parameter i on surface n from surface n!" See error 952.
958	"Cannot pick up TCE n from surface n!" See error 952.
959	"Aperture ray TIR at surface n!". This error occurs when the rays being traced to define the system surface semi-diameters pupil cannot be traced through the system.
960	"Aperture ray missed surface n!". See error 959.
961	"The thickness of the image surface is not used. For defocus, use a dummy surface prior to the image surface!". The image surface thickness is not used by ZEMAX, and therefore cannot be edited.

Number	Description
962	"For variable glass data, use the Index Data surface type!". The glass type may not be set to a variable status. See the Chapter "Optimization" for details on optimizing glass choice.
963	"Stop diameter must be fixed if using 'float by stop size! ". The stop surface semi-diameter cannot be allowed to float freely if the system aperture is based upon this value. The stop semi-diameter must be user defined.
964	"Unknown Glass Formula!". The glass catalog defines a formula type not recognized by the program. Usually caused by a corrupted GLC glass catalog file.
965	"This object surface type is not supported!". Not all ZEMAX surface types may be used as object surfaces. If this error occurs, then use a different surface type as the object surface.
966	"Maximum Radius cannot exceed "x". Try increasing grid size". The encircled energy plot generates this error message if the specified radius is too great. The maximum possible radius is determined by the grid size. Larger grid sizes yield larger maximum radii. Make sure the "Maximum Scale" box is checked on the settings dialog box.
967	"Error in target i. DENC has insufficient sampling!". Caused by too low a sampling, or too much wavefront error to accurately compute the diffraction encircled energy.
968	"Can't find GRIN 6 glass *xxx* in GLC.DAT!". Occurs when a gradient index 6 surface specifies a glass name which is not properly defined in the file GLC.DAT. See the "Surface Types" chapter for information on the gradient index 6 surface type.
969	"Can't edit a model glass directly. Double click on this cell to invoke model parameters!" Occurs when a new glass is entered in the lens data editor on a surface where a model glass has been defined. Choose the "Glass Solves" menu option or double click on the cell to invoke the model glass solve dialog box. Change the type to fixed, then enter the glass name in the lens data editor.
970	"Multi-configuration lens tolerancing not supported. Tolerance each configuration individually." Multi-configuration lenses cannot be toleranced directly. Each configuration must be toleranced individually. To select a particular configuration, delete the configuration data for all but the configuration of interest. Update that single configuration using System, Update All. Then delete all the configuration operands for the remaining configuration. There can be no multi-configuration operands listed on the multi-configuration editor.
971	"Glass XX could not be found in the current catalogs. However, it was found in the XX catalog. Do you want to add this catalog to this lens?". This error message is displayed when a glass name is provided in the glass column of the Lens Data Editor that does not match any of the names in the currently loaded glass catalogs, but the glass name was found in another catalog on disk. The suggested catalog name is displayed, and an option to add that catalog is displayed.
972	"Multi-configuration lens merit functions must begin with a CONF operand!" When evaluating a merit function for a multi-configuration lens, the active configuration must be defined. Therefore, the CONF operand must be placed at line 1 in the merit function. All operands that follow are enforced only in the specified configuration, until the next CONF operand is reached. It is not true that operands placed before the first CONF command are enforced in all configurations. The required constraints must be placed beneath each CONF command as appropriate.
973	"Multi-configuration parameters cannot use this solve dialog. Change the status of this parameter on the Multi-configuration editor!". This error message indicates that the solve status of a value which is also a multi-configuration parameter cannot be changed from the lens data editor. The primary purpose of this message is to prevent conflicting definitions for a single value. For example, this prevents a thickness from being defined by a ray height solve while simultaneously being defined to be a pickup from another configuration.
974	"Computation aborted; invalid results!" This error message is printed on graphs and text windows when the computation was terminated prematurely. This is usually caused by either a lack of sufficient free system memory to complete the calculation, or by a user requested termination. Pressing the escape key during processing will generate this error. This error message is intended to notify that whatever data is presented may be incomplete or in error.
975	Unused
976	"Cannot pick up extra data y on surface x from surface x!". See Error 952.
977	"Cannot pick up semi-diameter on surface x from surface x!". See Error 952.
978	"Can't find GRADIUM profile *xxx* in PROFILE.DAT!". Occurs when a GRADIUM surface specifies a profile name which is not properly defined in the file PROFILE.DAT. See the "Surface Types" chapter for information on the GRADIUM surface type.

Number	Description
979	"Glass family xxx in PROFILE.DAT is not defined in GRIN.DAT!". Occurs when a glass family name referenced in the PROFILE.DAT file is not defined in the GRIN.DAT file. See the "Surface Types" chapter for information on the GRADIUM surface type which uses the PROFILE.DAT file.
980	"Invalid coating data, cannot compute coating absorption!". This error occurs when ZEMAX cannot accurately compute the absorption through a metallic layer of a thin film coating. This is usually caused by an absorptive layer that is too thick for any light to transmit through, such as several microns of a metal layer.
983	"Wavelength x is not within valid range of dispersion formula for glass S!". All defined wavelengths must be within the min and max wavelength values specified in the glass catalog. This is to help prevent accidental use of a dispersion formula outside the valid range.

This page intentionally left blank.

# THE ZMX FILE FORMAT

## Introduction

ZEMAX saves and loads lens files using the ZMX file format. The ZMX format has been designed to be easy to maintain, modify, and understand. ZMX can be easily incorporated into other programs using the information in this chapter. For example, you may want to write a program to convert lens data from another lens design program to ZMX format, or you may want to read in ZMX files to perform additional analysis outside of ZEMAX. ZMX uses simple ASCII four letter mnemonic commands.

## ZMX commands

The following table defines each ZMX keyword and syntax.

ZMX FORMAT COMMANDS

Command and syntax	Description
APLA	Curvature aplanatic solve.
CHIA val	Curvature chief ray angle solve value.
CHZH hgt	Thickness chief ray height solve.
CLAP min max	Circular aperture min max.
CNOR	Curvature chief ray normal.
COAT	The name of the coating used by the surface, if any.
COMM	Comment for the current surface.
CONI val	Conic constant.
CURV val	The curvature value of a surface.
DIAM val solvecode pusurf	The semi-diameter of a surface. The solvecode is 0 for automatic, 1 for fixed, 2 for pickup and 3 for maximum. The pusurf value is only required for solvecode = 2.
DISZ val	The Z thickness of a surface.
EDGE thi rad	Thickness edge thickness solve.
ELAP xwid ywid	Elliptical aperture.
ELOB xwid ywid	Elliptical obscuration.
ENPD val	Entrance pupil diameter value.
ENVD temp pres	Environment data. The temp and pres values are in degrees c and atmospheres, respectively.
FLOA val	Float by stop size. The val argument is ignored.
FNUM val	Image space F/# value.
FTYP i	Field type. The integer i is 0, 1, or 2 for field angle, object height, or image height.
FWGT val1 val2 val3...	The field weights. See XFLD.
GARR i sag dx dy dxdy	Used for defining the grid surface data. "i" is the array index, starting at 0. The other values are the sag, the derivative of the sag in the x direction, the derivative of the sag in the y direction, and the second cross derivative. Must follow after the GDAT command.
GCAT name1 name2 name3 ...	Indicates which glas catalogs are used by the lens.
GDAT ix iy dx dy	Defines the grid data header for the surface. The values are the number of x and y pixels, and the delta x increment and delta y increment.
GFAC val type	Apodization factor value and type. The type code is 0 for none, 1 for Gaussian, and 2 for tangential.



Command and syntax	Description
GLAS name code pu nd vd pd vnd vvd vpd	The glass of the surface. Name is the glass name, such as BK7. code is 0 for fixed, 1 for model, 2 for pickup. Pu is the pickup surface, which must be supplied even if the solve is not a pickup. nd, vd, and pd are the index, abbe, and partial dispersion values, which are only used if the glass is a model. The vnd, vvd, and vpd codes are 0 if that parameter is variable, else they are unity. The glass name must be supplied, even if the glass is a model glass.
MARA val	Curvature marginal ray angle solve value.
MAZH hgt zone	Thickness marginal ray height solve.
MNOR	Curvature marginal ray normal.
MNUM n	Number of multiconfigurations.
NAME string...	Indicates the "lens name" description.
OBDC xdec ydec	Obscuration decenterx and decentery.
OBNA val	Object space numerical aperture value.
OBSC min max	Circular obscuration min max.
OPDZ opd zone	Thickness OPD solve.
PARM n val	Parameter n is the value of "val".
PCUP n val	Curvature pickup solve from surfac n times val
PICB i	Paraxial Ignore Coordinate Breaks, i = 0 is false, i = 1 is true. Default is true.
PKUP n val	Conic pickup from surface n times val.
PPAR i n val	Parameter i pickup from n times val.
PUSH val	Pupil Shift. Used only if ray aiming is on. The "val" is in lens units.
PWAV n	The primary wavelength indicator.
PZUP n val	Thickness pickup from n times val.
RADI val	The radius of curvature. Now obsolete, use CURV.
RAIM tol	If present, indicates ray aiming is turned on with the specified tolerance value.
ROPD i	Reference OPD setting. For i = 0, 1, and 2, the reference point is Absolute, Infinity, and Exit Pupil, respectively. Use Exit Pupil as the default.
SPID numarms width	Spider aberture.
SQAP xwid ywid	Rectangular aperture.
SQOB xwid ywid	Rectangular obscuration.
STOP	Indicates the current surface is the stop surface.
SURF	Increments the surface counter.
TCED val v	TCE value of surface. v is unity if variable, else 0.
TOLE n length	Thickness total length solve.
TYPE codename	Indicates the current surface type. The codenames are listed in a following section.
UNIT code	Specifies the lens units. The "code" must be either MM, CM, IN, or METER.
VCON	Conic is variable.
VCUR	Curvature is variable.
VDSZ	Thickness is variable.
VERS n	The version number of ZEMAX that created the file.
VPAR n	Parameter n is variable

Command and syntax	Description
WAVL val1 val2 val3...	The wavelength values in microns.
WWGT val1 val2 val3...	The wavelength weights.
XDAT n val v pus sca	Extra Data Parameter n has value val. If v is unity, then the value is variable. If v is 2, then there is a pickup solve on the extra data parameter. The pickup surface number is pus and the scale factor is sca.
XFLD val1 val2 val3...	The x-field values. There are from 1 to 9 values separated by spaces. The units depend upon the field type, either degrees or lens units.
YFLD val1 val2 val3...	The y-field values. See XFLD.
ZVCX val1 val2 val3...	The VCX vignetting factors.
ZVCY val1 val2 val3...	The VCY vignetting factors.
ZVDX val1 val2 val3...	The VDX vignetting factors.
ZVDY val1 val2 val3...	The VDY vignetting factors.

## ZMX format for optimization merit functions

---

ZEMAX simply appends the optimization merit function to the end of the ZMX file. After the lens data is all defined, the optimization operands follow in the following format:

```
NAME int1 int2 hx hy px py target weight
```

Note the arguments are all separated by spaces, and that the value of the operand is not stored. All operands use this format, even if not all data fields are used by the operand. unused values are ignored, but must be present. ZEMAX will keep track of the number of operands as the file is loaded again. There is no need to write out how many operands are present. For a list of optimization operands, see the chapter "Optimization".

## ZMX format for multi-configuration operands

---

ZEMAX also appends all multi-configuration data to the lens file. There are two formats used. If the multi-configuration operand is GLSS, then this format is used:

```
GLSS surface config name
```

Otherwise the more general format is used:

```
NAME surface config value variable_status
```

The variable\_status is 0 for not variable, and 1 for variable. For a list of multi-configuration operand names, see the chapter "Multi-configurations".

## Minimum ZMX file

---

The simplest "minimum" ZMX file looks like this:

```
NAME Lens has no title.
UNIT MM
ENPD 0
GFAC 0
GCAT schott misc
FTYP 0
ROPD 2
PICB 1
XFLD 0
YFLD 0
FWGT 1
WAVL 0.55
WWGT 1
PWAV 1
```

```

SURF 0
  TYPE STANDARD
  CURV 0
  TCED 0 0
  DISZ INFINITY
SURF 1
  STOP
  TYPE STANDARD
  CURV 0
  TCED 0 0
  DISZ 0
SURF 2
  TYPE STANDARD
  CURV 0
  TCED 0 0
  DISZ 0
MNUM 1

```

The data consists of a header block, starting with the NAME keyword, and ending with the PWAV keyword. The header block defines all of the system parameters, such as fields, wavelengths, units, and other data not associated with any surface. Note also the keyword "INFINITY" can be used for infinite distances.

After the header block, the SURF 0 command indicates that all data following refers to surface 0, the object surface. When the next SURF command is reached, the surface counter is incremented and then data for the next surface is defined. Note that the number following the SURF command is ignored! it is only printed by ZEMAX to make the file easier to read. Each SURF command increments the surface counter, and the counter cannot be reset to add data for an earlier surface.

The tabs which indent the data between SURF commands are also added by ZEMAX simply to make the file easier to read, and are not required.

## Surface type codes

---

The surface type codes are 8 character names which define the surface type. The names are similar to the names of the surface types, as defined in the Chapter "Surface Types":

```

STANDARD , EVENASPH , ODDASPHE , PARAXIAL , PARAX_XY , TOROIDAL , BICONICX , TOROGRAT
CUSPLINE , HOLOGRM1 , HOLOGRM2 , COORDBRK , POLYNOMI , FRESNELS , ABCDSURF , ALTERNAT
DGRATING , CONJUGAT , TILTSURF , IRREGULA , GRINSUR1 , GRINSUR2 , GRINSUR3 , GRINSUR4
GRINSUR5 , GRINSUR6 , GRINSUR7 , GRINSUR8 , GRINSUR9 , GRINSU10 , ZERNSURF , ZERNPHAS
XPOLYNOM , BINARY_1 , BINARY_2 , XCUSPLIN , XASPHERE , XOSPHERE , VARLSGRT , ELLIGRAT
SUPERCON , XFRESNEL , GRID_SAG , GRID_PHA , GEN_FRES , PERIODIC , TOROHOLO , JONESMAT
ATMOSPHR , ZONEPLAT , USERSURF

```

## Additional examples

---

For additional ZMX example files, see any of the sample files that come with ZEMAX.

This page intentionally left blank.

# Index

## A

---

Abbe V number  
  Boundary constraints on,13-7  
  Definition for,13-24  
  For MILNUM glasses,17-4  
  Operand,13-12 - 13-13

ABCD surfaces,11-9

Aberration coefficients,8-34 - 8-35, 13-17  
  Optimizing,13-6

Aborting  
  Editing of cell values,2-2  
  Long computations,2-5

Active configuration  
  Defined,3-1

Address box  
  Environment options,5-2

Advanced data,7-2

Advanced topics,21-1

Afocal systems,11-4

Air pressure,18-1

Airy disk  
  Defined,8-8  
  Point Spread Function,8-14

Aliasing,22-2

Alternate surface intersection,11-9

Angle of incidence,13-16

Angular magnification  
  Defined,3-1  
  Operand,13-7

Apertures  
  See Surface aperture  
  See System aperture

Aplanatic condition  
  Solving for,12-2

Aplanatic lens  
  Example of,23-4

Achromat  
  Example of,23-4

Apodization  
  Defined,3-1  
  Gaussian,7-4, 13-5

Apodization factor  
  Defined,3-1  
  Specifying,7-4

Appending  
  Data files in ZPL macros,20-28  
  Lens files,5-1

Array variables  
  ZPL macros,20-36, 20-40

Aspect ratio  
  Printing,5-3

Aspheric surfaces,11-3  
  High order terms,11-24 - 11-25  
  Superconic,11-27

Asterisk  
  Indicates surface apertures,6-3

Astigmatic sources,21-1

Astigmatism  
  Calculating,8-34  
  Operand,13-7

Athermal lenses  
  Analyzing and optimizing,18-4

Atmospheric refraction,11-33

Auto update  
  During optimization,13-22  
  Editors,5-3

AutoCAD  
  DXF format,8-4

Axial color  
  Operand,13-7

Axial symmetry,13-4

Axicons  
  Modeling,11-4, 21-3

## B

---

Beam expander  
  Tutorial,4-17

Beamsplitter  
  Polarization modeling,19-7

Bending  
  Using fold mirrors,9-6

Best fit glass,10-1

Best-fit sphere,8-37

Biconic  
  Surface type,11-5

Binary optics,11-19, 11-21 - 11-22  
  Sign conventions for,11-22

Binary surface  
  Optimization of,11-22, 11-24

Boundary conditions  
  On (center) thickness,13-8  
  On curvature,13-8

Boundary operands  
  Understanding,13-20

Bugs  
  Reporting,1-2 - 1-3

Button bar  
  Defining,5-5

## C

---

C programs,23-2

Capping  
  GRADIUM profiles,11-17

Cardinal points,3-1, 10-2

- Cassegrain
  - Example of,23-5
- Cell
  - Editing contents,2-2
- Cell width
  - Changing,5-3
- Center thickness
  - Operand,13-8, 13-12 - 13-13
- Centroid,8-7
  - Determining coordinates of,8-9
  - Encircled energy,8-19
  - Optimization,13-14, 13-18
- Chief ray
  - Definition of,3-1
  - Normalized coordinates,3-5
  - Solve,12-2 - 12-3
- Chromatic aberration,8-30
- Clipboard
  - Copying graphics to,2-2
  - Copying text to,2-4
  - Cut, Copy, and Paste,6-4 - 6-6
  - Selecting metafile format,5-3
  - Using,2-6
- Coatings
  - Editing,9-6
  - Optical,8-41
  - Properties of,8-41
- COLOR
  - ZPL macros,20-11
- Colors
  - Changing default,5-2
  - Defining,5-3
- Coma
  - Calculating,8-34
  - Operand,13-7
- Comments,6-1
  - Hide/Show menu,6-4
- Company name
  - On plots,5-2
- Cone angle,7-1
- Configuration
  - Changing,7-7, 16-3
  - See also Zoom position
- Conic
  - Description of,11-3
  - Solve for,12-5
- Conical surfaces
  - See Axicons
- Conjugate
  - Surface type,11-10 - 11-11
- Constant
  - Operand,13-8
- Cooke triplet
  - Example of,23-4
- Coordinate breaks
  - And paraxial rays,7-5
  - Detailed examples of,4-20
  - Pitfalls and suggestions,11-8
  - Reversing with "order",11-8
  - Surface type,11-8
- Coordinate system
  - Conventions for,3-2
- Copy
  - Operand data,6-5, 6-7
  - Surface data,6-4
- Cosine surface,11-31
- Cross sections
  - Point Spread Functions,8-15
- Cubic spline
  - Extended surface type,11-24
- Cubic spline surfaces,11-6
- Curvature
  - Boundary conditions on,13-8
  - Operand,13-8
- Cut
  - Surface data,6-4
- Cylindrical lens
  - Example of,23-4
  - Paraxial model for,11-4
- Cylindrical surfaces,11-5, 11-31
  - Gratings placed on,11-6

## D

---

- Date
  - Printing on plots and text,5-3
- Debugging
  - Tolerancing,15-15
- Decenters
  - Tolerancing,15-3
- Decenters and tilts
  - With coordinate break surfaces,11-8
- Decimals
  - Changing default,5-3
- Default merit function,13-1
  - Start at row,13-9
- Default path
  - For output,5-2
- Defaults, changing,5-2
- Defocus,22-1
- Delano diagrams,8-29
- Deleting
  - Operands,6-5 - 6-6
  - Surfaces,6-1, 6-4
  - Surfaces in ZPL macros,20-12, 20-14
- Density
  - Of lenses,10-2
- Dialogs,2-1

- Diameter
  - Operand,13-9
- Diattenuation
  - Of coatings,8-42
- Diffraction gratings,11-10
  - On cylinders,11-6
  - On toroids,11-6
- Diffraction Imaginary Transfer Function,8-11 - 8-12, 8-15 - 8-17
- Diffraction limited,3-2
- Diffraction Modulation Transfer Function,8-11 - 8-12, 8-15 - 8-17
- Diffraction Phase Transfer Function,8-11 - 8-12, 8-15 - 8-17
- Diffraction Real Transfer Function,8-11 - 8-12, 8-15 - 8-17
- Diffractive optics,11-30
- Diodes
  - Simulating,21-1
- Directories
  - Default path for output,5-2
- Directory names
  - Environment options,5-2
- Dispersion,8-30 - 8-31
- Dispersion coefficients,17-2
- Distortion,8-26, 8-28
  - Calculating,8-34
  - Definition of,8-27
  - F-Theta,8-27
  - Grid,8-27
  - Misleading data,8-24
  - Operand,13-9
  - Representing,8-22
- dn/dt,18-4
- Double Gauss lens
  - Example of,23-4
- Doublet
  - Example of,23-5
  - Tutorial,4-5
- DXF files
  - On 2D layouts,8-2
  - On solid model layouts,8-3
  - On wireframe layouts,8-4

## **E**

---

- Edge response,8-21
- Edge thickness
  - Controlling,12-3
  - Defined,3-2
  - Determining,10-1
  - Operand,13-10, 13-12 - 13-13, 13-18
  - Questions about,22-3
  - X- direction,13-19

- Editing
  - Spreadsheet values,2-2
- Editor windows,2-1
  - Environment options,5-3
  - Menu,6-1
- Editors
  - Environment options,5-3
- Effective focal length
  - Defined,3-2
  - Displaying,10-1
  - Operand,13-9
  - Questions about,22-3
- Efficiency
  - Coupling into fibers,8-22
- Element drawings,8-4
- Element power
  - Solve,12-2
- Ellipse
  - Defining,11-3
- Elliptical source
  - Defining,21-3
- Elliptical surface
  - Defining,11-10 - 11-11
- Encircled energy,8-19
- END
  - ZP macros,20-20
- Ensquared energy,8-19 - 8-20
- Entrance pupil diameter
  - Defined,3-2
  - Operand,13-9
  - Setting,7-1
- Entrance pupil position
  - Defined,3-3
  - Operand,13-9
- Environment,5-2
  - Address box,5-2 - 5-4
  - Defining,18-1
  - Directory names,5-2
  - Temperature and pressure,18-1
- Environment menu,5-2
- EQUA
  - Operand,13-10
- Error messages
  - Table of,24-1
- Escape
  - Aborting long computations,2-5
- Exit pupil
  - Significance of,7-4
- Exit pupil diameter
  - Defined,3-3
- Exit pupil position
  - Defined,3-3
  - Operand,13-10
- Extended asphere

- Optimization of,11-25
- Extended cubic spline
  - Surface type,11-24
- Extended polynomial
  - Optimization of,11-21 - 11-22, 11-24 - 11-25
  - Surface type,11-20
- Extended sources,8-22, 22-2
- Extra data
  - Defined,3-3, 11-1
  - Operand,13-19
  - Optimization of,13-25
  - ZPL macros,20-13, 20-16

## F

---

- F-theta
  - DISC operand,13-8
- F-Theta distortion,8-27
- F/#
  - Displaying,10-1
  - For fields, wavelengths,10-2
- Fabrication support,8-38
- Faceted surfaces,11-31
- Fast Asphere Trace,7-5
- Fiber optics
  - Coupling into,8-22
- Field angles
  - Conventions for,3-3
- Field curvature,8-25 - 8-26
  - Calculating,8-34
  - Operand,13-10
- Field curvature plot
  - Meaning of,8-26 - 8-27
  - Offsets on axis,8-27, 8-29
- Field positions
  - Defining,7-6
  - Optimizing,22-3
  - ZPL macros,20-26
- File format
  - ZMX,25-1
- File menu,2-1, 5-1
- File names
  - On plots,5-2
- Files
  - Finding names in ZPL macros,20-13
- Fitting
  - Test plates,9-5
- Flip
  - Lens elements,9-3
- Float by stop size,7-1
  - Defined,3-3
  - Described,7-1
- Focus
  - Quick focus tool,9-6

- Fold mirrors
  - Example of,23-5 - 23-6
  - Implementing,4-20
  - Inserting,9-6
- Font size
  - Changing,5-3
  - In text windows,5-3
  - Setting in ZPL macros,20-35 - 20-36
  - When printing,5-4
- Foot prints
  - Beam,8-22
- FOR loop
  - ZPL macros,20-14
- Fresnel
  - General polynomial form,11-31
  - Zone plates,11-33 - 11-34
- Fresnel surfaces,11-9, 11-28, 11-31

## G

---

- Gaussian beams,8-32
  - Operands,13-10
- Gaussian factor
  - Optimization,13-5
- General data,7-1
- Geometric Modulation Transfer Function,8-10
- Geometric MTF,8-12
- Ghost reflections,9-3
- Glass
  - Excluding from optimization,17-2
  - Transmittance of,8-30
- Glass catalog,8-30 - 8-31, 9-1, 17-1
  - Adding new glasses to,17-2 - 17-3
  - Creating new,17-2
  - Managing multiple,17-1
  - Selecting,17-1
  - Sorting and searching,17-4
  - Transmission data,17-4
- Glass catalogs
  - Described,23-1
- Glass data
  - Obsolete glasses,17-6
- Glass name
  - ZPL macros,20-9
- Glasses
  - Best fit,10-1
  - Constraining,13-24
  - Entering,3-3
  - Model,10-1
  - Optimizing,13-23
  - Optimizing with RGLA,13-17
  - Pick up solves,12-4
  - Removing in multi-configurations,16-3
- Global coordinates



- Calculating,10-2
- From ZPL macros,20-5
- Operand,13-10, 13-16
- Reference data,10-2
- Global optimization,9-1, 13-27, 14-1
- Global search
  - Problems with boundaries,22-2
- GOSUB,20-20
- Gradient index,11-12 - 11-14, 11-17 - 11-18
  - Boundary operands for,13-25
  - Dispersion model,11-13, 11-18
  - Operand,13-11
  - Selfoc lenses,23-5
  - Vignetting,7-5
- Gradient Lens Corporation,11-15
- Graphics
  - Colors,5-3
  - Copying to clipboard,2-2
  - Environment options,5-2
  - Exporting using Clipboard,2-6
- Gratings,11-10
  - Example of,23-5
  - Toroidal,11-6
- Grid phase surface,11-30
- Grid sag surface,11-29

## H

---

- Hammer optimization,9-1
- Hardware lock
  - Installing,1-2
  - Policy regarding,1-3
  - Problems with,22-1
- Help
  - Button,2-5
  - Menu option,2-1
- Hexapolar rings
  - Defined,3-3
- Hidden line drawings,8-2 - 8-4
- Holograms,11-7
  - Example of,23-5
  - General polynomial form,11-21 - 11-22
  - Toroidal,11-31
  - Used in reflection,11-7
  - Using Zernike coefficients,11-19
- HPGL
  - File format,2-6
- Huygens
  - Point Spread Function,8-15
- Hx
  - Defined,3-4
- Hy
  - Defined,3-4
- Hyper hemispheres

- Preventing with HHCN,13-11
- Hyperbola
  - Defining,11-3

---

- Illumination,8-22
- Image analysis,8-22
  - Aborting long computations,2-5
  - Binary format sample,23-2
  - Editing IMA files,8-23
  - IMA file format,8-23
- Image height
  - Field definition,3-3
- Image plane
  - Coordinates,8-9
- Image space F/#
  - Operand,13-11
  - Setting,7-1
- Images
  - Appearance of,8-22
- Index data
  - Operand,13-11
  - Optimizing,13-23
- Index of refraction,17-3 - 17-4
  - Determining,10-1
  - Dispersion plot,8-30
  - Table of values,10-2
  - Temperature and pressure effects,18-1
- Inequality
  - Operand,13-14
- Input
  - ZPL macros,20-24
- Input focus,2-2
- Inserting
  - Operands,6-5 - 6-6
  - Surfaces,6-1
  - Surfaces,6-4
  - Surfaces in ZPL macros,20-24
- Installation
  - Of the hardware lock,1-2
  - Questions about,22-1
- Integer
  - Rounding to using INTE,20-4
- Interferogram
  - Data for surface deformation,11-18 - 11-19
  - Drawing,8-17
- Interferometers
  - Using data from,11-19
- Invariant,8-35
- Inverted telephoto
  - Example of,23-5
- Irregular
  - Surface type,11-11

Irregularity  
Tolerancing,15-4

## J

---

Jones matrix,19-6  
Jones matrix surface,11-32  
Jones vector,19-1

## K

---

Keyboard  
Shortcuts,2-5  
Use of,2-5  
Keyboard shortcuts,2-2, 2-5  
Kinofoms,11-21 - 11-22

## L

---

Lagrange invariant,8-35  
Operand,13-12  
Lagrange multipliers  
Problems with,22-2  
Using,13-6  
Lambertian  
Sources,8-22  
Landscape orientation,5-4  
Laser diodes  
Simulating,21-1  
Lateral color,8-29  
Operand,13-11  
Layout  
Element drawing,8-4  
Layouts  
Phantom rays on,22-2  
Scale bar width,5-4  
Three-dimensional,8-2 - 8-4  
Two-dimensional,8-1  
Lens catalog,9-2  
Lens Data Editor  
Printing contents of,10-2  
Lens design  
References on,1-1  
Lens element  
Reversing,9-3, 9-6  
Lens files  
Appending,5-1  
Deleting,5-1  
Examples of,23-3 - 23-4  
Loading,5-1, 20-25  
Recently used list,5-5  
Saving,5-1, 20-34  
ZMX format,25-1  
Lens mounts  
Thermal expansion of,18-1

Lens search,9-2  
Lens units  
Defined,3-4  
Specifying,7-2  
Lenses  
Plotting to scale,8-2  
Line response,8-21  
Loading  
Lens files,5-1  
Lock Window,2-2, 2-4  
Long computations  
Aborting from,2-5  
Longitudinal aberration,8-21, 8-29  
Longitudinal aberrations  
Calculating,8-34  
Longitudinal color  
Calculating,8-34  
LPTD  
Operand,13-25

## M

---

Macros  
See ZPL macros  
Magnification  
Displaying,10-1  
Operand,13-7, 13-15  
Main window,2-1  
Maksutov  
Example of,23-5  
Mangin mirror  
Example,23-5  
Marginal ray  
Definition of,3-4  
Normalized coordinates,3-5  
Solve,4-2, 12-2 - 12-3  
Margins  
Setting for printing,5-4  
Mass  
Of lenses,10-2  
Maximum aspheric deviation,8-38  
Maximum field  
Defined,3-4  
Merit function  
Default,13-1, 22-3  
Defining,13-1  
Formula for,13-6  
Gaussian apodization of,13-5  
Ignoring operands in,13-9  
Loading and saving,6-5  
Loading from ZPL,20-26  
Modifying,13-5  
Percent contribution of components,13-20  
Printing,10-2

- ZPL macros,20-27
- Metafile
  - Windows clipboard format,5-3
- MIL number glasses,17-4
- Mirror
  - Inserting,9-6
- Mirrors
  - Conventions for,3-7
  - Specifying,3-3, 6-2
  - Tutorial on fold mirrors,4-20
- Model
  - Glass,12-4
- Model glass,10-1
- Modulation Transfer Function,8-11 - 8-12, 8-15 - 8-17
- Monochromatic quartet
  - Example of,23-5
- Mouse
  - Shortcuts,2-3 - 2-4
- MTF
  - Aborting from long computation,2-5
  - Calculating in macros,20-17
  - Geometric approximation,8-12
  - Operand,13-13
  - Optimization of,13-21
  - Through-Focus,8-11
  - Visualizing,8-22
- Multi-configuration,16-1
  - Operand values,13-12
  - Optimization and,16-3
  - Tolerancing,15-14
  - Tutorial,4-17

## N

---

- Narcissus
  - YNI contribution,13-19
- Neural networks,14-1
- Newtonian telescope
  - Example of,23-5
- Nodal points,3-1, 10-2
- Non-paraxial system
  - Defined,3-4
- Normal distribution,15-14
- Normalized coordinates,13-6, 20-30, 20-32
  - Defined,3-4
- Note files,23-3
  - Editing,8-5
- Notes,7-2
- NSG America, Inc.,11-18
- Numerical aperture
  - And ray aiming,7-2

## O

---

- Object height,22-1
- Object plane
  - Tilting,11-11, 23-6
- Object space N.A.
  - Setting,7-1
- Object space numerical aperture
  - Defined,3-5
  - Operand,13-14
- Obscurations,6-3
  - Circular,6-3
  - Elliptical,6-3
  - Rectangular,6-3
  - Spider,6-3
  - Vignetting from,8-25
- Obsolete
  - glasses,17-6
- Off-axis parabolas
  - Example of,23-5
- Offner lens
  - Example of,23-5
- OPD
  - See Optical path differences
- OpenGL
  - Shaded model graphics,8-4
- Operand data
  - Cut, Copy, and Paste,6-5
- Operands
  - Defining complex,13-22
- Optical design
  - References on,1-1
- Optical fibers
  - Efficiency,8-22
- Optical invariant
  - See Lagrange invariant
- Optical path differences
  - Operand,13-14
  - Referencing,7-4
  - Solves,12-4
- Optimization,9-1, 13-1
  - Global,9-1, 14-1
  - Global minimum,13-27
  - Global search,14-1
  - Hammer,9-1, 14-1
  - Merit functions for,13-1
  - Of glasses,13-23
  - Operands,13-5, 13-22
  - Requirements,13-1
  - Selecting variables for,13-1
  - Suggestions for use,13-27
  - With multi-configurations,16-3
  - ZPL macros,13-26

Orientation  
Graphics,5-4

## P

---

Pan and zoom,2-4

Parabasal

Discussion of,3-5

Parabola

Defining,11-3

Example of,23-5

Parameter data

Defined,3-5, 11-1

Tolerancing,15-5

Paraxial

Definition of,3-5

Working F/#,3-6

Paraxial focus,12-3

Moving to,4-2

Paraxial image height

Defined,3-6

Paraxial lenses,11-4

Paraxial magnification

Defined,3-6

Paraxial ray data

Operands,13-15

Paraxial rays

And coordinate breaks,7-5

Tracing,8-31

Paraxial surfaces,11-4

Paste

Operand data,6-5, 6-7

Surface data,6-4

Pasting

Graphics and text,2-6

Paths

For data files,5-2

PAUSE

ZPL macros,20-29

Peak to valley

Wavefront error,8-35

Pen

Changing width of,5-4

Pen color

ZPL macros,20-11

Pen colors,5-3

Pen width,5-3 - 5-4

Periodic surface,11-31

Petzval

Operand,13-15

Petzval curvature

Calculating,8-34

Petzval lens

Example of,23-5

Phantom rays,22-2

Phase

Point Spread Function,8-13

Pick up solves,12-1

On extra data values,6-7

Plot width,5-4

Plots

Changing pen width,5-4

Point spread function,8-12, 8-15

Cross sections,8-15

Problems with,22-2

Polarization

Ray trace,8-38

Reflectivity, Transmission, Absorption,8-41

ZPL macros,20-29 - 20-30

Polarization analysis,19-1

Polynomial

Extended surface type,11-20

Polynomial surfaces,11-8

Pop-up screens

Configuring,5-2

Portrait orientation,5-4

Power

Function,20-5

Maintaining with glass change,6-2

Operand,13-16

Solve,12-2

Prescription Data,10-2

Pressure

Setting,7-5

Pressure effects,18-1

Primary wavelength,7-7

Defined,3-6

Multi-configuration,16-2

Principal planes,3-1, 10-2

Printing

ASCII from ZPL macros,20-31

Changing pen widths,5-4

Coating definitions,10-3

Contents of a window,10-1

Environment options,5-4

Font size,5-4

Graphic windows,2-2

Merit function,10-2

Orientation of,5-4

Prescription data,10-2

Spreadsheet data,10-2

Surface data,10-1

System data,10-1

Text windows,2-4, 10-1

To a file,2-6

Tolerance data,10-2

Using ZPL macros,20-30

Windows in ZPL macros,20-31

- Prism
  - Example file,23-5
  - Example of,23-5
- Pupil aberration
  - Controlling,7-2
- Pupil positions
  - Displaying,10-1
- Pupil shift,7-3
- Px
  - Defined,3-4
- Py
  - Defined,3-4

## Q

---

- QSUM
  - Operand,13-16
- Quadratic sum
  - Operand,13-16
- Questions
  - About ZEMAX,22-1
- Quick focus,9-6
- Quitting ZEMAX,5-5

## R

---

- Radii
  - Entering,6-1
- Radius
  - See also Curvature
- Radius of curvature
  - Conventions for,3-6
- Ray aiming,6-2, 7-2
  - While tolerancing,15-11
  - ZPL macros,20-35
- Ray fans,8-5 - 8-7
- Ray tracing,8-31
- Rayleigh range,8-33
- Real ray
  - Tracing,8-31
- Recovering lost files,6-8
- Redo,6-8
- Reference sphere,7-4
- Referencing OPD
  - Described,7-4
- Reflections
  - Ghost,9-3
- Reflectivity
  - Of coatings,8-41
- Reflectors
  - Specifying,3-3, 6-2
- Refractive index
  - See Index of refraction
- Resolution
  - Estimating,8-22

- Resolving
  - Two field points,8-9
- Retardance
  - Of coatings,8-42
- RETURN
  - ZPL macros,20-20
- Reversing
  - Lens elements,9-3
- RGLA
  - Operand,13-24
- RMS
  - Spot size,8-17 - 8-18, 13-1
  - Wavefront error,8-17 - 8-18, 8-35, 13-1
- Rotating
  - 3D layout plots,8-3
- Rumak
  - Example of,23-5

## S

---

- Sag
  - Constraining,13-17
  - Operand,13-17
  - ZPL macros,20-6
- Sag table,8-37
- Sagittal
  - Definition of,3-6
- Sampling
  - Hexapolar rings,3-3
- Saving
  - Lens files,5-1
- Scale bar
  - On layout plots,8-1
  - Setting accurately,5-4
- Scale factor,8-2
- Scaling
  - On printers,5-4
- Scanning lens
  - Example of,23-5
- Schmidt telescope
  - Example,23-5
- Searching
  - For stock lenses,9-2
- Seidel coefficients,8-34 - 8-35
- SELFOC lenses,11-17 - 11-18, 23-5
- Semi-diameter
  - Accuracy of,7-6
- Semi-diameters
  - Accuracy of,3-7
  - Defined,3-6
  - Entering,6-2
- Set to default,13-1
- Set vignetting
  - On field dialog box,7-6

- ZPL macros,20-37
- Variables,6-3
- SETVECSIZE
  - ZPL macros,20-35 - 20-36
- Shaded model,8-4
- Short course
  - Files,23-5
- Shortcuts
  - Keyboard,2-1 - 2-2
  - Summary of,2-5
- Sign conventions
  - For binary optics,11-22
  - For radius of curvature,3-6
  - For tilted and decentered systems,4-20
- Simulated annealing,14-1
- Singlet
  - Tutorial,4-1
- Solid modeling,8-2 - 8-4
- Solves,12-1
  - Aplanatic,12-2 - 12-4
  - Chief ray angle,12-2
  - Chief ray height,12-3
  - Chief ray normal,12-2
  - Compensator,12-4
  - Edge thickness,12-3
  - Marginal ray angle,12-2
  - Marginal ray height,12-3
  - Marginal ray normal,12-2
  - Optical path difference,12-4
  - Pick up,12-2, 12-4 - 12-5, 16-3
  - Position,12-4
  - Suggestions for use,12-5
  - Thermal Pick Up,18-3
  - Turning off,12-1
- Sources
  - Extended,8-22
  - Laser diodes,21-1
- Spherical aberration
  - Calculating,8-34
  - Operand,13-17
- Spiders,6-3
- Spline
  - Bicubic using grid surface,11-29
- Spline surfaces,11-6
- Spot diagrams,8-7, 8-9
  - Full field,8-9
  - Matrix,8-9
- Spot size,8-17 - 8-18
- Spreadsheet
  - Printing Lens Data Editor contents,10-2
- Spreadsheet editor
  - Cursor controls for,6-1
  - Shortcuts,6-1
- Square root
  - Operand,13-17
- Square wave MTF
  - Operand,13-13
- Standard deviation
  - Wavefront error,8-35
- Status bar
  - Modifying,5-5
- Stock lens
  - Catalogs,23-3
- Stock lens catalog,9-2
- Stock lenses
  - Using search to find,9-2
- Stop surface
  - Defining,6-2
  - Zooming in multi-configurations,16-2
  - ZPL macros,20-38
- Strehl ratio,3-2, 8-17 - 8-18, 8-36
  - ZPL macros,20-19
- String length
  - ZPL macros,20-6
- String variables
  - ZPL macros,20-8
- Subroutines
  - ZPL macros,20-20
- Substitute
  - Glass,12-4
- Superconic
  - Surface type,11-27
- Surface apertures
  - Defined,3-7
  - Multi-configuration values,16-2
  - Specifying,6-3
- Surface data
  - Cut, Copy, and Paste,6-4
- Surface numbering,6-1
- Surface types,11-1
  - ABCD,11-9
  - Alternate intersection,11-9
  - Conic,6-2, 11-1
  - Coordinate breaks,11-8
  - Cubic spline,11-6
  - Diffraction gratings,11-10
  - Fresnel,11-9
  - Holograms,11-7
  - Plane,6-2, 11-1
  - Polynomial,11-8
  - Selecting,6-2, 11-1
- Surfaces
  - Deleting,6-1
  - Inserting,6-1
  - Miscellaneous data,10-1
  - Reversing order of,9-3, 9-6
- Symmetry
  - Default merit function,13-4

- System aperture
  - Defined,3-7
  - Multi-configuration value,16-2
  - Optimizing,22-3
  - Specifying,7-1
- System data
- ZPL Macros,20-16 - 20-17
- System menu,7-1

## T

---

- Tangnetial
  - Definition of,3-6
- TCE
  - Adding data,18-4
  - Data defined,18-3
  - In glass catalogs,17-2
- Technical support,1-2, 22-1
- Telecentric lens
  - Example of,23-4
- Telecentric systems,22-3
  - In image space,22-3
  - In object space,22-3
- Temperature
  - Setting,7-5
- Temperature effects,18-1
- Test plate fit files,8-31, 23-3
- Test plate fitting,9-5
- Text
  - Copying to clipboard,2-4
- Thermal effects,18-1
  - Auto setup,6-6
- Thickness
  - Boundary conditions on,13-8
  - Conventions for,3-7
  - Negative center,22-3
  - Of image plane,22-1
  - Operand,13-18
  - Solves on,12-3
- Thicknesses
  - Entering,6-1
- Thin films,8-41
  - See Coatings
- Thin lens
  - Cylindrical,11-4
  - Surface type,11-4
- Through focus
  - Geometric MTF,8-12
- Through-Focus
  - MTF,8-11
- Tilted surface,11-11
- Tilting
  - Object plane,23-6
- Tilts
  - Tolerancing,15-4
- Tilts and decenters
  - With coordinate break surfaces,11-8
- Time
  - Printing on plots and text,5-3
- TIR
  - Frustrated,19-7
  - See also Total Internal Reflection
- Tolerance data
  - Cut, Copy, and Paste,6-6
- Tolerancing,9-1, 15-1
  - Debugging,15-15
  - Defining compensators for,15-9
  - Irregularity,15-4
- Toolbar
  - Defining,5-5
- Tools menu,9-1
- Toroidal lens
  - Example of,23-6
  - Paraxial model for,11-4
- Toroidal surfaces,11-5, 11-31
  - Gratings placed on,11-6
  - Thermal effects,18-3
- Total Indicator Runout
  - Tolerancing,15-4
- Total Internal Reflection
  - ZPL Macros,20-32
- Total track
  - Defined,3-7
  - Operand,13-18
- Transfer Functions,8-11 - 8-12, 8-15 - 8-17
- Transmission,8-39 - 8-40
  - Internal,8-30
  - Of coatings,8-41
- Transmission data
  - Glass catalogs,17-4
- Transverse aberrations
  - Calculating,8-34
- Transverse color
  - Calculating,8-34
- Tutorial
  - Chapter on,4-1

## U

---

- Undo,6-8
  - Disabling,5-4
- Uniform distribution,15-14
- Units
  - Determining in ZPL macros,20-6
  - ZPL macros,20-36
- Update
  - Described,7-1
- Update all

- Described,7-1
- Upgrades
  - Policy on,1-3
- User defined surface
  - Diffractive optics,11-35
  - Gradient index,11-35
  - Lenslet arrays,11-36
  - Refractors and reflectors,11-35
  - Using linked code,11-34
  - Using look up table,11-29 - 11-30, 23-2

## V

---

- Vacuum
  - Index of refraction,18-1
  - Modeling,18-1
- Variable line space gratings,11-26
- Variables
  - Defining within ZPL macros,20-36
  - Removing,9-6
- Variance
  - Wavefront error,8-35
- Vector variables
  - Setting the size of,20-36
- Vignetting,6-3, 8-6, 8-25, 22-1
  - Program ignores,22-1
  - ZPL macros,20-32, 20-37
- Vignetting factors
  - Adjusting during optimization,13-17
  - Defined,3-7
  - Field curvature and distortion,8-27
  - Multi-configuration values,16-1
  - Setting,7-6
- Volume
  - Of an element,10-1, 13-6
  - Operand,13-18

## W

---

- Wavefront aberrations,8-34
- Wavefront error,8-17 - 8-18
  - Peak to valley,8-35
  - RMS,8-35
- Wavelengths
  - Conventions for,3-8
  - Defining,4-1
  - Optimizing,22-3
  - Weights for,7-7
  - ZPL macros,20-26
- Wedge
  - Example of,23-5
- Weight
  - Configuration,16-2
  - Of lenses,10-2
- Wide angle apertures

- In object space,7-1
- Wide angle lens
  - Example of,23-6
- Window size
  - Changing default,5-2 - 5-3
- Windows
  - Graphic,2-1
  - Resizing and moving,2-1
  - Shortcuts,2-5
  - Text,2-1
  - Types of,2-1
- Wire frame model,8-4
- Wolter type telescopes
  - Examples of,23-6
- Working F/#,7-1
  - Defined,3-8
  - Operand,13-18
  - Paraxial,3-6

## Y

---

- Y-Ybar diagrams,8-29
- YNU tracing,8-31
- Yolo telescope
  - Example of,23-6

## Z

---

- ZEMAX Programming Language
  - See ZPL macros
- Zernike
  - Deformed surfaces,11-18 - 11-19
- Zernike coefficients
  - Calculation of,8-35
  - Optimization of,13-6, 13-19
  - Table of,8-36
- Zernike phase surface
  - Optimizing coefficients,11-20
- Zernike surface
  - Optimizing coefficients,11-19 - 11-22, 11-24 - 11-25
- ZMX format,25-1
- Zone plates,11-33 - 11-34
- Zoom and pan,2-4
- Zoom lenses
  - Eliminating variables,21-3
  - Example lens file,23-6
  - Maintaining length of,12-4
  - Maximum semi-diameter solve,12-5
  - Multi-configuration feature,16-1
  - Operand,13-19
  - Phantom elements,16-3
- Zoom positions
  - Changing,7-7
  - On plots,5-2
  - Showing number on plots,5-2



ZPL macros,9-5, 20-1  
Editing,20-1  
Examples,20-41  
Optimization of,13-26  
Printing,20-1  
Status message,20-1  
String variables,20-8