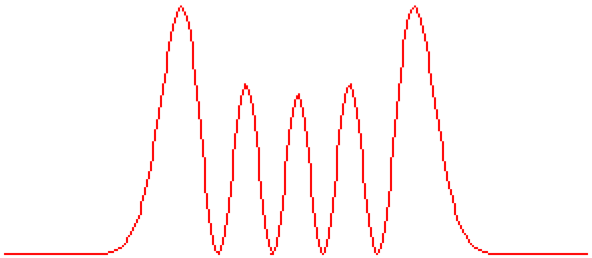
EP547 Computational Methods in QM

Topic 8

Roots of Equations & Optimisation



Department of Engineering Physics

University of Gaziantep

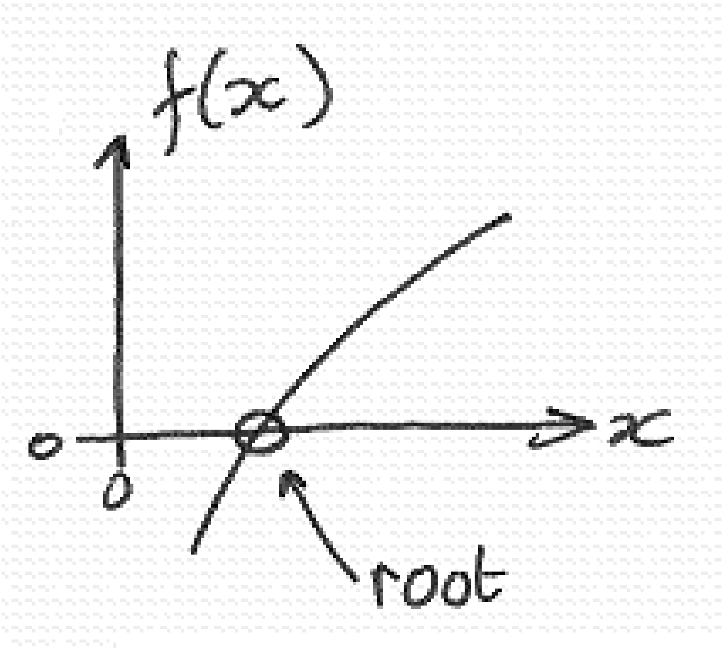
Feb 2013

Content

1. MATLAB **`solve()`** Function
2. MATLAB **`fminsearch()`** Function

Root Finding

The root x_0 of a function f(x) is such that $f(x_0)=0$.



Roots of Polynomials

- $x^2 - 3x + 2 = 0$ (roots are: 2 and 1)

```
>> roots([1 -3 2])
ans =
      2
      1
```

- $x^5 + 2x^4 - 5x^3 + x + 3 = 0$

```
>> c = [1 2 -5 0 1 3];
>> roots(c)
ans =
  -3.4473
   1.1730 + 0.3902i
   1.1730 - 0.3902i
  -0.4494 + 0.6062i
  -0.4494 - 0.6062i
```

MATLAB solve() Function

We can use **`solve()`** function in MATLAB.

```
>> solve('x^2-4=0')
ans =
      2
     -2
```

```
>> solve('sin(x)+2-x=0')
ans =

2.5541959528370430378296661737919
```

```
>> solve('3*sin(x)+2-x=0')


ans =
-1.2467199751961775376089438821225-
.52620368582988923192255563579236*i
```

```
>> solve('a*x^2 + b*x + c')

ans =

 1/2/a*(-b+(b^2-4*a*c)^(1/2))
 1/2/a*(-b-(b^2-4*a*c)^(1/2))
```

```
>> solve('a*x^2 + b*x + c','b')

ans =

-(a*x^2+c)/x
```

```
>> solve('a*x^2 + b*x + c','c')

ans =

-a*x^2-b*x
```

```
>> s = solve('x + y = 1','x - 11*y = 5')

s =
     x: [1x1 sym]
     y: [1x1 sym]



>> s.x
ans =
4/3



>> s.y
ans =
-1/3
```

```
>> s =solve('x^2+y^2-4=0','x*y-1=0')

s =
     x: [4x1 sym]
     y: [4x1 sym]

>> s.x
ans =
   -(1/2*6^(1/2)+1/2*2^(1/2))^3+2*6^(1/2)+2*2^(1/2)
   -(1/2*6^(1/2)-1/2*2^(1/2))^3+2*6^(1/2)-2*2^(1/2)
  -(-1/2*6^(1/2)+1/2*2^(1/2))^3-2*6^(1/2)+2*2^(1/2)
  -(-1/2*6^(1/2)-1/2*2^(1/2))^3-2*6^(1/2)-2*2^(1/2)

>> s.y
ans =

   1/2*6^(1/2)+1/2*2^(1/2)
   1/2*6^(1/2)-1/2*2^(1/2)
  -1/2*6^(1/2)+1/2*2^(1/2)
  -1/2*6^(1/2)-1/2*2^(1/2)
```

Example 1:

The speed v of a rocket in vertical flight near the surface of earth can be approximated by

$$v = u \ln\left(\frac{m_0}{m_0 - \dot{m}t}\right) - gt$$

where

u = velocity of exhaust relative to the rocket (2500 m/s)

m_0 = mass of rocket at liftoff (3.0e6 kg)

m-dot = rate of fuel consumption = 13.0e4 kg/s

g = 9.8 m/s^2

t = time measured from liftoff

Determine the time when the rocket reaches the speed of sound (340 m/s).

```
>> solve('340=2500*log(3e6/(3e6-13e4*t))-9.8*t','t')

ans =

  3.1842633689729853500532750649539
 -1002.6281724348762640806634674228
```

So, within 3.2 seconds the rocket reaches the speed of sound!

Example 2:

Find the solution of

$$\sin x + y^2 + \ln z - 7 = 0$$

$$3x + 2^y - z^3 + 1 = 0$$

$$x + y + z - 5 = 0$$

```
>> coz = solve('sin(x) + y^2 + log(z)-7=0',...
               '3*x + 2^y - z^3 + 1=0',...
               'x + y + z - 5=0')
>> coz.x
ans =
.59905375664056731520568183824539

>> coz.y
ans =
2.3959314023778168490940003756591

>> coz.z
ans =
2.0050148409816158357003177860955
```

Optimisation

- Optimization is the term used for minimizing or maximizing a function.

- In general, it is sufficient to consider the problem of minimization only; maximization of $f(x)$ is achieved by simply minimizing $-f(x)$.

- The function $f(x)$ that we want to optimze is called the **merit function** or **objective function**.

MATLAB `fminsearch()` Function

`x = fminsearch(@func, x0)`

returns the vector of independent variables that minimizes the multivariate function **`func`**.

The vector **`x0`** contains the starting values of **`x`**.

```
>> xopt = fminsearch(@sin,1)

xopt =

   -1.5708
```

Example 3:

Locate the minimum the function f(x) = exp(x)/x

optfun.m

```
function y = optfun(x)
  y = exp(x)/x;
end
```

```
>> xmin = fminsearch(@optfun, 0.4);
xmin = 1.0000
```

```
>> [xmin, fmin] = fminsearch(@optfun, 0.4);
xmin = 1.0000
fmin = 2.7183
```

Example 4:

Locate the minimum of

$$f(x, y) = 10x^2 + 3y^2 - 10xy + 2x.$$

Start with: $(x_0 , y_0) = (0, 0)$.

optfun.m

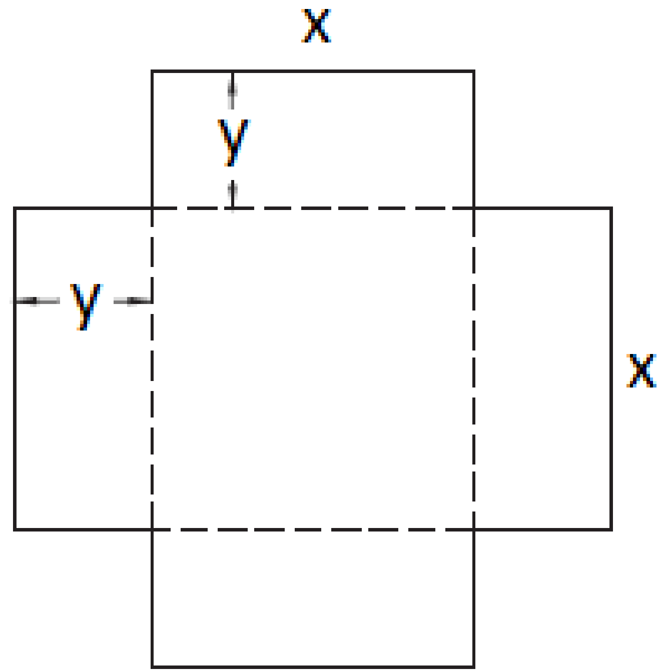
```
function y = optfun(x)
 y = 10*x(1)^2 + 3*x(2)^2 - 10*x(1)*x(2) + 2*x(1);
end
```

```
>> x = fminsearch(@optfun, [0 0]);
x =

   -0.6000   -1.0000
```

Example 5:

Consider a box with open top to carry V = 0.2 m^3 waste water. The cost of material used to form the box is C_m = 10 TL/m^2 and welding cost is C_w = 5 TL/m. Design the box so that its total cost is minimum. Verify the result analytically.



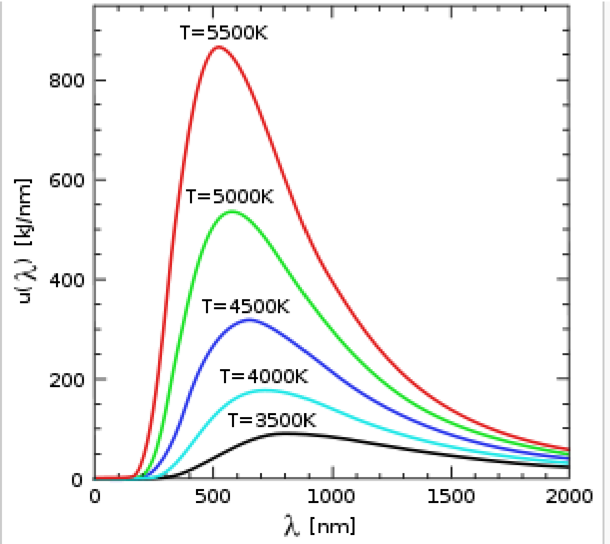
Example 6:

Using Plank's formula for a black-body radiator, derive Wein law:

$$k_B T \lambda_{\max} = 0.2014$$

or

$$\lambda_{\max} T = 0.0029 \;\; \mathrm{m \cdot K}$$



This diagram shows how the peak wavelength and total radiated amount vary with temperature according to Wien's displacement law. Although this plot shows relatively high temperatures, the same relationships hold true for any temperature down to absolute zero. Visible light is between 380 and 750 nm.

Hint: Plank formula is given by:

$$u(\lambda) = \frac{8\pi hc}{\lambda^5} \frac{1}{\exp(hc / k_B T\lambda) - 1}$$

use dimensionless variable: $x = \dfrac{hc}{k_B T \lambda}$

and solve $\dfrac{du}{dx} = 0$

http://en.wikipedia.org/wiki/Thermal_radiation

References

[1]. http://www.mathworks.com/products/matlab

[2]. Numerical Methods in Engineering with MATLAB,
J. Kiusalaas, Cambridge University Press (2005)

[3]. Numerical Methods for Engineers, 6th Ed.
S.C. Chapra, Mc Graw Hill (2010)