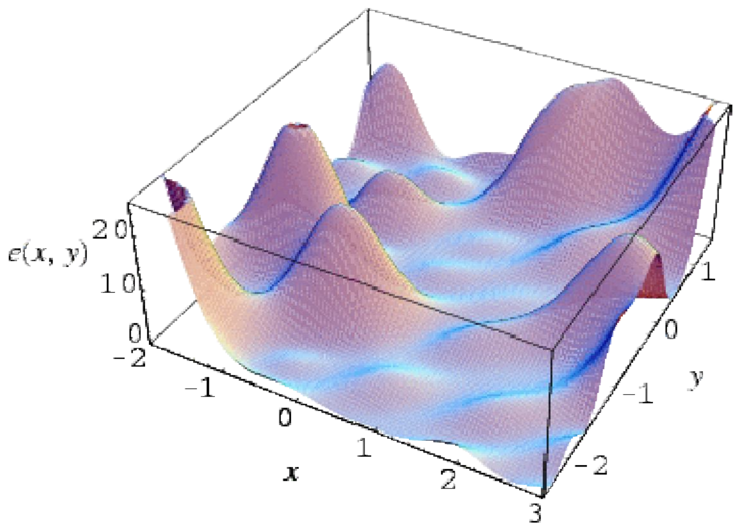
Chapter 7

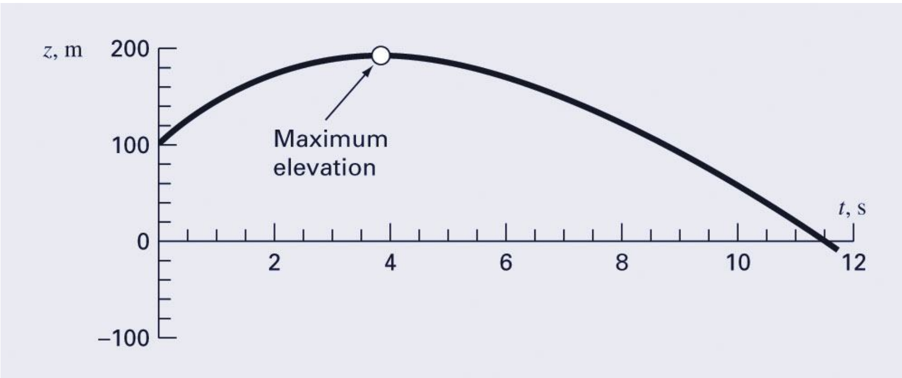
Optimization

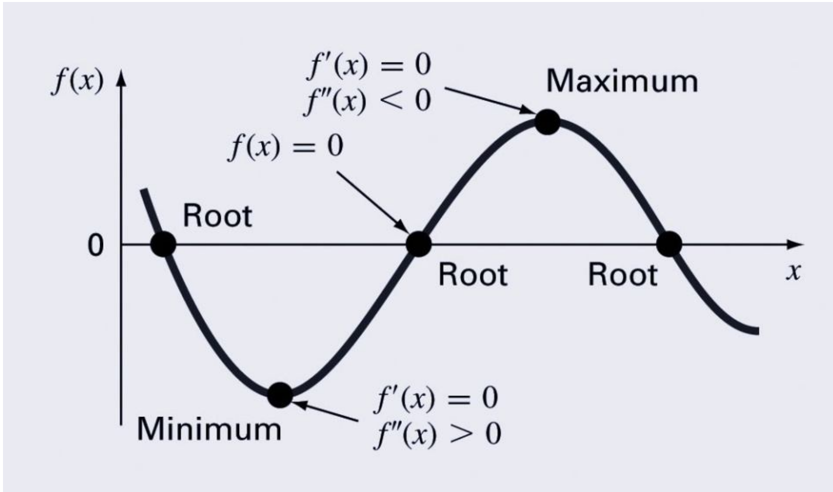


Optimization

- Optimization is the process of creating something that is as effective as possible.

- From a mathematical perspective, optimization deals with finding the maxima and minima of a function that depends on one or more variables.

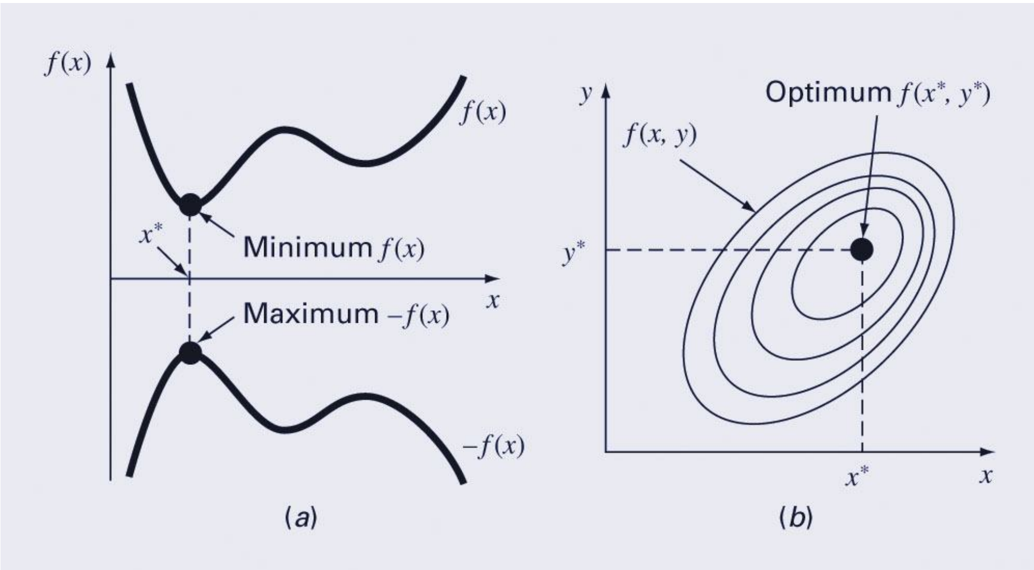


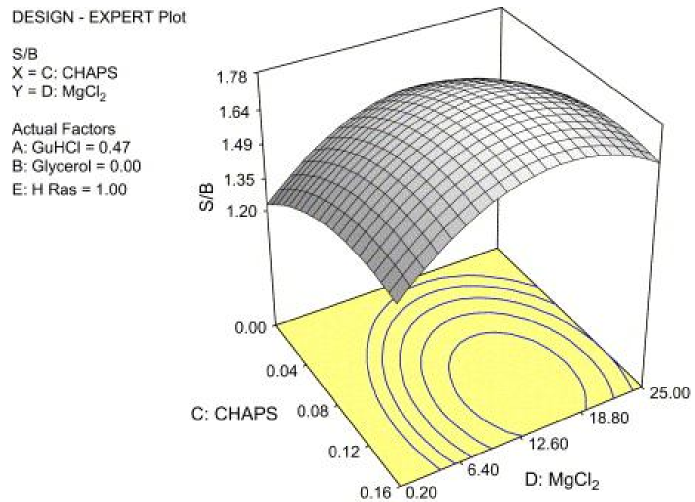


Multidimensional Optimization

- One-dimensional problems involve functions that depend on a single dependent variable : *f* (*x*)

- Multidimensional problems involve functions that depend on two or more dependent variables : *f* (*x, y*)

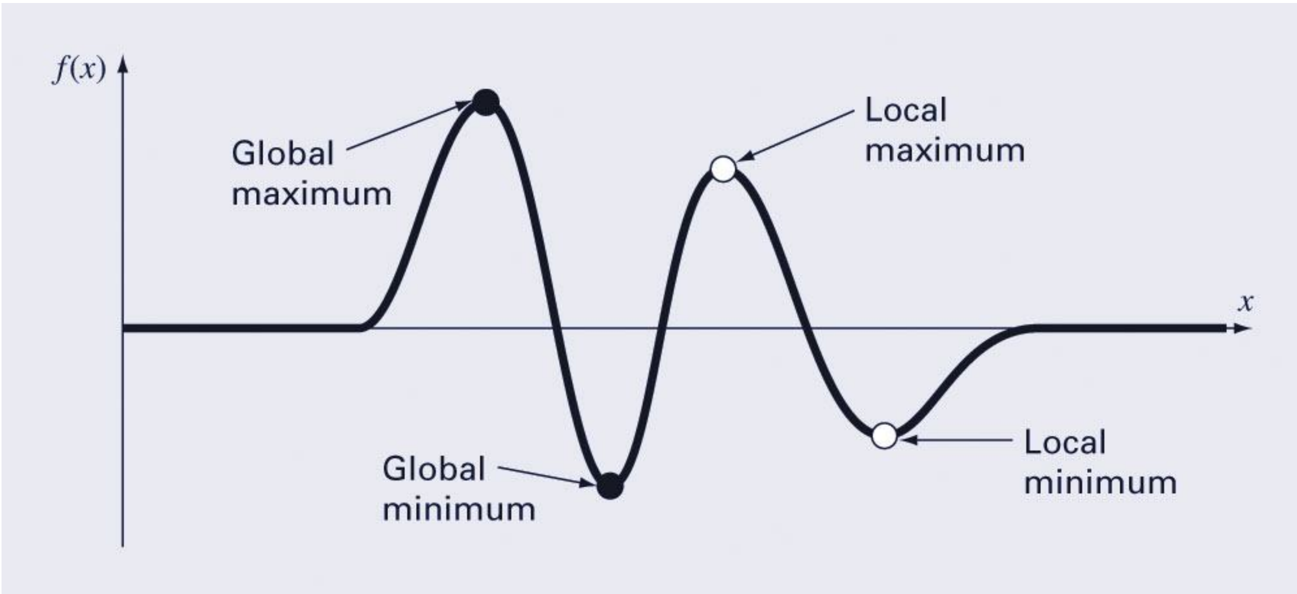




Global vs. Local

- A global optimum represents the very best solution while a local optimum is better than its immediate neighbors.
  - Cases that include local optima are called multimodal.

- Generally desire to find the global optimum.



Golden-Section Search

- Search algorithm for finding a min/max on an interval $[x_{L,}\ x_u]$ with a single min/max (unimodal interval)

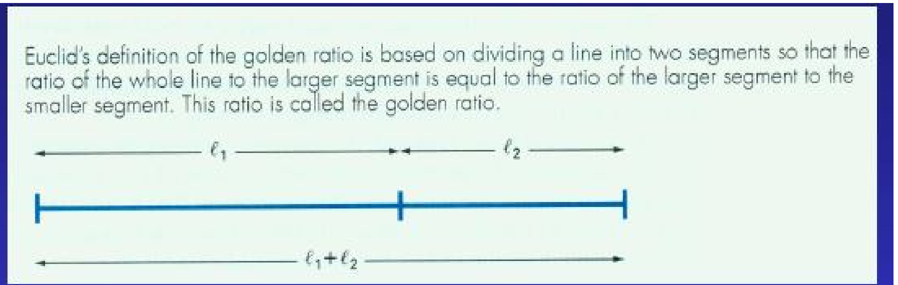
- Uses the *golden ratio* ϕ*=1.6180…* to determine location of two interior points x_1 and x_2
  - by using the golden ratio, one of the interior points can be re-used in the next iteration.

$$\frac{\ell_1+\ell_2}{\ell_1}=\frac{\ell_1}{\ell_2}$$

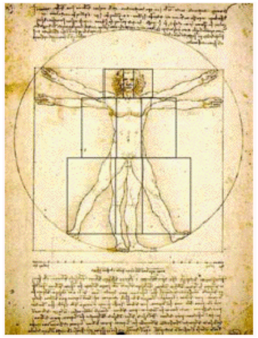
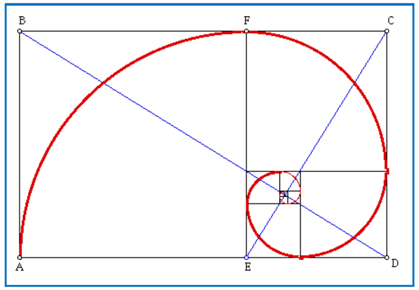
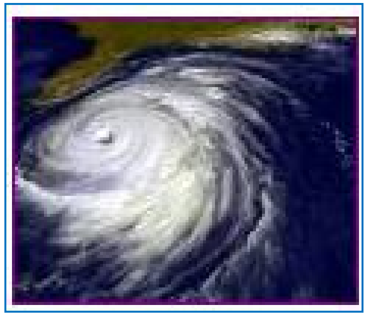
$$\text{Let } \varphi=\frac{\ell_1}{\ell_2}$$

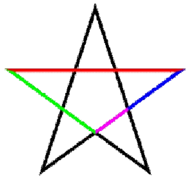
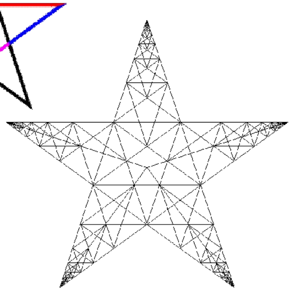
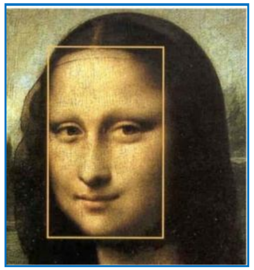
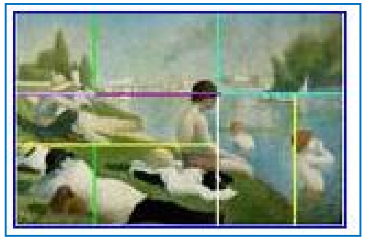
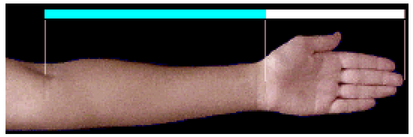
$$\text{Then, } \varphi^2-\varphi-1=0$$

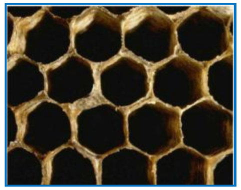
$$\varphi=\frac{1+\sqrt{5}}{2}=1.6180339...$$



Golden Ratio Examples

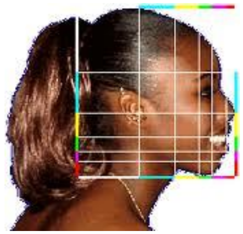



The Secret of the Universe : pentagon (Pythagoras 학파의 symbol**)**

❑ Fibonacci Numbers

0, 1, 1, 2, 3, 5, 8, 13, 21, 34, …..

한 옥타브 피아노 건반의 수는 13
(흰 건반 8개, 검은 건반 5개),
검은 건반은 다시 3개와 2개로 나뉨 → 2,3,5,8,13









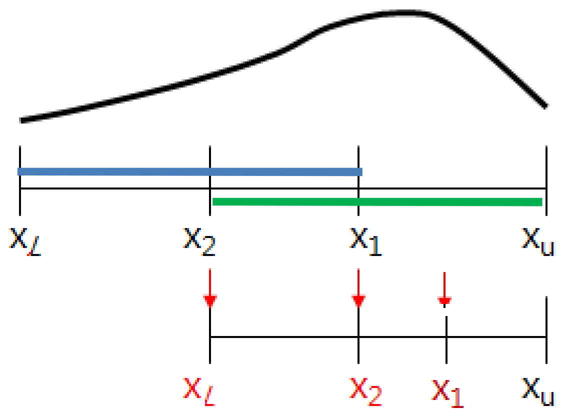
Golden-Section Search

1. Guess x_L , x_u *(Assume max problem)*

2. Estimate d = $R^*(x_u - x_L) = 0.618^*(x_u - x_L)$

3. $x_1 = x_L + d$, $x_2 = x_u - d$, $f(x_1)$, $f(x_2)$ watch out $x_1 > x_2$

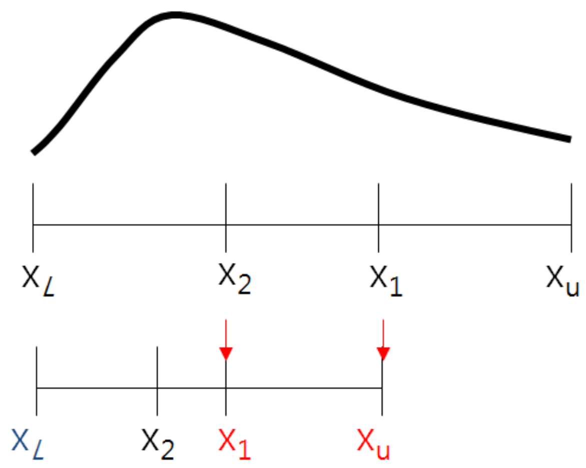
4. If $f(x_1) > f(x_2)$, eliminate the domain x



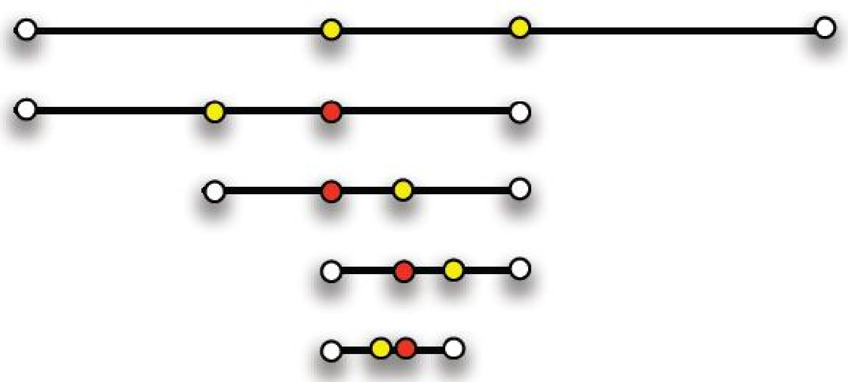
- x_L = old x_2
- x_2 = old x_1
- x_u = old x_u
- $d = R^*(x_u - x_L)$
- $x_1 = x_L + d$

5. If $f(x_1) < f(x_2)$, eliminate the domain $x_1 \sim x_u$

- x_u = old x_1
- x_1 = old x_2
- x_L = old x_L
- $d = R(x_u - x_L)$
- $x_2 = x_u - d$



6. Repeat step 2~5 until converge



Convergence test

$$e_a = (1 - R) \times \frac{|x_u - x_l|}{x_{opt}} \times 100 \le e_s$$

if $(x_1 - x_L) <= \delta$ (절대오차) , stop

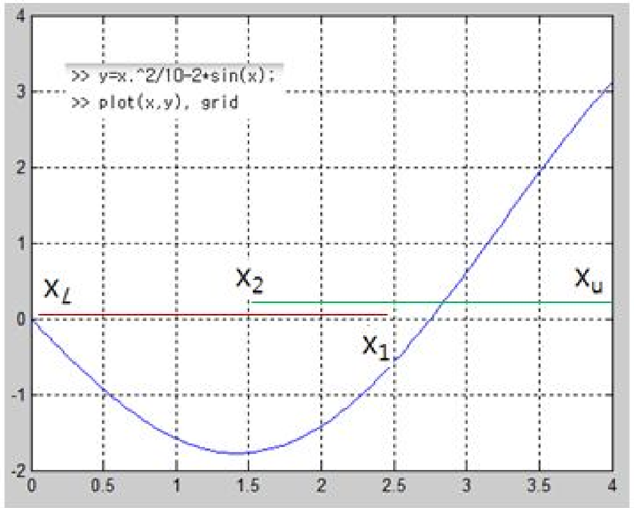
$\delta = \varepsilon_a * x_{opt}$

Example 7.2

- Find the minimum of $f(x)=x^2/10 - 2*\sin(x)$, [0,4]

$$d = 0.61803(4-0) = 2.4721$$
$$x_1 = 0 + 2.4721 = 2.4721$$
$$x_2 = 4 - 2.4721 = 1.5279$$

$$f(x_2) = \frac{1.5279^2}{10} - 2\sin(1.5279) = -1.7647$$
$$f(x_1) = \frac{2.4721^2}{10} - 2\sin(2.4721) = -0.6300$$



- $f(x_2) < f(x_1)$ (min is between x_L , x_2, x_1) : eliminate $[x_1, x_U]$
  - $(x_u \leftarrow x_1)$ $(x_1 \leftarrow x_2)$ → new d → (new $x_2 = x_u - d$)
  - $[\ x_L,\ x_2,\ x_1,\ x_U]$: [0, 1.5729, 2.4721, 4.0] → [0, 0.9443, 1.5279, 2.4721]

Example 7.2

$d = R(x_u - x_L) = 0.618*(x_u - x_L)$

i	x_l	$f(x_l)$	x_2	$f(x_2)$	x_1	$f(x_1)$	x_u	$f(x_u)$	d
1	0	0	1.5279	-1.7647	2.4721	-0.6300	4.0000	3.1136	2.4721
2	0	0	0.9443	-1.5310	1.5279	-1.7647	2.4721	-0.6300	1.5279
3	0.9443	-1.5310	1.5279	-1.7647	1.8885	-1.5432	2.4721	-0.6300	0.9443
4	0.9443	-1.5310	1.3050	-1.7595	1.5279	-1.7647	1.8885	-1.5432	0.5836
5	1.3050	-1.7595	1.5279	-1.7647	1.6656	-1.7136	1.8885	-1.5432	0.3607
6	1.3050	-1.7595	1.4427	-1.7755	1.5279	-1.7647	1.6656	-1.7136	0.2229
7	1.3050	-1.7595	1.3901	-1.7742	1.4427	-1.7755	1.5279	-1.7647	0.1378
8	1.3901	-1.7742	1.4427	-1.7755	1.4752	-1.7732	1.5279	-1.7647	0.0851

(cf) true min. is -1.7757 at x=1.4276

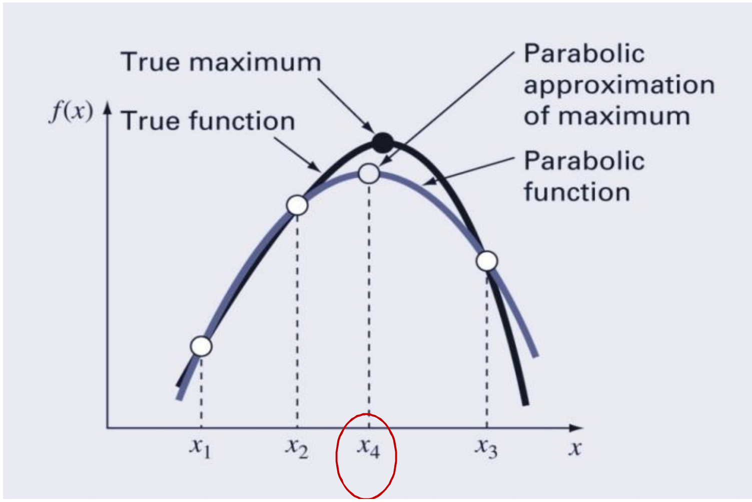
Parabolic Interpolation

- Another algorithm uses parabolic interpolation of three points to estimate optimum location. ($y=ax^2+bx+c$)

- The maximum/minimum of a parabola ($y'=0$) defined as the interpolation of three points (x_1, x_2, and x_3) is

$$x_4 = x_2 - \frac{1}{2}\frac{(x_2-x_1)^2[f(x_2)-f(x_3)]-(x_2-x_3)^2[f(x_2)-f(x_1)]}{(x_2-x_1)[f(x_2)-f(x_3)]-(x_2-x_3)[f(x_2)-f(x_1)]}$$

- The new point x_4 and the two surrounding it (either x_1 and x_2 or x_2 and x_3) are used for the next iteration of the algorithm.



Example 7.2

- Find the minimum of $f(x)=x^2/10 - 2\sin x$, $[x_1=0, x_2=1, x_3=4]$

$$
\begin{aligned}
x_1 &= 0 & f(x_1) &= 0 \\
x_2 &= 1 & f(x_2) &= -1.5829 \\
x_3 &= 4 & f(x_3) &= 3.1136
\end{aligned}
$$

$$x_4 = 1 - \frac{1}{2}\frac{(1-0)^2[-1.5829-3.1136]-(1-4)^2[-1.5829-0]}{(1-0)[-1.5829-3.1136]-(1-4)[-1.5829-0]} = 1.5055$$

$$\boxed{f(1.5055) = -1.7691}$$

- [0,1,1.5055,4]

$$
\begin{aligned}
x_1 &= 1 & f(x_1) &= -1.5829 \\
x_2 &= 1.5055 & f(x_2) &= -1.7691 \\
x_3 &= 4 & f(x_3) &= 3.1136
\end{aligned}
$$

$$x_4 = 1.5055 - \frac{1}{2}\frac{(1.5055-1)^2[-1.7691-3.1136]-(1.5055-4)^2[-1.7691-(-1.5829)]}{(1.5055-1)[-1.7691-3.1136]-(1.5055-4)[-1.7691-(-1.5829)]}$$

i	x_1	$f(x_1)$	x_2	$f(x_2)$	x_3	$f(x_3)$	x_4	$f(x_4)$
1	0.0000	0.0000	1.0000	-1.5829	4.0000	3.1136	1.5055	-1.7691
2	1.0000	-1.5829	1.5055	-1.7691	4.0000	3.1136	1.4903	-1.7714
3	1.0000	-1.5829	1.4903	-1.7714	1.5055	-1.7691	1.4256	-1.7757
4	1.0000	-1.5829	1.4256	-1.7757	1.4903	-1.7714	1.4266	-1.7757
5	1.4256	-1.7757	1.4266	-1.7757	1.4903	-1.7714	1.4275	-1.7757

true min. is -1.7757
at x=1.4276

fminbnd : Optimization in one variable

- MATLAB has a built-in function, fminbnd, which combines the golden-section search and the parabolic interpolation.
  - [xmin, fval] = fminbnd(function, x1, x2)

```
>> g=9.81; v0=55; m=80;c=15;z0=100;
>> z=@(t) -(z0+m/c*(v0+m*g/c)*(1-exp(-c/m*t))-m*g/c*t);
>> [x,f]=fminbnd(z,0,8)
x =
    3.8317
f =
  -192.8609
```

```
>> options = optimset ('display', 'iter') ;
>> fminbnd(z, 0, 8, options)
```

```
Func-count     x          f(x)        Procedure
    1       3.05573    -189.759       initial
    2       4.94427    -187.19        golden
    3       1.88854    -171.871       golden
    4       3.87544    -192.851       parabolic
    5       3.85836    -192.857       parabolic
    6       3.83332    -192.861       parabolic
    7       3.83162    -192.861       parabolic
    8       3.83166    -192.861       parabolic
    9       3.83169    -192.861       parabolic
Optimization terminated:
 the current x satisfies the termination criteria using
OPTIONS.TolX of 1.000000e-004

ans =
    3.8317
```

fminsearch : Multidimensional optimization

- [*xmin*, *fval*] = fminsearch(*function*, *x0*)

- The function must be written in terms of a single variable, where different dimensions are represented by different indices of that variable.

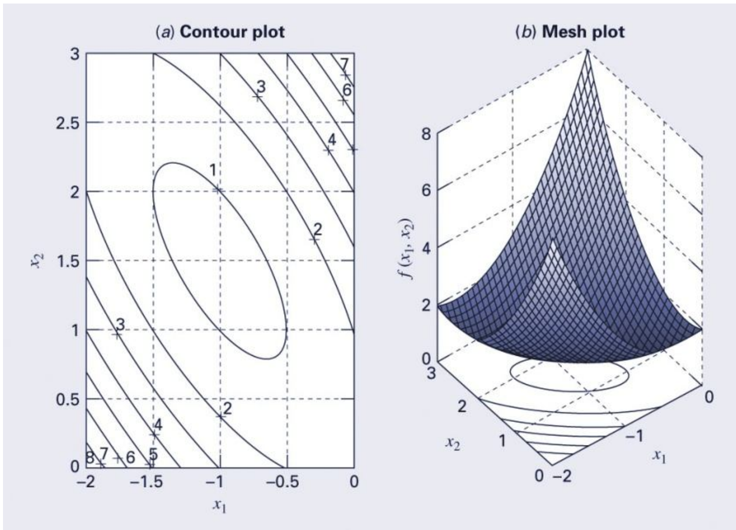
  – To minimize $f(x,y)=2+x-y+2x^2+2xy+y^2$

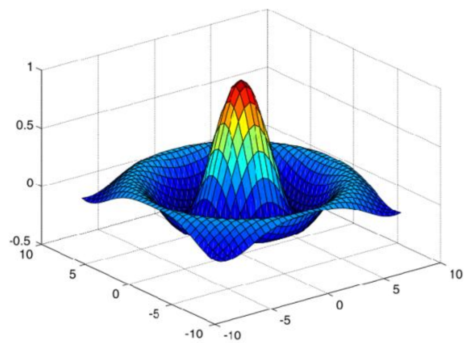
  – Rewrite as $f(x(1), x(2))=2+x(1)-x(2)+2*(x(1))^2+2*x(1)*x(2)+(x(2))^2$

```
>> f=@(x) 2+x(1)-x(2)+2*x(1)^2+2*x(1)*x(2)+x(2)^2;
>> [x, fval]=fminsearch(f, [-0.5, 0.5])
x =
    -1.0000   1.5000
fval =
    0.7500
```

Multidimensional Visualization

- MATALB 2-dimension plot commands
  - contour
  - Mesh
    - mesh(X, Y, Z) : draws a wireframe mesh with color determined by Z so color is proportional to surface height
    - surfc(X,Y, Z)) : draws a contour plot beneath the mesh
  - Surf
    - surf(X,Y, Z) : creates a a three-dimensional shaded surface
    - surfc(X,Y, Z)) : draws a contour plot beneath the surface

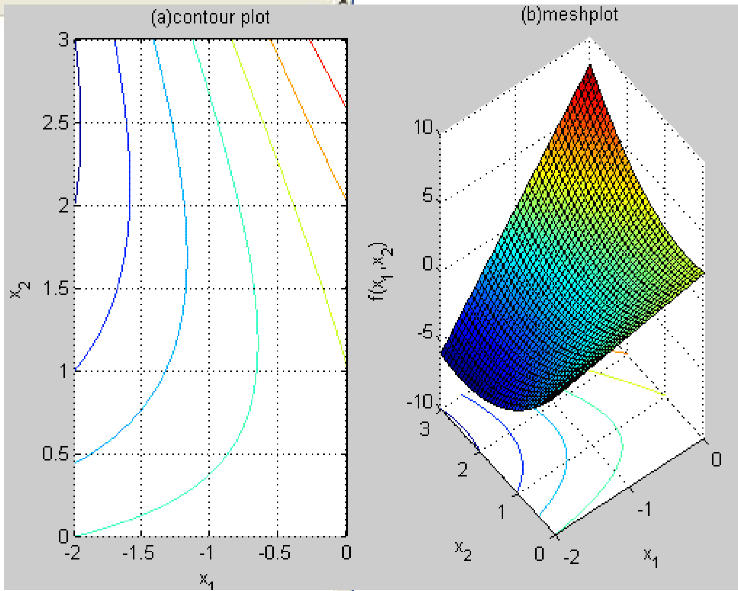




$$f(x_1, x_2) = 2 + x_1 - x_2 + 2x_1^2 + 2x_1x_2 + x_2^2$$

$-2 \le x_1 \le 0, \quad 0 \le x_2 \le 3$

```
x=linspace(-2,0,40);y=linspace(0,3,40);
[X,Y] = meshgrid(x,y);
Z=2+X-Y+2*X.^2+2*X.*Y+Y.^2;
subplot (1,2,1);
cs=contour(X,Y,Z); clabel(cs);
xlabel('x_1'); ylabel('x_2');
title('(a) Contour plot') ;grid;
subplot(1,2,2);
cs=surfc(X,Y,Z);
zmin=floor(min(Z));
zmax=ceil(max(Z));
xlabel('x_1'); ylabel ('x_2'); zlabel('f(x_1,x_2)');
title ('(b) Mesh plot') ;
```



subplot (nrows,ncols,plot_number) : display multiple plots in the same window
clabel(C) : adds labels to the current contour plot
Plot command allow the use of underscore '_' and cap '^' to be used as subscript and superscript command, respectively.
floor(x) : closest integer less than or equal to x (floor(-3.1)=-4, floor(2.9)=2)
ceil(x) : closest integer greater than or equal to x (ceil(-3.1)=-3, ceil(2.9)=3)
round(x) : closest integer (round(-1.9)=-2.0)
fix(x) : closest integer toward zero (fix(-1.9)=-1.0)

THE END

Homework : MATLAB 예제 7.1, m-file 실행

Report : 연습문제 7.7, 7.24