Bölüm 8

Sınırsız Çok Değişkenli Optimizasyon

Gradient and conjugate direction methods

Newton's method

Quasi-Newton methods

Gradient yöntemi

(1) Hesapla $\underline{s}^k$

(2) *f(x) i azaltacak yönde hesap uzunluğunu seç*

$$\underline{x}^{k+1} = \underline{x}^k + \alpha \underline{s}^k = \underline{x}^k + \Delta \underline{x}^k$$

<u>Dik iniş yöntemi</u>

arama yönü

$$\underline{s}^k = -\nabla f(\underline{x}^k)$$

Yöntem durdurulur. Herhangi bir sabit noktada

$$\nabla f(\underline{x}) = \underline{0}$$

$\underline{\underline{H}}(\underline{x}^*)$ Hessien matrisin positif tanımlı olması gerekir x minimum ise

<u>Hesap uzunluğu (artırım)</u>

α

- analitik
- sayısal

Analatik hesaplama :

$$\Delta \underline{x}^k = \alpha \underline{s}^k.$$

$$f(\underline{x}^k + \alpha \underline{s}^k) = f(\underline{x}^{k+1}) \cong f(\underline{x}^k) + \nabla^T f(\underline{x}^k)(\Delta \underline{x}^k) + \frac{1}{2}(\Delta \underline{x}^k)^T \underline{\underline{H}}(\underline{x}^k)(\Delta \underline{x}^k)$$

$$\frac{df(\underline{x}^k + \alpha \underline{s}^k)}{d\alpha} = 0 = \nabla^T f(\underline{x}^k)(\underline{s}^k) + \alpha(\underline{s}^k)^T \underline{\underline{H}}(\underline{x}^k)(\underline{s}^k)$$

α **için çözüm;**

$$\alpha = -\frac{\nabla^T f(\underline{x}^k)(\underline{s}^k)}{(\underline{s}^k)^T \underline{\underline{H}}(\underline{x}^k)(\underline{s}^k)}$$

Sayısal Hesaplama

α (α = 1) yada α (α = 1, 2, ½, v.b) devam et

$\underline{\alpha}$

(1) quadratic, cubic interpolasyon kullan
(2) (Golden Search)
(3) Newton, Secant, Quasi-Newton
(4) Random
(5) Analytical optimization

(1), (3), (5) tercih edilir.

Durdurma kriteri

(1)

$$\frac{f(\underline{x}^k) - f(\underline{x}^{k+1})}{f(\underline{x}^k)} < \in_1$$

dışında $f(\underline{x}^k) \to 0$

kullan $f(\underline{x}^k) - f(\underline{x}^{k+1}) < \in_2$

(2)

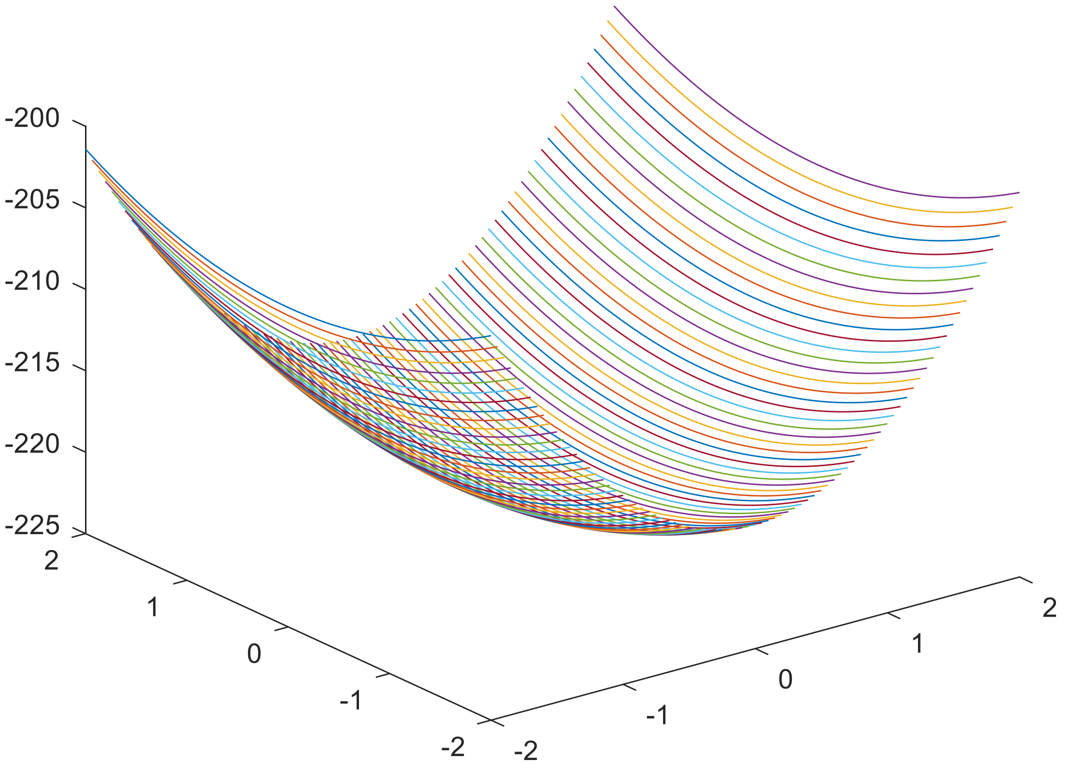
$$\left| \frac{x_i^{k+1} - x_i^k}{x_i^k} \right| < \in_3$$

dışında $x^k \to 0$

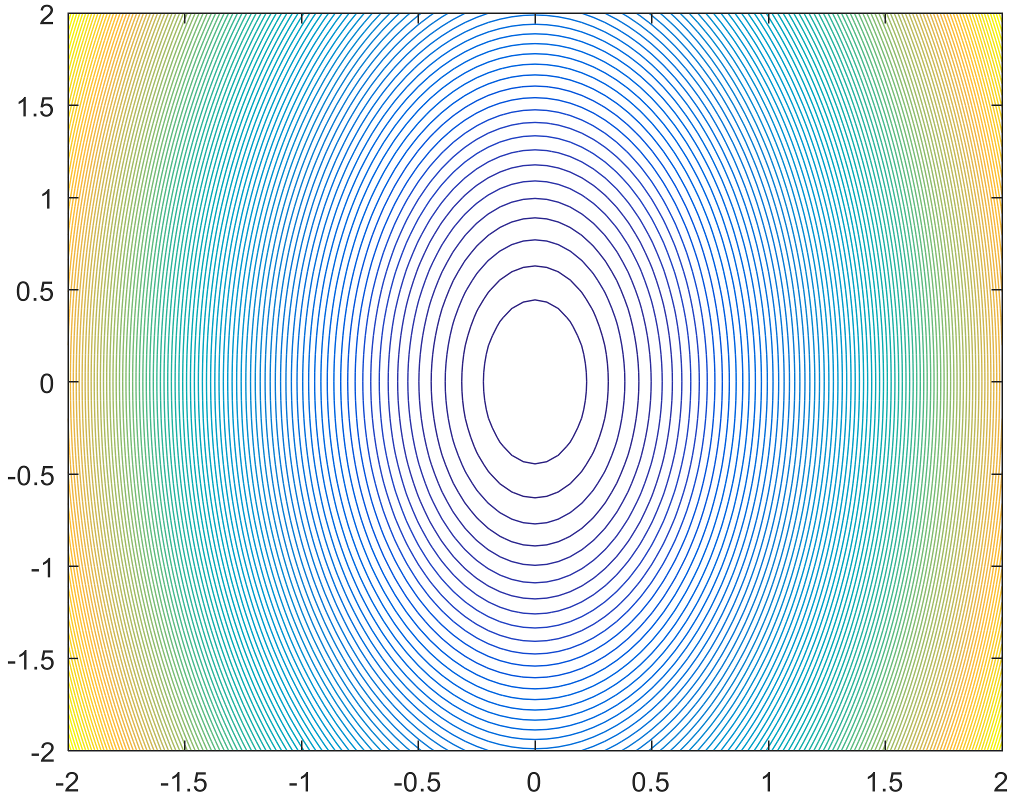
kullan: $\left| x^{k+1} - x^k \right| < \in_4$

(3) $\nabla f(x^k) < \in_5$ **ve** $\left| s_i^k \right| < \in_6$

Örnekler:



v = 4*x.^2 + y.^2 - 2*x*y;



v = 4*x.^2 + y.^2 - 2*x*y;

Fonksiyonunun kontur grafiği

```
function f=fun208(x);
f=4*x(1).^2+x(2).^2-2*x(1)*x(2);

end
>> x0=[0  0];
>> x=fminunc('fun208',x0)

x =

     0     0
```

We minimize the function

$$f(\mathbf{x}) = 4x_1^2 + x_2^2 - 2x_1x_2$$

starting at $\mathbf{x}^0 = [1 \quad 1]^T$

$$\nabla f(\mathbf{x}) = \begin{bmatrix} 8x_1 & -2x_2 \\ 2x_2 & -2x_1 \end{bmatrix}$$

$$\mathbf{H}(\mathbf{x}) = \begin{bmatrix} 8 & -2 \\ -2 & 2 \end{bmatrix} \qquad \mathbf{H}^{-1}(\mathbf{x}) = \begin{bmatrix} \frac{1}{6} & \frac{1}{6} \\ \frac{1}{6} & \frac{2}{3} \end{bmatrix}$$

with $\alpha = 1$,

$$\Delta\mathbf{x}^0 = -\mathbf{H}^{-1}\,\nabla f(\mathbf{x}^0) = -\begin{bmatrix} \frac{1}{6} & \frac{1}{6} \\ \frac{1}{6} & \frac{2}{3} \end{bmatrix}\begin{bmatrix} 6 \\ 0 \end{bmatrix} = \begin{bmatrix} -1 \\ -1 \end{bmatrix}$$

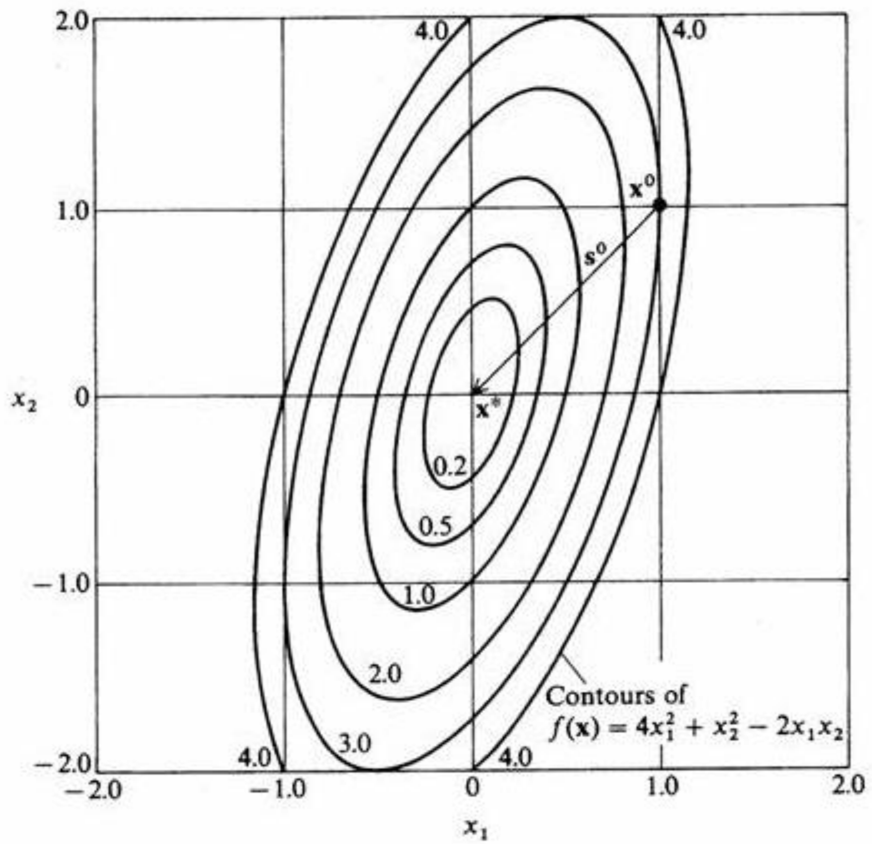
hence,

$$\mathbf{x}^1 = \mathbf{x}^* = \mathbf{x}^0 + \Delta\mathbf{x}^0 = \begin{bmatrix} 1 \\ 1 \end{bmatrix} + \begin{bmatrix} -1 \\ -1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$f(\mathbf{x}^*) = 0$$

Instead of taking the inverse of **H**, we can solve Equation (6.15)

$$\begin{bmatrix} 8 & -2 \\ -2 & 2 \end{bmatrix}\begin{bmatrix} \Delta x_1^0 \\ \Delta x_2^2 \end{bmatrix} = -\begin{bmatrix} 6 \\ 0 \end{bmatrix}$$



$$\Delta x_1^0 = -1$$

$$\Delta x_2^0 = -1$$

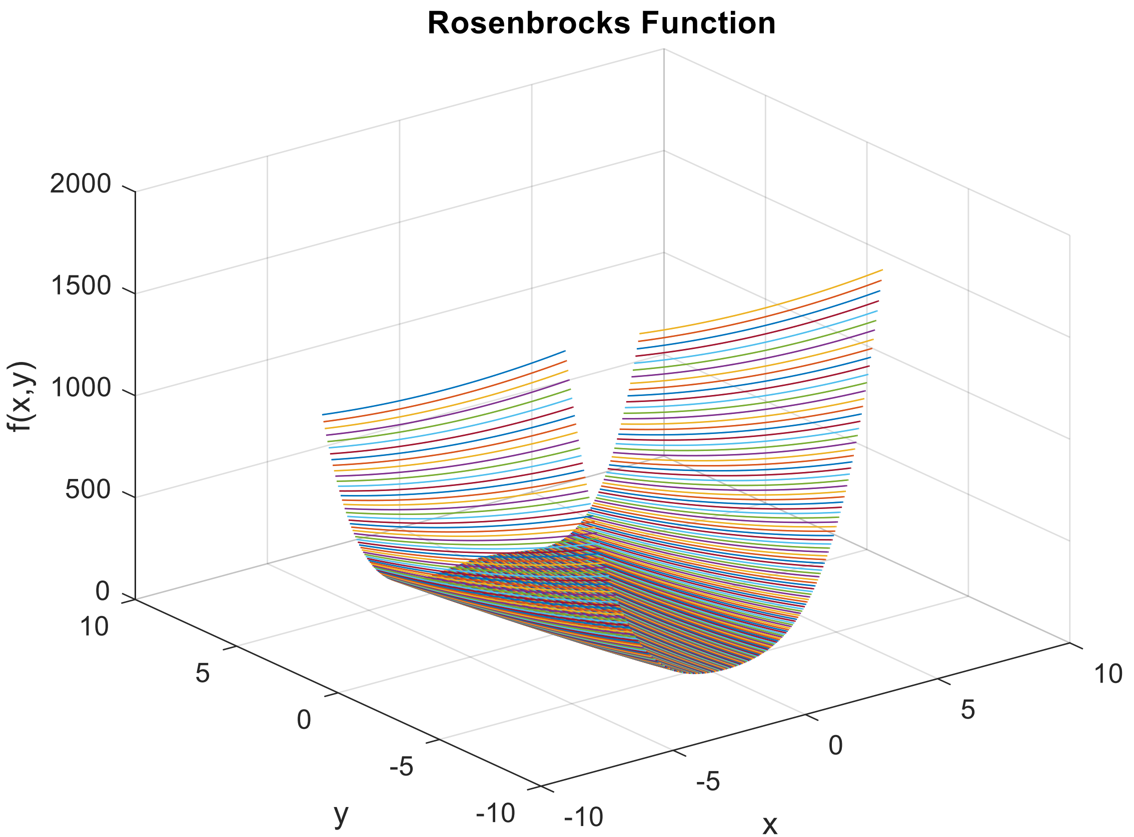
$$\mathbf{s}^0 = -\mathbf{H}^{-1}\,\nabla f(\mathbf{x}^0)$$

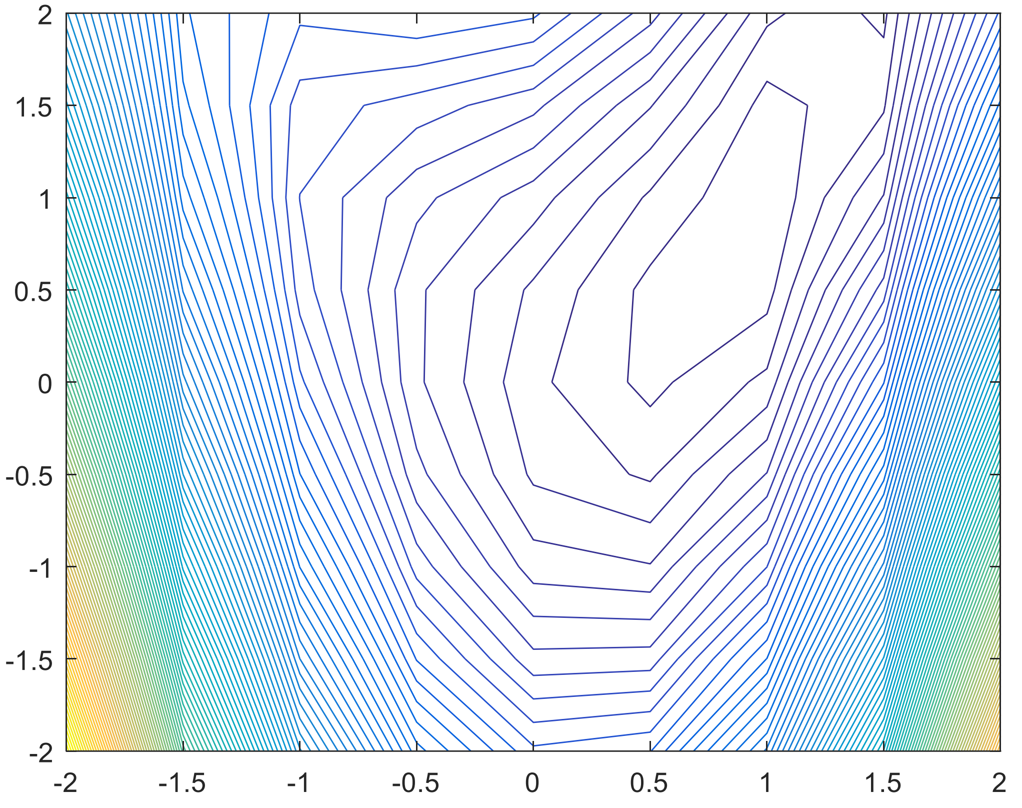
Edgar T.F., Himmelblau D.M.,. Lasdon L.S. Optimization of chemical processes New York: McGraw-Hill, 2001

```
f=100*(x(2)-x(1).^2).^2+(1-x(1)).^2;
```

Iteration	Number of function calls	$f(\mathbf{x})$	x_1	x_2	$\frac{\partial f(\mathbf{x})}{\partial x_1}$	$\frac{\partial f(\mathbf{x})}{\partial x_2}$
0	1	24.2	−1.2	1.0	−215.6	−88.00
1	4	4.377945	−1.050203	1.061141	−21.65	−8.357
5	14	3.165142	−0.777190	0.612232	−1.002	−1.6415
10	28	1.247687	−0.079213	−0.025322	−3.071	−5.761
15	41	0.556612	0.254058	0.063189	−1.354	−0.271
20	57	0.147607	0.647165	0.403619	3.230	−3.040
25	69	0.024667	0.843083	0.710119	−0.0881	−0.1339
30	80	0.0000628	0.995000	0.989410	0.2348	−0.1230
35	90	1.617×10^{-15}	1.000000	1.000000	-1.60×10^{-8}	-3.12×10^{-8}

v=100*(y-x.^2).^2+(1-x).^2;





Rosenbrocks Fonksiyonunun kontur grafiği

```
function f=fun208(x);
f=100*(x(2)-x(1).^2).^2+(1-x(1)).^2;

end
```

```
>> x0=[0  0];
>> x=fminunc('fun208',x0)


x =

    1.0000    1.0000
```