

Optimizasyon Teknikleri

Ders Notu – 3

TEK DEĞİŞKENLİ OPTİMİZASYON

Prof. Dr. Bilal ALATAŞ

İÇERİK

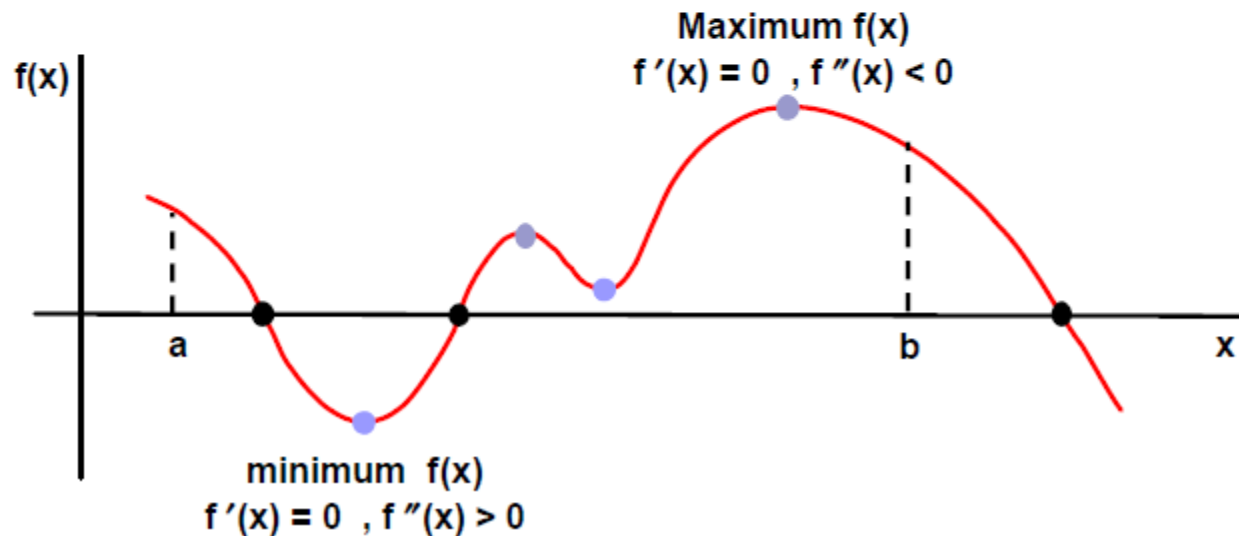
- **TEK DEĞİŞKENLİ OPTİMİZASYON METOTLARI**
 - GOLDEN SECTION METODU
 - BISECTION METODU
 - POLINOM METODU
 - NEWTON-RAPHSON METODU
 - SECANT METODU
- **METOTLAR HAKKINDA GENEL DEĞERLENDİRME**

TEK DEGISKENLI OPTIMIZASYON METOTLARI

Tek degiskenli optimizasyon ifadesi:

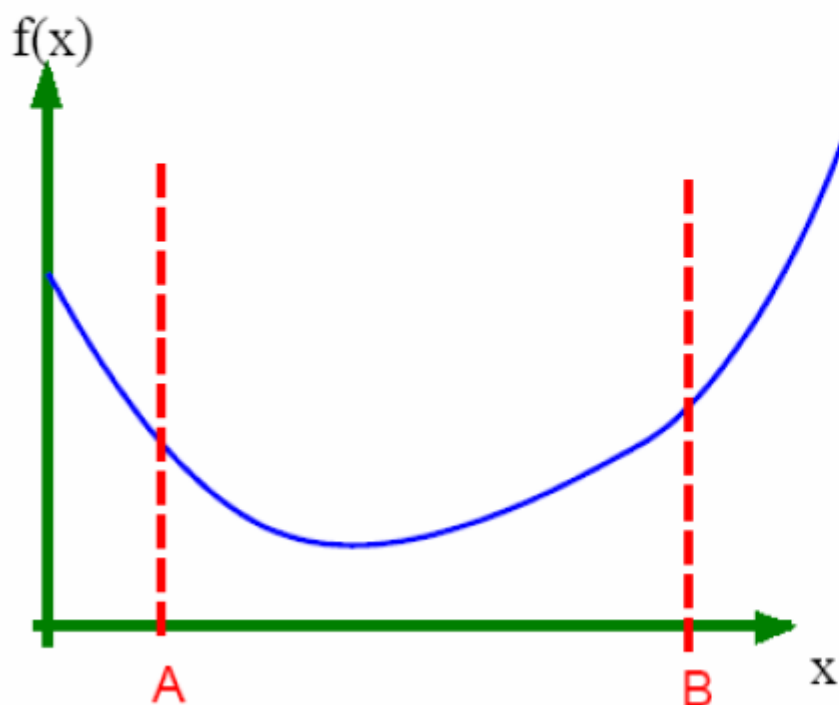
Minimize: $f(x)$

x : optimizasyon degiskeni



TEK DEGISKENLI OPTIMIZASYON METOTLARI

Optimizasyon probleminin çözümü:



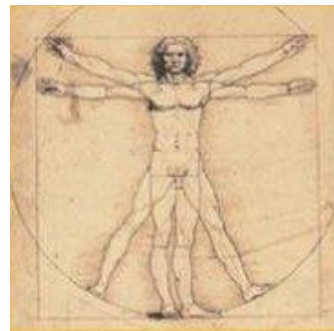
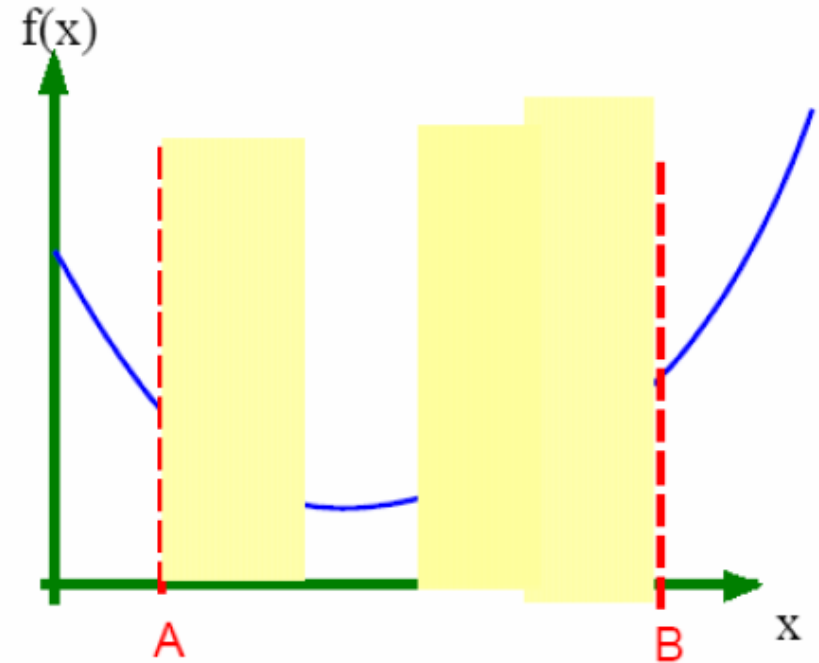
Verilen $f(x)$ fonksiyonunun $[A, B]$ aralığında minimumunu bulmak istiyoruz.

Minimumu en az hesapla ($f(x)$ degeri kullanarak) bulmak istiyoruz.

$f(x)$ fonksiyonunun açık halinin (analitik ifadesi) bilinip bilinmediği durumlarda minimum nasıl bulunur?

GOLDEN SECTION METODU

- Golden section (Altın Bölme) metodun temeli İtalyan matematikçi Fibonacci tarafından atılmıştır (1202).
- Metot, optimumun araştırılacağı alanı sürekli daraltmaya dayanır.
- Optimizasyon sonunda çok küçük alanın optimum çözümü barındırdığı düşünülür.



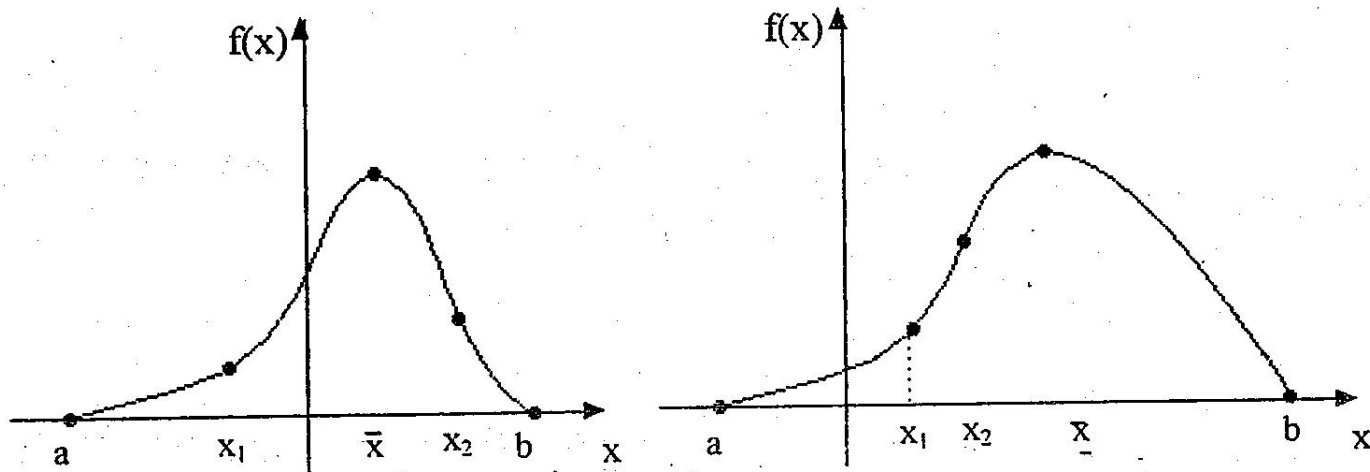
- $f'(x)$ bulunamadığında veya $f'(x) = 0$ eşitliğinin çözülemediği durumlarda tek değişkenli bir fonksiyonun daha önce anlattığımız yollarla çözülmesi zor olabilir.
- Bu durumda tek değişkenli $f(x)$ fonksiyonunun en iyi değerinin araştırılmasında tek değişkenli araştırma teknikleri kullanılabilir. Bu tekniklerin kullanılabilmesi için fonksiyonun tek modlu olması şarttır.
- $f(x)$ fonksiyonu $a \leq x \leq b$ koşulu altında maksimize edilmek istensin. Bu problem genel gösterimle aşağıdaki gibi ifade edilir:
 - $\max_{a \leq x \leq b} f(x)$

TANIM: $f(x)$, $[a, b]$ kapsamındaki bir nokta (\bar{x}) için $[a, \bar{x}]$ aralığında hızla artarken, $[\bar{x}, b]$ aralığında hızla azalıyorsa tek modlu bir fonksiyondur.

Eğer $f(x)$ tek modlu bir fonksiyon ise bahsedilen problemin en iyi çözümü $[a, b]$ aralığındadır. En iyi çözümün bulunduğu kesin olan bu aralığın küçültülmesi uygun olur. Bunun için $x_1 < x_2$ olmak üzere iki nokta seçilir ve fonksiyonun bu noktalardaki değeri hesaplanır. $f(x_1)$ ve $f(x_2)$ değerlerinin karşılaştırılması sonucunda, üç durumdan biriyle karşılaşılır:

Durum1: $f(x_1) < f(x_2)$

$F(x)$ tek modlu olduğundan $[x_1, x_2]$ alt aralığının en azından bir bölümünde artacaktır. Bu durumda problemin en iyi çözümü $[a, x_1]$ aralığında bulunamaz. En iyi çözüm $\bar{x} \in (x_1, b]$ olur.

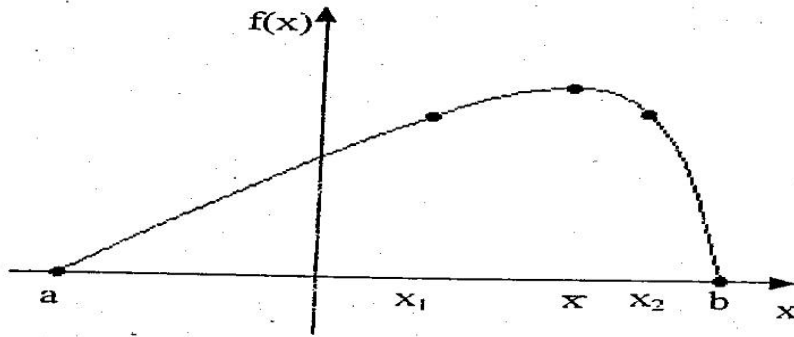


Şekil 6.21

$f(x_1) < f(x_2)$ ise, $\bar{x} \in (x_1, b]$

Durum2: $f(x_1)=f(x_2)$

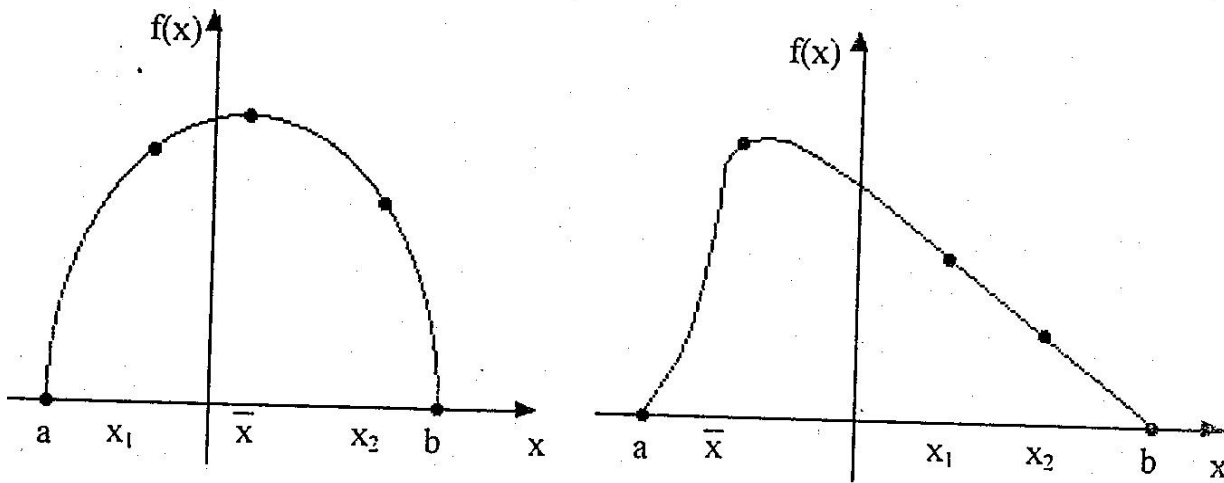
$f(x)$ tek modlu olduğundan, $[x_1, x_2]$ alt aralığının belirli bir bölümünde azalır. U durumda, problemimizin en iyi çözümü $\bar{x} \in [a, x_2)$ aralığında olur.



Şekil 6.22
 $f(x_1) = f(x_2)$ ise, $\bar{x} \in [a, x_2)$

Durum3: $f(x_1) > f(x_2)$

Bu durumda fonksiyonun eğrisi x_2 ye ulaşmadan önce azalmaya başlar. Buna göre $\bar{x} \in [a, x_2)$ olur.



Şekil 6.23

$f(x_1) > f(x_2)$ ise, $\bar{x} \in [a, x_2)$

Bu durumda $f(x)$ in en büyük noktası $[a, x_2)$ veya $(x_1, b]$ aralıklarından birinde bulunur. \bar{x} in bulunduğu aralığa belirsizlik aralığı denir.

Adım1: fonksiyonun tanımlandığı $[a, b]$ kapalı aralığı x 'in belirsizlik aralığı olarak seçilir. $[a, b]$ araştırmanın başlatıldığı ilk aralık olduğundan $[a, b]$ ye ilk veya başlangıç belirsizliği denir. Bu aralıktan uygun olarak x_1 ve x_2 gibi herhangi iki nokta seçilir ve $f(x)$ in bu noktalardaki değerleri hesaplanır.

Adım2: $f(x_1)$ ve $f(x_2)$ değerleri karşılaştırılarak yukarıda açıklanan üç durumdan hangisinin ortaya çıktığı belirlenir. Saptanan durum doğrultusunda başlangıç belirsizlik aralığı daraltılabilir. Bu adımda belirlenen daha dar aralık üçüncü adımın belirsizlik aralığı olur.

Adım3: ikinci adımda belirlenen belirsizlik aralığından aynı yaklaşımla iki yeni nokta seçilir ve fonksiyonun bu noktalardaki değeri hesaplanır. Hesaplanan değerler karşılaştırılarak 2. Adımda belirlenenden daha kısa olan bir aralık saptanır. Belirlenen aralığın uzunluğu yeterince kısa değilse ikinci adıma dönülür. Bu adımlar yeterince kısa adımlar elde edinceye kadar tekrar edilir.

Bu adımların kullanıldığı Golden kesit araştırmasında noktaların seçilmesi için golden oran adı verilen ve mimarlıkta sıklıkla kullanılan bir oran kullanılır.

$r^2 + r = 1$ kareli eşitliğinin tek pozitif kökü r olsun. Bu durumda golden oran (r);

$$r = \frac{\frac{1}{52} - 1}{2} = 0.618$$

olarak belirlenir.

Golden kesit araştırmasının ilk adımında $f(x)$ in $x_1 = b - r(b - a)$ ve $x_2 = a + r(b - a)$ olarak belirlenen noktalardaki değerleri hesaplanır.

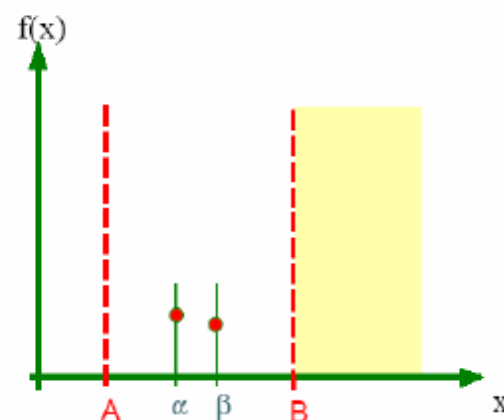
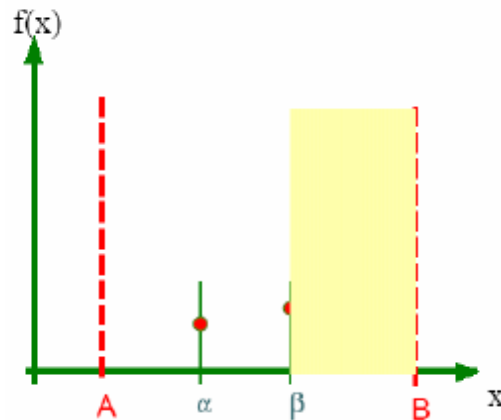
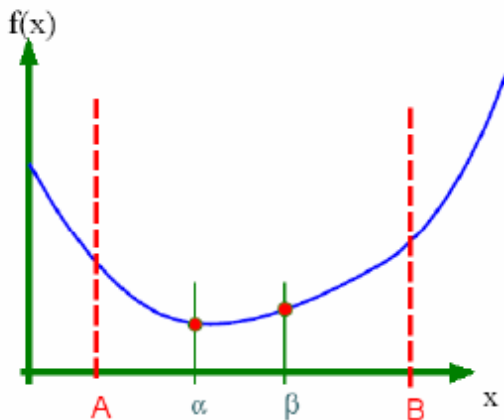
Yeni sol taraf noktası: yürürlükteki belirsizlik aralığının sağ bitim noktasından r kesri ile belirlenen noktaya hareket edilir.

Yeni sağ taraf noktası: yürürlükteki belirsizlik aralığının sol bitim noktasından r kesri ile belirlenen noktaya hareket edilir.

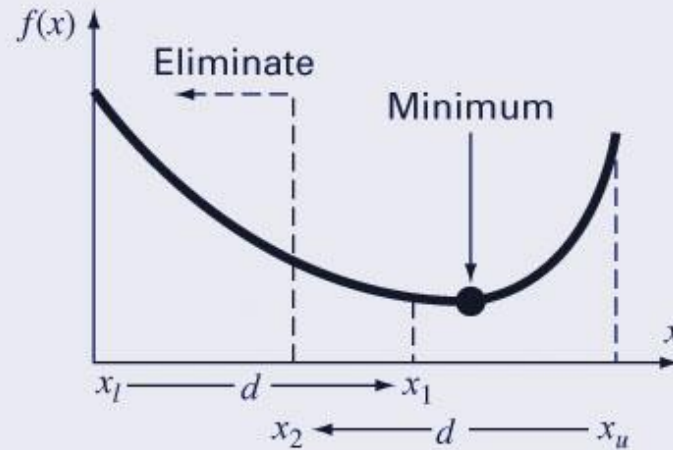
GOLDEN SECTION METODU

Golden Section metodu çözüm algoritması:

- Her iterasyonda, bir önceki alanının bir kısmi atılarak alan daraltılır.
- $f(x)$ 'in türevi (gradyenti) kullanılmadığından, alanın hangi bölgesinin atılacağına $f(x)$ fonksiyonunun 2 noktadaki değeri kullanılarak karar verilir.
- $f(a) < f(\beta)$ ise, $[\beta, B]$ aralığı atılır ve bir sonraki iterasyonda $B = \beta$ olur
- $f(a) > f(\beta)$ ise, $[A, a]$ aralığı atılır ve bir sonraki iterasyonda $A = a$ olur
- Noktaların (a, β) yerlerinin seçiminde altın bölme (Golden Section) kuralı uygulanır. Bu işlem yenilenmiş $[A, B]$ aralığı ile çözüm yakınsayana kadar tekrar edilir.

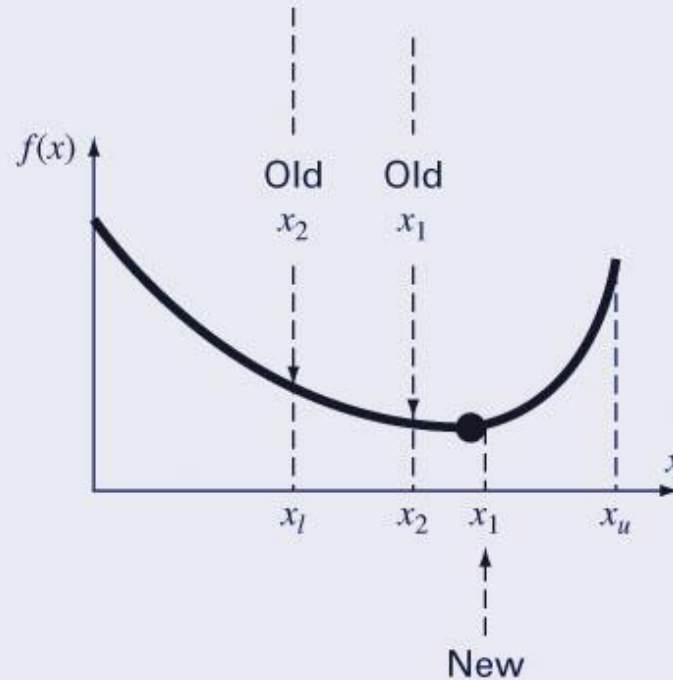


- $f(x_1) < f(x_2) \rightarrow x_2$
yeni alt limittir; x_1
de yeni x_2 dir.



(a)

- $f(x_2) < f(x_1) \rightarrow x_1$
yeni üst limittir; x_2
de yeni x_1 dir.



(b)

Initialize:

$x_1 = a + (b-a) \cdot 0.382$

$x_2 = a + (b-a) \cdot 0.618$

$f_1 = f(x_1)$

$f_2 = f(x_2)$

Loop:

if $f_1 > f_2$ then

$a = x_1; x_1 = x_2; f_1 = f_2$

$x_2 = a + (b-a) \cdot 0.618$

$f_2 = f(x_2)$

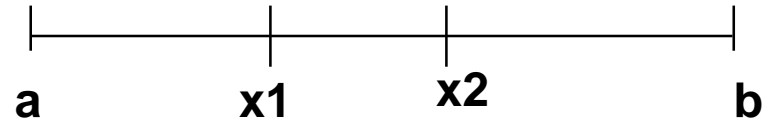
else

$b = x_2; x_2 = x_1; f_2 = f_1$

$x_1 = a + (b-a) \cdot 0.382$

$f_1 = f(x_1)$

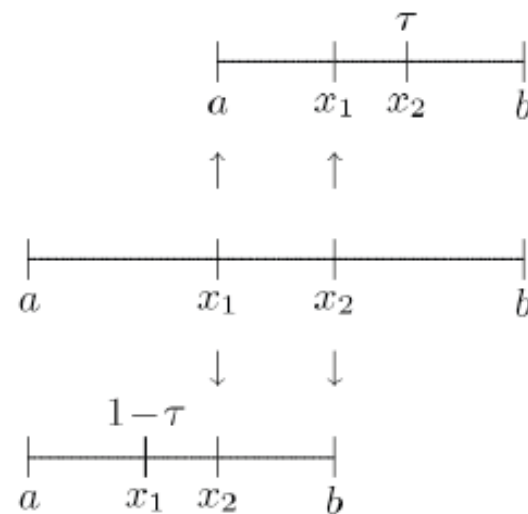
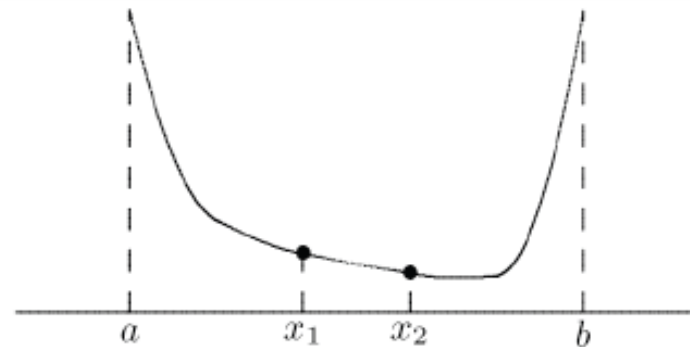
endif



```

 $\tau = (\sqrt{5} - 1)/2$ 
 $x_1 = a + (1 - \tau)(b - a); f_1 = f(x_1)$ 
 $x_2 = a + \tau(b - a); f_2 = f(x_2)$ 
while  $((b - a) > tol)$  do
  if  $(f_1 > f_2)$  then
     $a = x_1$ 
     $x_1 = x_2$ 
     $f_1 = f_2$ 
     $x_2 = a + \tau(b - a)$ 
     $f_2 = f(x_2)$ 
  else
     $b = x_2$ 
     $x_2 = x_1$ 
     $f_2 = f_1$ 
     $x_1 = a + (1 - \tau)(b - a)$ 
     $f_1 = f(x_1)$ 
  end
end

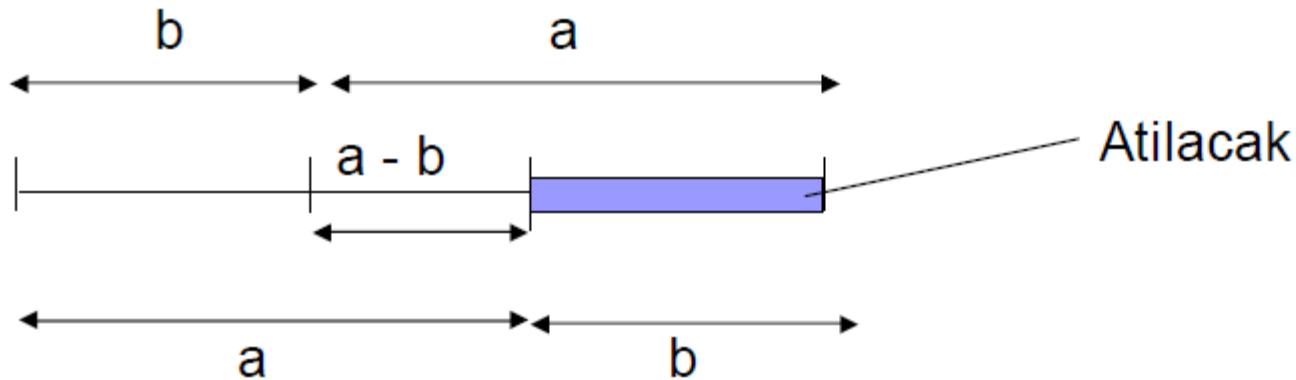
```



GOLDEN SECTION METODU

Altın Bölme (Golden Section) kuralı:

✓ Hesap maliyetini azaltmak için Golden Section metodunda aralık sürekli belirli oranda (% 38 oranında) daraltılır.



$$b/(a-b) = a/b \gg b*b = a*a - ab$$

Çözüm:

$$a = (b \pm b * \text{sqrt}(5)) / 2 \gg a/b = -0.618 \text{ or } 1.618 \text{ (Golden Section oranı)}$$

Not: $1/1.618 = 0.618$

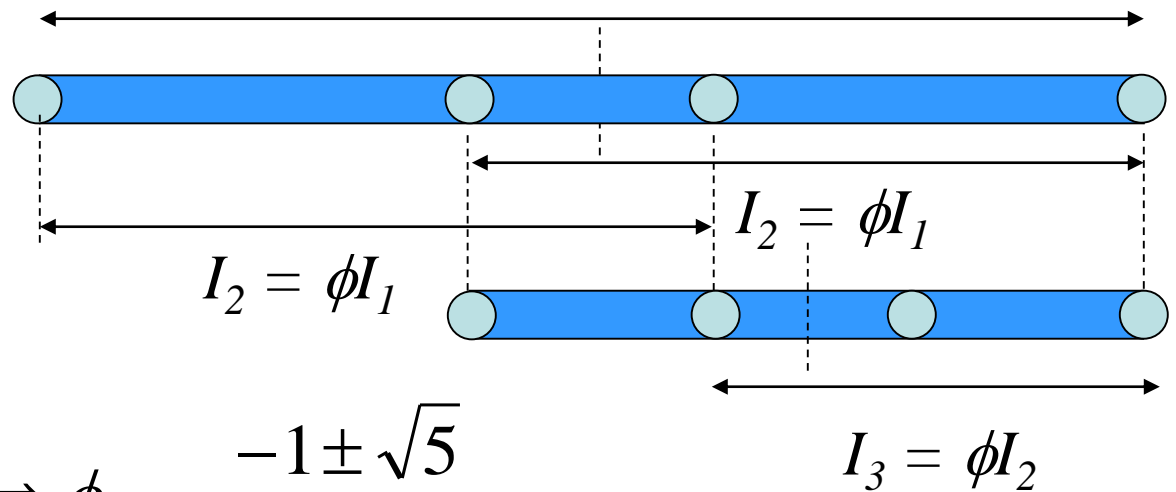
- GOLDEN SECTION'un temeli: I_1

$$I_1 = I_2 + I_3$$

$$\Rightarrow I_1 = \phi I_1 + \phi^2 I_1$$

$$\phi^2 + \phi - 1 = 0 \Rightarrow \phi_{1,2} = \frac{-1 \pm \sqrt{5}}{2}$$

$$\phi = \frac{\sqrt{5} - 1}{2} \approx 0.618034$$



- Son Aralık: $I_N = \phi^N I_1$

- Example

- Find maximum of $f(x) = \sin x - x^2$

- Start with $x_l=0$ and $x_u=1$

$$x_1 = 0 + \frac{\sqrt{5}-1}{2}(1-0) = 0.618 \quad \Rightarrow f(x_1) = 0.197$$

$$x_2 = 1 - \frac{\sqrt{5}-1}{2}(1-0) = 0.382 \quad \Rightarrow f(x_2) = 0.226$$

$$f(x_2) > f(x_1)$$

$$x_{u,new} = x_1 = 0.618$$

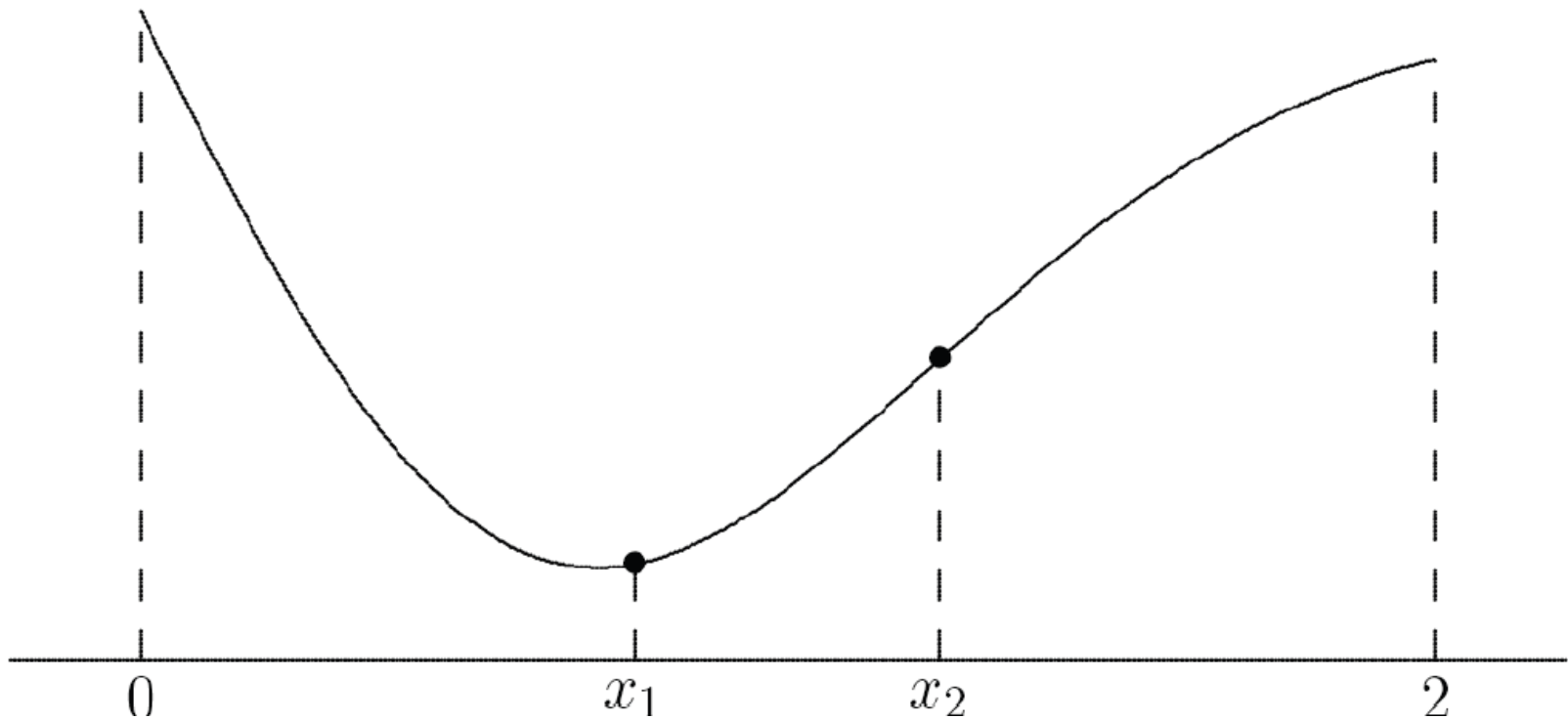
$$x_{l,new} = x_2 = 0.382$$

$$\begin{aligned} x_{2,new} &= x_{u,new} - \frac{\sqrt{5}-1}{2}(x_{u,new} - x_l) \\ &= 0.618 - 0.618(0.618 - 0) \\ &= 0.236 \end{aligned}$$

i	x_l	x_2	x_1	x_u	$f(x_2)$	$f(x_1)$
0	0.0000	0.3820	0.6180	1.0000	0.2268	0.1975
1	0.0000	0.2361	0.3820	0.6180	0.1782	0.2268
2	0.2361	0.3820	0.4721	0.6180	0.2268	0.2319
3	0.3820	0.4721	0.5279	0.6180	0.2319	0.2250
4	0.3820	0.4377	0.4721	0.5279	0.2323	0.2319
5	0.3820	0.4164	0.4377	0.4721	0.2311	0.2323

Örnek-2

Minimize $f(x) = 0.5 - x \exp(-x^2)$



x_1	f_1	x_2	f_2
0.764	0.074	1.236	0.232
0.472	0.122	0.764	0.074
0.764	0.074	0.944	0.113
0.652	0.074	0.764	0.074
0.584	0.085	0.652	0.074
0.652	0.074	0.695	0.071
0.695	0.071	0.721	0.071
0.679	0.072	0.695	0.071
0.695	0.071	0.705	0.071
0.705	0.071	0.711	0.071

GOLDEN SECTION METODU

Genel Değerlendirme:

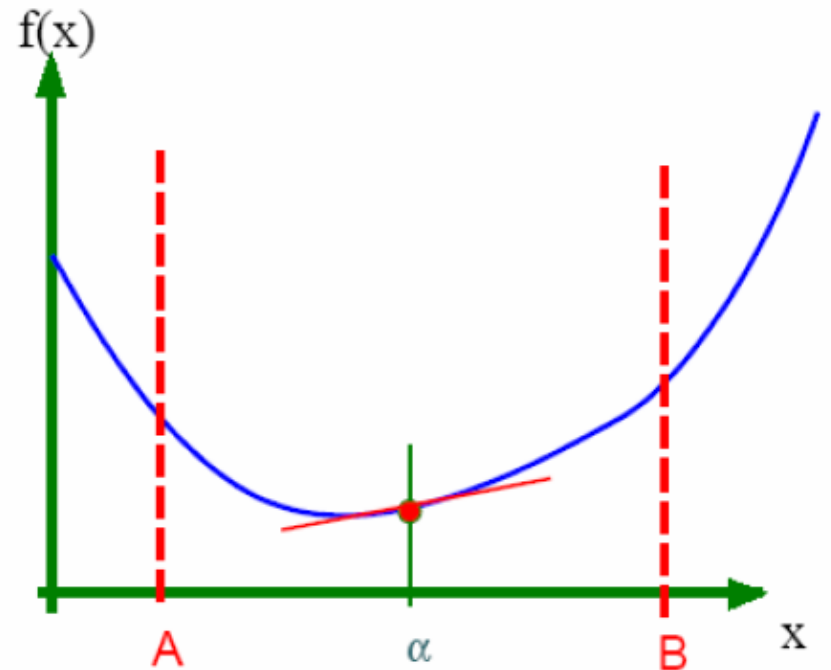
- Her iterasyon için çözüm aralığının $(1 - 0.618) = 38\%$ 'i elimine edilir.
- n iterasyon sonunda çözüm aralığı 0.618^n kat küçülmüş olacaktır.
- $n = 10$ olduğu düşünülürse bulunan çözüm aralığı orjinalinin % 1 'inden küçük olacaktır.
- Verilen çözüm aralığı, içinde tek bir minimum noktası içeren fonksiyonlarda kullanılabilir.

Java Animasyonu

- <http://www.cs.illinois.edu/~heath/iem/optimization/GoldenSection/>

BISECTION METODU

- Aralık daraltma metotlarında, elde türev bilgisi olursa çözüm hızlandırılabilir.
- Bu gibi durumda aralıkları ikiye bölerek (bisection) (daha hızlı) daraltmak mümkün olur.



BISECTION (İkiye Bölme) Metodu

Optimizasyonla kök bulma birbirine benzer. Her ikisi de bir fonksiyonun veya türevinin sıfır olduğu yeri bulmaya çalışır.

- Kök bulma: $f(x)=0$,
- Optimizasyon: $f'(x) = 0$

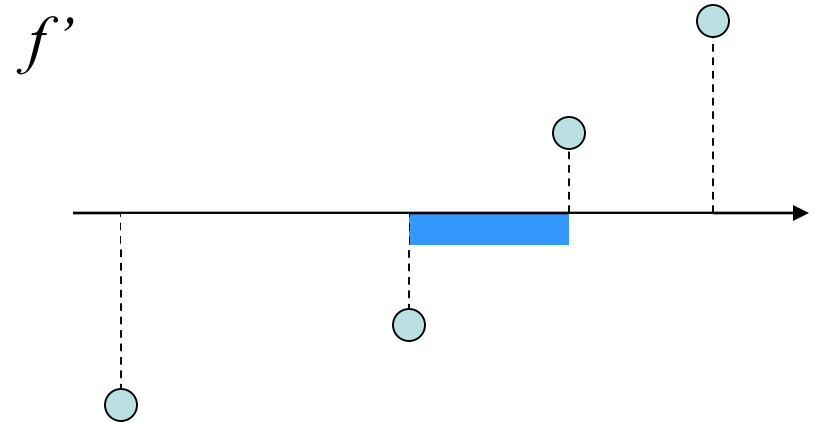
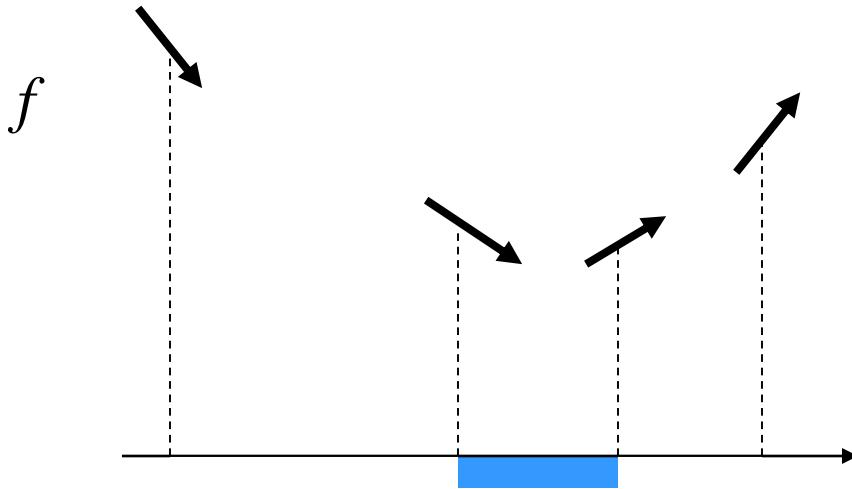
O halde optimum çözüm, $f'(x) = 0$ probleminin kök bulma yöntemleri ile çözümünden elde edilebilir (eğer fonksiyon türetilebilir ise).

f' köklerinin bulunmasına eşdeğerdir...

Minimumu sınırlayan iki nokta (x_a , $x_{\bar{u}}$) bulunur.

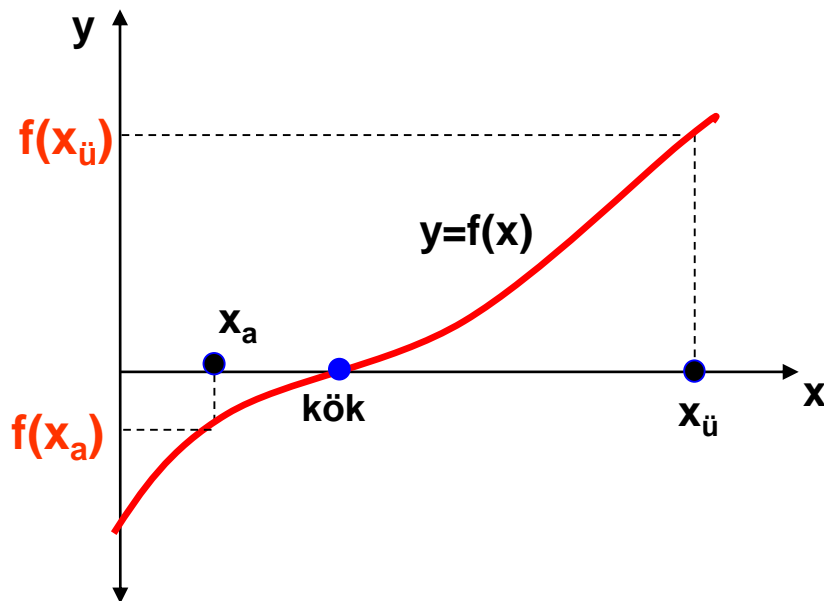
$$f'(x_a) < 0 \text{ ve } f'(x_{\bar{u}}) > 0$$

- Golden sectiona benzer ancak türev kullanır



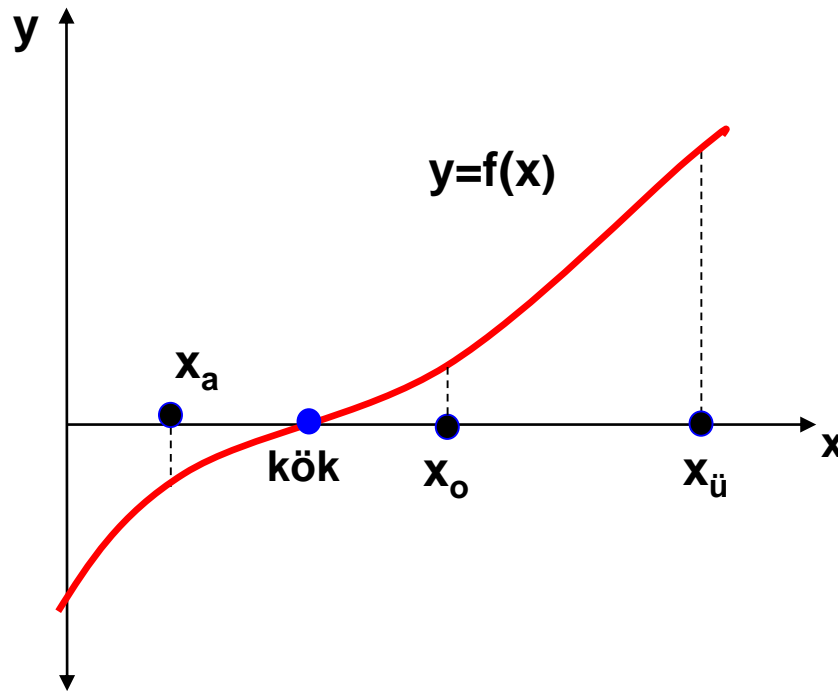
HATIRLAMA : İkiye Bölme Yöntemi

Genel olarak x_a ve $x_{\bar{u}}$ aralığında fks **sürekli** ve $f(x_a)$ ile $f(x_{\bar{u}})$ 'nün işaretleri ters ise yani **$f(x_a) \cdot f(x_{\bar{u}}) < 0$** ise bu aralıkta bir **kök** vardır.



İkiye bölme yönteminde **kökün bulunduğu aralık** adım adım **daraltılarak** gerçek köke ulaşılmaya çalışılır.

HATIRLAMA : İkiye Bölme Yöntemi



$f(x) = 0$ 'ı sağlayan kökün içinde bulunduğu aralığın **alt** ve **üst değeri** biliniyorsa bu iki değer **orta noktası** için **değeri** bulunabilir.

$$x_o = \frac{x_a + x_{\ddot{u}}}{2}$$

HATIRLAMA : İkiye Bölme Yöntemi

İşlem adımları

- 1) Kökün bulunduğu aralık için x_a ve $x_{\bar{u}}$ değerleri tahmin edilir ve $f(x_a).f(x_{\bar{u}}) < 0$ şartı aranır.
- 2) Üst ve alt değerlerle **orta değer** (x_o) hesaplanır.

$$x_o = \frac{x_a + x_{\bar{u}}}{2}$$

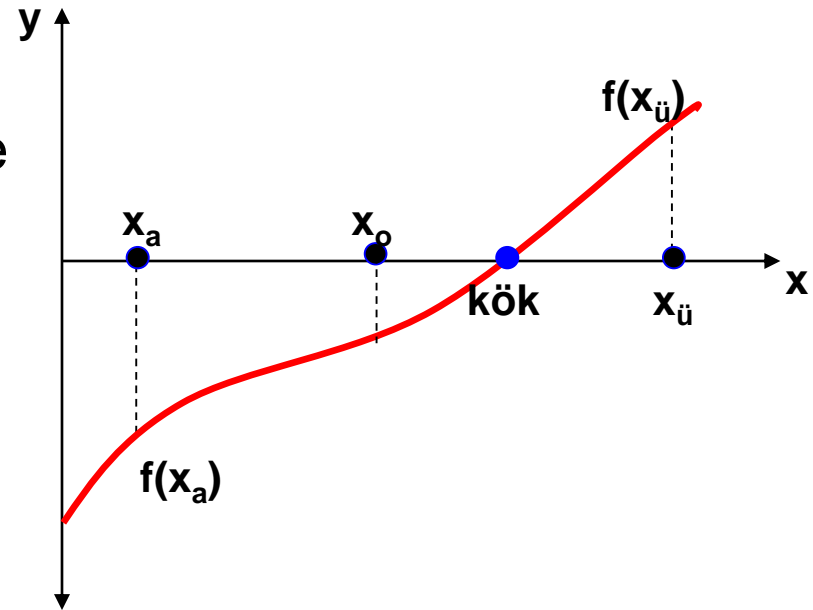
- 3) $f(x_o)$ değeri hesaplanır
Eğer $f(x_o) = 0$ ise **kök** x_o 'dır.
Eğer $f(x_o) \neq 0$ ise **işleme devam** edilir
- 4) $f(x_a)$ hesaplanır

HATIRLAMA : İkiye Bölme Yöntemi

5. a)

$f(x_a) \cdot f(x_o) > 0$ ise

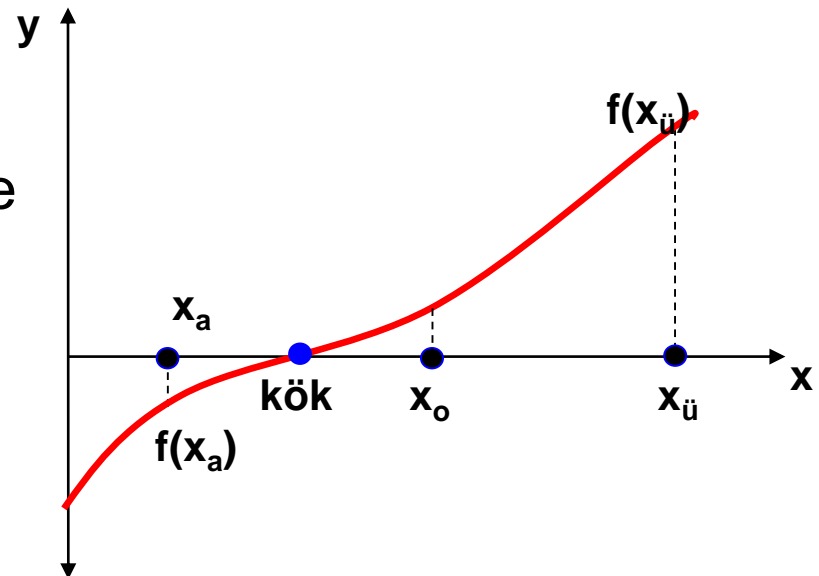
x_a yerine x_o yazılarak işleme devam edilir



b)

$f(x_a) \cdot f(x_o) < 0$ ise

$x_{\bar{u}}$ yerine x_o yazılarak işleme devam edilir.



HATIRLAMA : İkiye Bölme Yöntemi

İşleme son verme

- 1) $f(x_o)=0$ olunca işleme son verilir
Kök x_o 'dır.
- 2) $|\epsilon_t| < \epsilon_k$ ise işleme son verilir.

$$\epsilon_t = \frac{\text{son deger} - \text{bir önceki deger}}{\text{son deger}} = \frac{x_{o,k+1} - x_{o,k}}{x_{o,k+1}}$$

HATIRLAMA : İkiye Bölme Yöntemi

İkiye Bölme Yöntemi Özet Adımlar

Adım 1: $f(x_a) f(x_{\bar{u}}) < 0$ şartını sağlayan x_a ve $x_{\bar{u}}$ değerlerini tahmin et ve yaklaşım kriteri için uygun bir % sapma değeri (ε_k) kabul et.

Adım 2: aşağıdaki eşitliği kullanarak kök için tahmini bir değer hesapla

$$x_o = (x_a + x_{\bar{u}}) / 2$$

Adım 3: Eğer $f(x_a) f(x_{\bar{u}}) < 0 \Rightarrow x_{\bar{u}} = x_o$ yazarak Adım 2' ye dön

Eğer $f(x_a) f(x_{\bar{u}}) > 0 \Rightarrow x_a = x_o$ yazarak Adım 2' ye dön

Adım 4:

$$\varepsilon_t = \left| \frac{x_o^{\text{yeni}} - x_o^{\text{eski}}}{x_o^{\text{yeni}}} \right| \leq \varepsilon_k$$

oluncaya kadar bu işlemlere devam et.

Buradaki mantıkla f' köklerinin bulunması işi yapılır ...

Minimumu sınırlayan iki nokta $(x_a, x_{\bar{u}})$ bulunur.

$$f'(x_a) < 0 \text{ ve } f'(x_{\bar{u}}) > 0$$

ve işlemler f' için açıklanan şekilde uygulanır...

Türevsiz Algoritma

algorithm *MinBisect* ($f, a, b, tol, maxits$)

for $k := 1$ **to** $maxits$ **do**

$m := a + (b - a)/2$

if $f(m - \epsilon) < f(m + \epsilon)$ **then**

$b := m + \epsilon$

else

$a := m - \epsilon$

endif

if $|a - b| \leq tol$ **then return** m

endfor

endalg {MinBisect}

Örnek

$$\text{Max } f(x) = 12x - 3x^4 - 2x^6$$

	$df(x)/dx$	\underline{x}	\bar{x}	New x'	$f(x')$
0					
1					
2					
3	4.09	0.75	1	0.875	7.8439
4	-2.19	0.75	0.875	0.8125	7.8672
5	1.31	0.8125	0.875	0.84375	7.8829
6	-0.34	0.8125	0.84375	0.828125	7.8815
7	0.51	0.828125	0.84375	0.8359375	7.8839

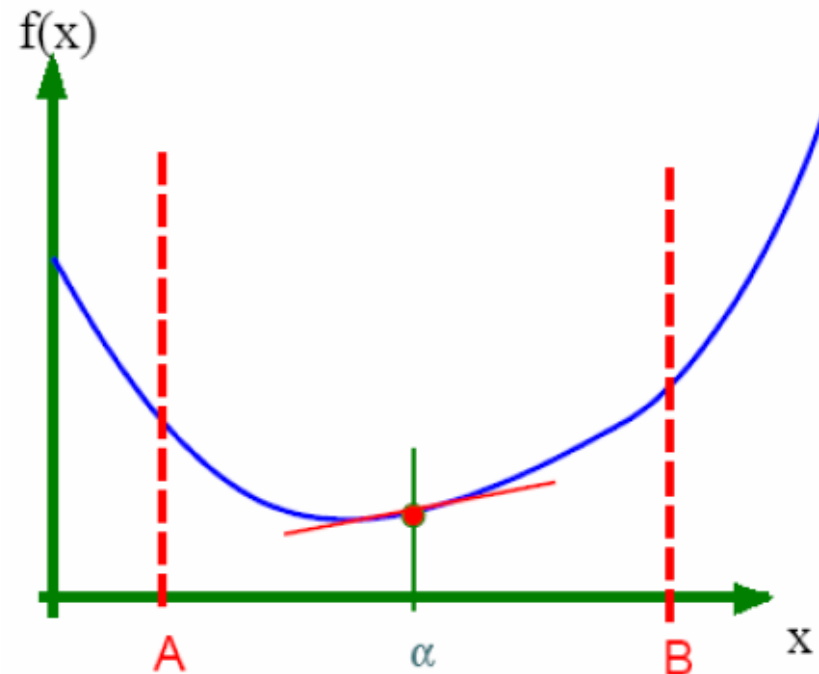
BISECTION METODU

Avantaj:

- Daha hızlı yakinsama. Örneğin, 10 iterasyondan sonra alan %99 oranında daraltılmış olur.

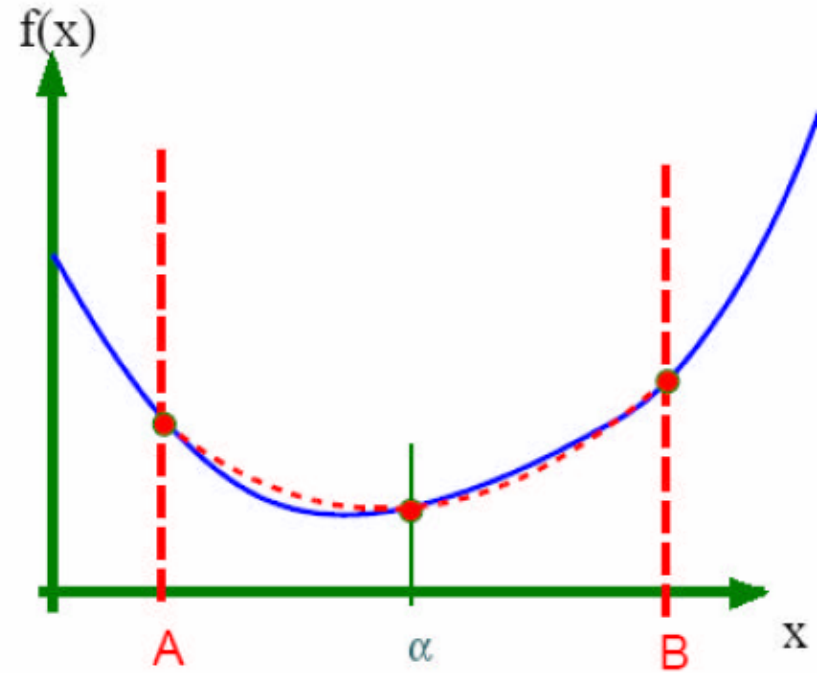
Dezavantaj:

- Türev bilgisi gerekli. Türev hesabı genellikle biraz fazla hesap gerektirir.
- Türev bilgisi kullanıldığı zaman fonkiyonun sürekli (türetilir) olduğu kabul edilmiş olur. Golden section metodunda bu sınırlama yoktur.
- Bisection metodu, golden section metodundan daha az garantilidir.



POLINOM METODU

- Aralığın uç ve orta noktalarında fonksiyon değerleri hesaplanır.
- Hesaplanan 3 değerden bir parabol geçirilir.
- Parabolün minimum noktası (yeni nokta) ve karşı gelen fonksiyon değeri analitik olarak hesaplanır.
- Yeni nokta, eski noktalardan en kötü olanın yerini alarak iterasyon yakinsayana kadar devam edilir.



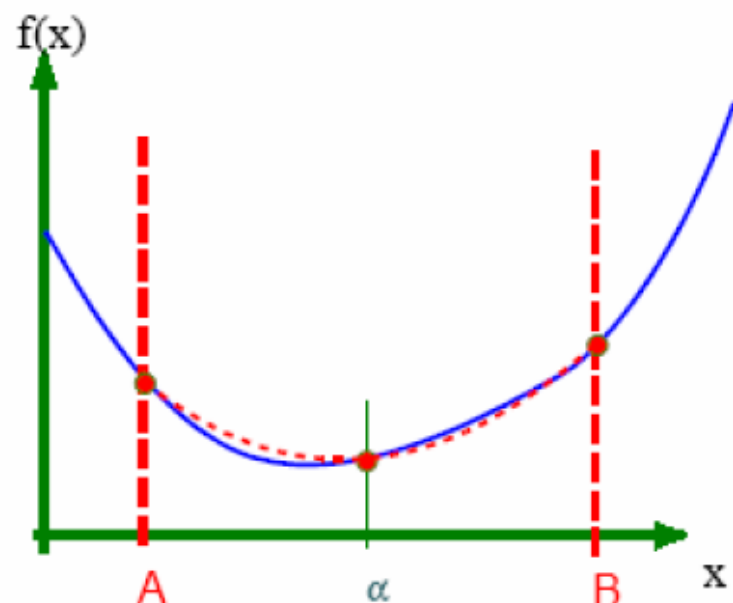
POLINOM METODU

Avantaj:

- Fonksiyon düzgünse yakinsama çok hızlıdır.
- Türev bilgisi gerekmez. Her iterasyonda fazladan bir nokta için fonksiyon değeri hesaplanır.

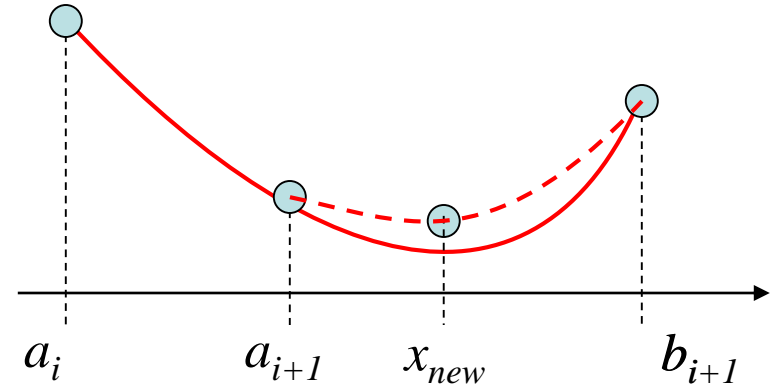
Dezavantaj:

- Algoritma non-konveks fonksiyonlara karşı çok hassastır.
- Optimizasyon için fonksiyonun 2. mertebeden türetilir olması lazımdır.



Not: Genel uygulamalar için tavsiye edilmemesine rağmen, özel uygulamalarda faydalı olabilir.

$$\tilde{f}(x) = ax^2 + bx + c$$



- Yeni nokta parabolun minimumunda değerlendirilir:

$$\tilde{f}' = 2ax + b = 0 \Rightarrow x_{new} = \frac{-b}{2a}$$

- Minimum için: $a > 0$!
- Mevcut noktaya çok yakınsa x_{new} kaydır

Üç nokta quadratik yaklaşım

- Üç noktadaki fonksiyonun değerine ihtiyaç gösterir (x_1, f_1) , (x_2, f_2) , (x_3, f_3)
- $q(x) = a_0 + a_1(x - x_1) + a_2(x - x_1)(x - x_2)$
- $x = x_1 \rightarrow q = f_1$ $a_0 = f_1$
- $x = x_2 \rightarrow q = f_2$ $f_2 = f_1 + a_1(x_2 - x_1) \rightarrow a_1 = (f_2 - f_1)/(x_2 - x_1)$
- $x = x_3 \rightarrow q = f_3$ $f_3 = f_1 + [(f_2 - f_1)/(x_2 - x_1)](x_3 - x_1)(x_3 - x_2) \rightarrow a_2 = [(f_3 - f_1)/(x_3 - x_1) - (f_2 - f_1)/(x_2 - x_1)]/(x_3 - x_2)$
- $dq/dx = 0 \rightarrow x^*$ (opt. noktayı buluruz)

Örnek

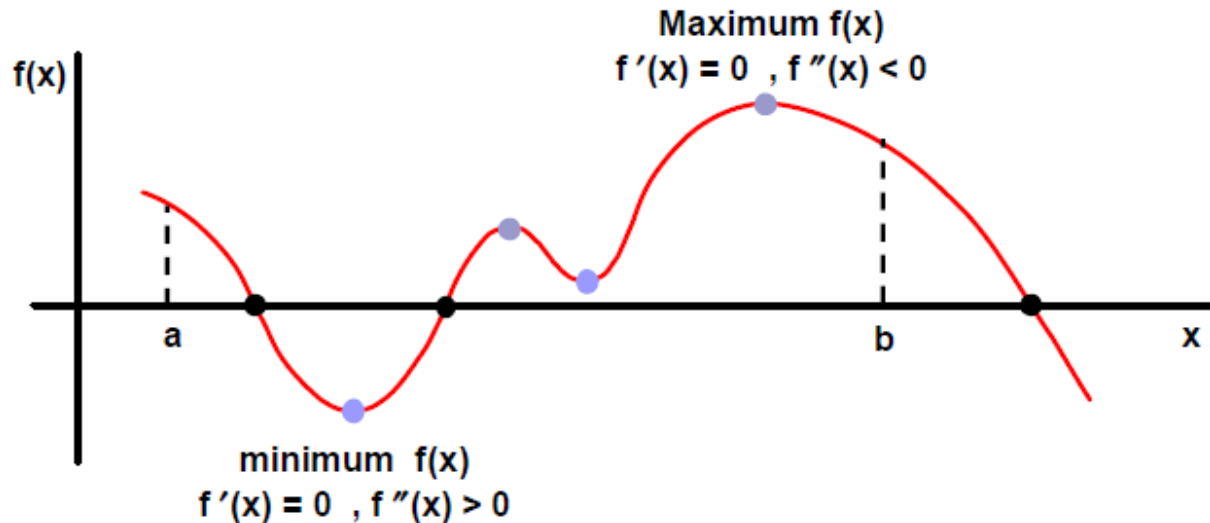
- $f(x)=(2x^3+16)/x \quad 1 \leq x \leq 5$
 - $x_1=1 \rightarrow f_1=18$
 - $x_2=3 \rightarrow f_2=23.33$
 - $x_3=5 \rightarrow f_3=53.2$
- Formüllerle
- $a_0=18, a_1=2.67, a_2=3.07$
- $q(x)=18+2.67(x-1)+3.07(x-1)(x-3)$
 - $dq/dx=0 \rightarrow x^*=[(x_2+x_1)/2](-a_1/2a_2)=1.565$
 - Gerçek $x^*=1.5874$

NEWTON-RAPHSON METODU

■ Optimizasyonla kök bulma birbirine benzer. Her ikisi de bir fonksiyonun veya türevinin sıfır olduğu yeri bulmaya çalışır.

✓ Kök bulma: $f(x)=0$,

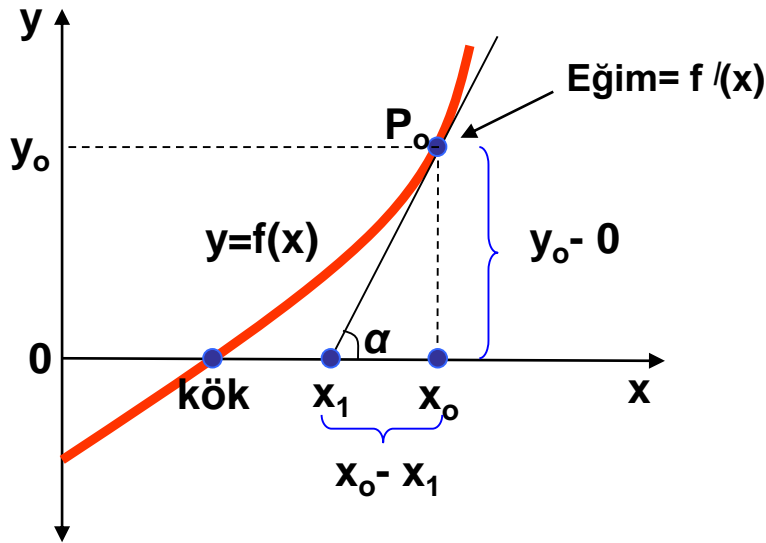
✓ Optimizasyon: $f'(x) = 0$



✓ O halde optimum çözüm, $f'(x) = 0$ probleminin kök bulma yöntemleri ile çözümünden elde edilebilir (eger fonksiyon türetilebilir ise).

Hatırlama (Kök Bulma)

Eğer kökün ilk tahmini x_0 ise, $[x_0, f(x_0)]$ noktasındaki teğet uzatılabilir. Uzatılan teğetin x eksenini kestiği nokta genellikle kökün daha iyi bir tahminini verir.

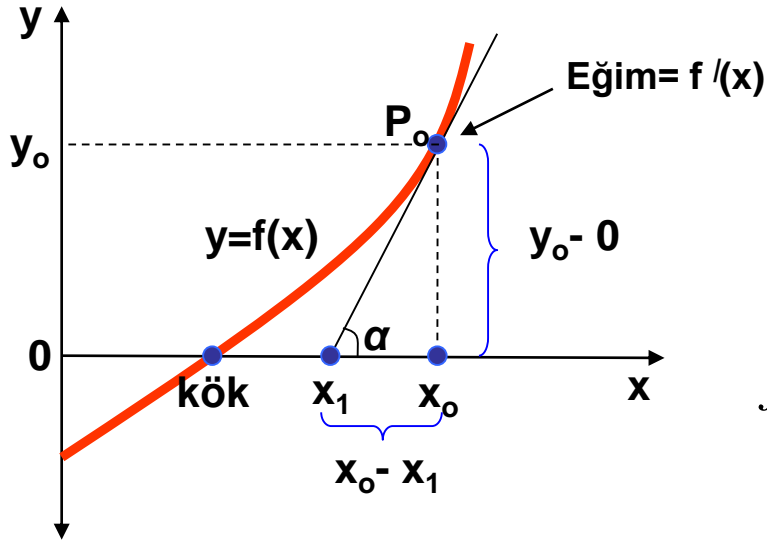


$y = f(x)$ fks.nun x_0 değeri $y_0 = f(x)$ 'dir.
 P_0 noktasından çizilen teğetin eğimi $\tan(\alpha)$ 'dır.

$$\tan \alpha = \frac{y_0 - 0}{x_0 - x_1} = \frac{f(x_0) - 0}{x_0 - x_1}$$

Newton-Raphson yönteminin grafik gösterimi. x_{i+1} kökünü tahmin edebilmek için x_i 'de fonksiyonun teğeti [yani $f'(x)$] x eksenini kesecek şekilde uzatılmaktadır. Teğetin x eksenini kestiği nokta kök için yeni değerdir

Hatırlama (Kök Bulma)



Aynı zamanda x_0 noktasındaki eğim bu noktadaki fonksiyonun türevine eşit olacağından :

$$f'(x_0) = \frac{f(x_0)}{x_0 - x_1} \longrightarrow x_0 - x_1 = \frac{f(x_0)}{f'(x_0)}$$

Buradan x_1 değeri ; $x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$

Bir sonraki adımdaki değer: $x_2 = x_1 - \frac{f(x_1)}{f'(x_1)}$

En genel şekilde: $x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$

Hatırlama (Kök Bulma)

- Newton-Raphson yönteminde iterasyona iki şekilde son verilir.
 1. Bulunan x değeri için $f(x)$ fks.nun değerinin 0 'a yaklaşımına bakarak,
 2. x değerinin bir önceki hesaplanan değerine ϵk kadar yaklaşmasına bakarak;
- İterasyona son verilir.

NEWTON-RAPHSON METODU

Taylor seri açılımı:

$$f(x) = f(x_k) + f'(x_k)(x - x_k) + \frac{1}{2} f''(x_k)(x - x_k)^2 + \dots$$

$f(x)$ 'in x 'e göre türevi alınırsa:

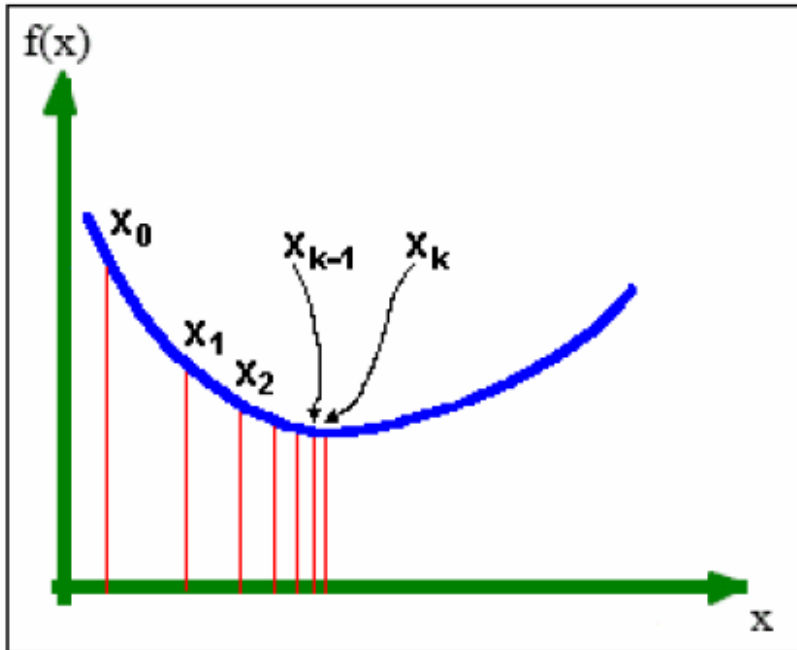
$$f'(x) = 0 + f'(x_k) + f''(x_k)(x - x_k)$$

$f'(x)=0$ noktası, Newton-Raphson metodunda optimum noktayı verir:

$$x_{k+1} = x_k - \frac{f'(x_k)}{f''(x_k)}$$

NEWTON-RAPHSON METODU

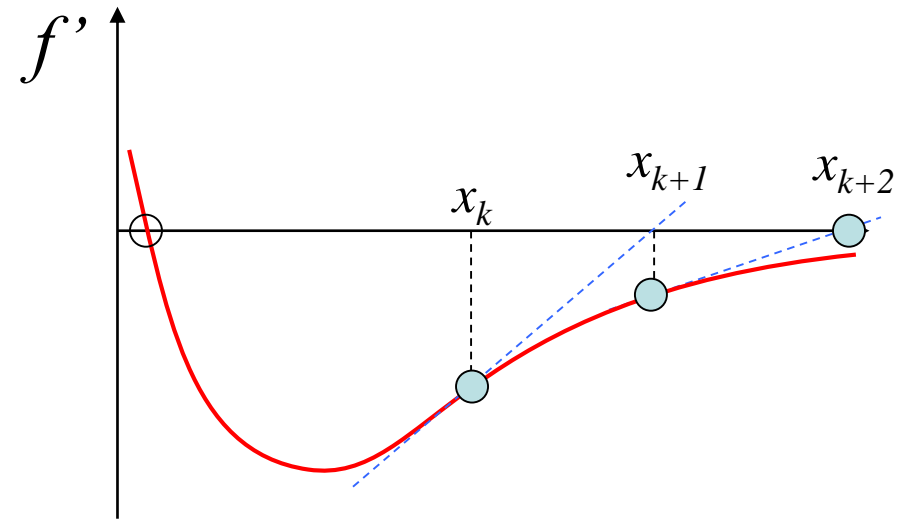
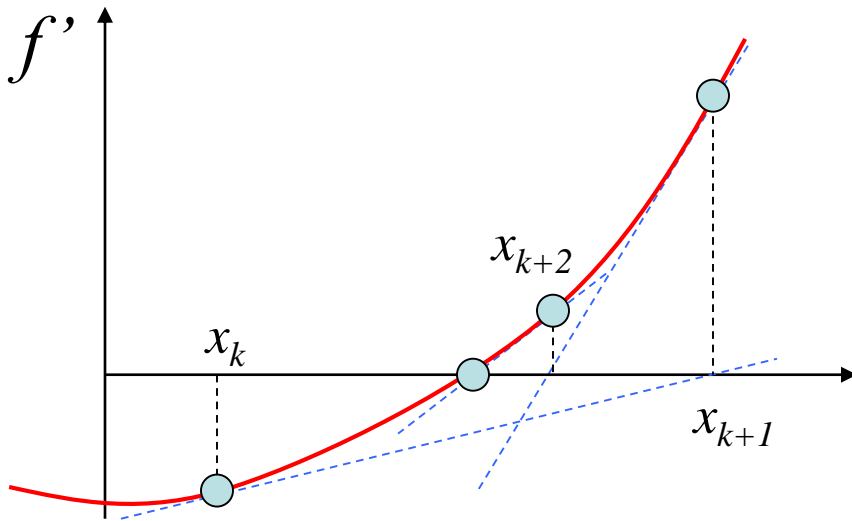
- Optimizasyon problemi iteratif olarak çözer.
- Çözüm için $f''(x) \neq 0$ olmamalı.



$$x_1 = x_0 - \frac{f'(x_0)}{f''(x_0)} \rightarrow x_2 = x_1 - \frac{f'(x_1)}{f''(x_1)} \rightarrow \dots \rightarrow x_k = x_{k-1} - \frac{f'(x_{k-1})}{f''(x_{k-1})}$$

NEWTON-RAPHSON METODU

- Tüm metotların en iyi yakınsayanıdır:



Örnek

$$\text{Max } f(x) = 12x - 3x^4 - 2x^6$$

Select $\varepsilon = 0.00001$, and choose $x_1 = 1$.

Iteration i	x_i	$f(x_i)$	$f'(x_i)$	$f''(x_i)$	x_{i+1}
1					
2					
3	0.84003	7.8838	-0.1325	-55.279	0.83763
4	0.83763	7.8839	-0.0006	-54.790	0.83762

Örnek-2

- Use Newton's method to minimize $f(x) = 0.5 - x \exp(-x^2)$
- First and second derivatives of f are given by

$$f'(x) = (2x^2 - 1) \exp(-x^2)$$

and

$$f''(x) = 2x(3 - 2x^2) \exp(-x^2)$$

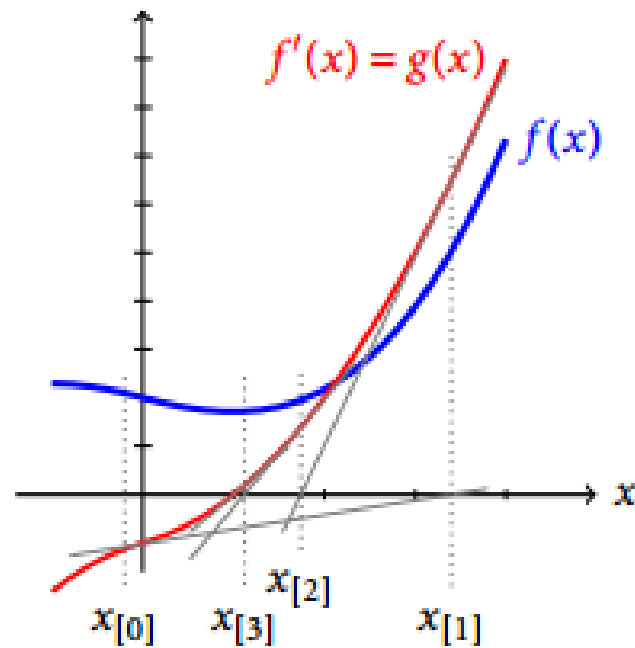
- Newton iteration for zero of f' is given by

$$x_{k+1} = x_k - (2x_k^2 - 1)/(2x_k(3 - 2x_k^2))$$

- Using starting guess $x_0 = 1$, we obtain

x_k	$f(x_k)$
1.000	0.132
0.500	0.111
0.700	0.071
0.707	0.071

Örnek-3



$$f(x) = \frac{1}{3}x^3 + \frac{1}{4}x^2 - 0.5x + 1$$

$$f'(x) = x^2 + \frac{1}{2}x - 0.5$$

$$f''(x) = 2x + \frac{1}{2}$$

x	$f(x)$	$f'(x)$ $= g(x)$	$f''(x)$ $= g'(x)$
$x[0] = -0.1$	1.0522	-0.54	0.3
$x[1] = 1.7$	2.5102	3.24	3.9
$x[2] = 0.8692$	0.9732	0.6902	2.2385
$x[3] = 0.5609$	0.8570	0.0951	1.6218
$x[4] = 0.5023$	0.8542	0.0034	1.5046
$x[5] = 0.5000$	0.8542	0.0000	1.5000

SECANT METODU

Newton metodunda:

- 2. türev bilgisinin elde edilmesi çok hesap gerektirebilir. (özellikle çok değişkenli problemlerde).
- 2. türevin yaklaşıklığının kullanılması hesapları çok kolaylaştırır.

2. türevin, 1. türev bilgisi kullanarak sonlu farklarla yaklaşıklığı:

$$f''(x) = \frac{f'(x_k) - f'(x_{k-1})}{x_k - x_{k-1}}$$

2. türev ifadesi Newton-Rapson formülünde yerine konursa Secant metodu elde edilir

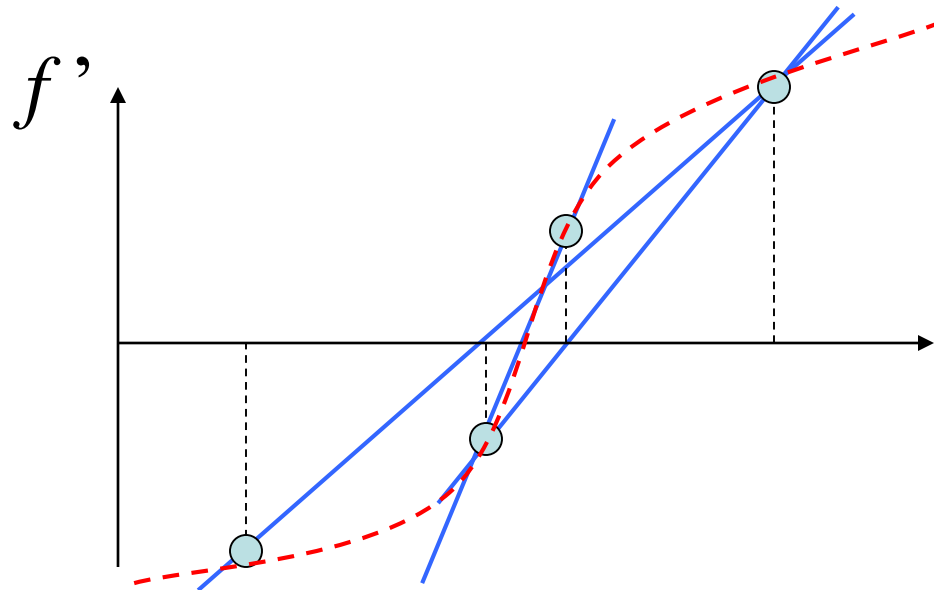
$$x_{k+1} = x_k - \frac{x_k - x_{k-1}}{f'(x_k) - f'(x_{k-1})} f'(x_k)$$

Not: Secant metodu, yüksek dereceli problemleri Newton-Raphson metodu gibi iteratif çözer.

SECANT METHODU

f' kökünü bulma temellidir.

Lineer interpolasyon kullanır



Örnek

Minimize

$$f(x) = -\frac{1}{2}x^4 - x^3 + 6x^2 + 6x + 4.$$

Then

- gradient

$$f'(x) = -2x^3 - 3x^2 + 12x + 6$$

gives secant iterations

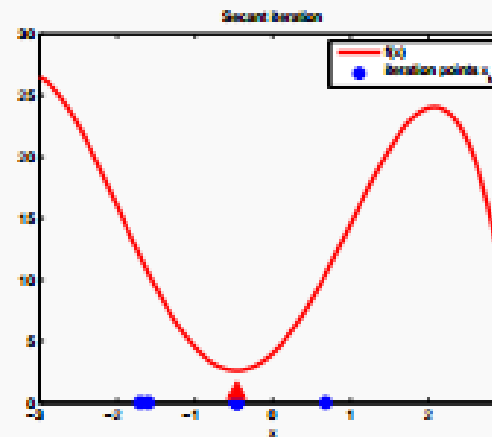
$$\begin{aligned}x_0 &= -1.6000, & x_1 &= 0.6820, \\x_2 &= -0.4345, & x_3 &= -0.4709, \\x_4 &= -0.4630, & x_5 &= -0.4630\end{aligned}$$

initial condition $x_0 = -1.6$

initial condition $x_{-1} = -1.7$

convergence to $x^* = -0.4630$

gradient $f'(x^*) = 5.96 \times 10^{-5}$



- Tek-değişken yöntemler
 - 0. derece (sadece f)
 - 1. derece (f ve f')
 - 2. derece (f , f' ve f'')

<ul style="list-style-type: none"> • Dichotomous sectioning • Fibonacci sectioning • Golden ratio sectioning • Quadratic interpolation 	0 th order
<ul style="list-style-type: none"> • Cubic interpolation • Bisection method • Secant method 	1 st order
<ul style="list-style-type: none"> • Newton method 	2 nd order

METOTLAR HAKKINDA GENEL DEGERLENDIRME

Golden Section, Bisection ve Polinom metotlari çözüm bölgesini sürekli belirli oranda daraltarak optimumu arastirirlar.

Golden Section:

- Fonksiyon degerlerini kullanir.
- yakinsama hizlari çözülen problemin tipinden bagimsizdir.
- Çözüm garantidir (fonksiyon sürekli olsun veya olmasin).

Bisection

- Türev bilgisini kullanir (fonksiyon 1. dereceden sürekli olmak zorunda).
- yakinsama hizi çözülen problemin tipine baglidir.

Polinom metodu

- Fonksiyon degerlerini kullanir.
- yakinsama hizi çözülen problemin tipine baglidir.
- Keskin dönüşlü problemlerde zorlanirlar (fonksiyon 1. ve 2. dereceden sürekli olmak zorunda).

METOTLAR HAKKINDA GENEL DEGERLENDIRME

Newton Raphson ve Secant metotlari bir noktadan baslayarak adim adim (iteratif olarak) optimumu arastirirlar.

Newton Raphson:

- Fonksiyonun 1. ve 2. türevlerini kullanir.
- yakinsama hizi çözülen problemin tipine baglidir.
- Keskin dönüslü problemlerde zorlanirlar (fonksiyon 1. ve 2. dereceden sürekli olmak zorunda).
- Çözüm garanti degildir

Secant metodu:

- Fonksiyonun 1. türevini kullanir.
- yakinsama hizi çözülen problemin tipine baglidir.
- Keskin dönüslü problemlerde zorlanirlar (fonksiyon 1. dereceden sürekli olmak zorunda).
- Çözüm garanti degildir.

Not: Optimizasyon problemlerinin çözümünde genellikle Polinom veya Golden Section kullanilir.

Kaynaklar

- Hasan KURTARAN, “Mühendislik tasarımların optimizasyonu”, 2005
- Edwin K.P. Chong Stanislaw H.Zak, “An Introduction to Optimization”, 2001, John Wiley&Sons
- Jorge Nocedal, Stephan J. Wright, “Numerical Optimization”, 1999, Springer-Verlag
- R. Fletcher, “Practical Methods of Optimization”, 1987, John Wiley&Sons
- David G.Luenberger, “Introduction to Linear and Nonlinear Programming”, 1987, Addison-Wesley
- S.G. Nash, A. Sofer, “Linear and Nonlinear programming”, 1996, McGraw Hill
- Dimitri P. Bertsekas, “Constrained Optimization and Lagrange Multiplier Methods”, 1982, Academic Press
- Abbas Azimli, “Matematiksel Optimizasyon”, 2011, Papatya Yayınevi