YMH 214 SAYISAL ANALİZ

Dr. Öğretim Üyesi Bihter DAŞ

Fırat Üniversitesi Teknoloji Fakültesi Yazılım Mühendisliği

4.Hafta

LİNEER OLMAYAN (NONLİNEER) DENKLEMLERİN ÇÖZÜMÜ

Lineer Olmayan Denklemlerin Çözümü

KAPALI YÖNTEMLER:

- Grafik yöntemi
- Bisection (Ikiye bölme yöntemi)
- Regula Falsi yöntemi (Yer değiştirme yöntemi)

AÇIK YÖNTEMLER:

- Fixed Point Iteration yöntemi (Sabit nokta iterasyon yöntemi)
- Newton Rapson yöntemi
- Secant yöntemi

REGULA FALSI(YER DEĞİŞTİRME YÖNTEMİ)
FALSE POSITION METHOD(DOĞRUSAL INTERPOLASYON YÖNTEMİ)

- Bazı problemlerin Bisection yöntemiyle çözümü çok uzun sürer. Çözümü hızlandırmak için **Yer Değiştirme** yöntemi kullanılabilir.
- Bu yöntemin Bisection yönteminden tek farkı **orta değerin hesaplanmasındadır.**
- **a** ve **b** değerleri için **f (x)** fonksiyonunun değerleri **f(a)** ve **f(b)** ters işaretli ise

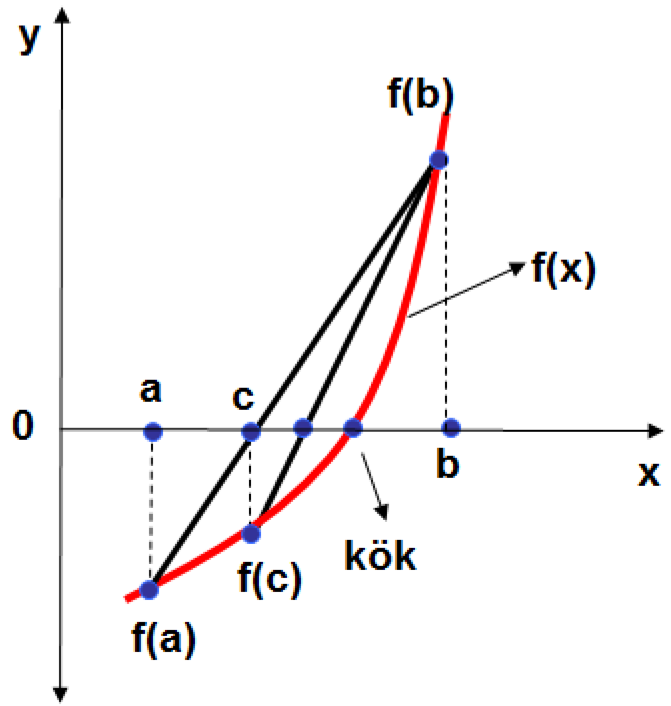
f(a)·f(b) < 0 bu aralıkta bir **kök** vardır.

- Bu yöntemde **(a, b)** aralığında; **fonksiyon,** uygun **bir doğru** ile **yer değiştirilerek** kök aranır.
- (**a**, **f(a)**) ve (**b** **f(b)**) **noktaları arasında çizilen doğrunun x** eksenini kesim noktası **c** olarak alınır ki bu genelde kök değerine daha yakındır.

Regula Falsi (Yer Değiştirme Yöntemi)

Fonksiyonun **f(a)** ile **f(b)** arasında kalan yayı **doğru halinde getirildiğinde** **x** ekseninini kesen **c** noktası **kök** değerine daha yakındır.

f(a) ·f(c) > 0

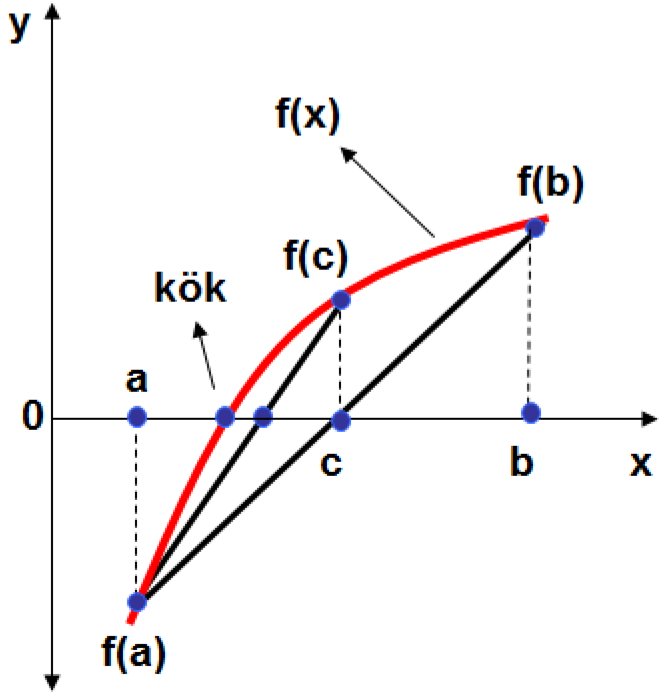


c ile **a** aynı tarafta (**f(a) ·f(c) > 0**) ise **kök** **c** ile **b** arasında aranır.

Regula Falsi (Yer Değiştirme Yöntemi)

Fonksiyonun **f(a)** ile **f(b)** arasında kalan yayı **doğru halinde getirildiğinde** **x** eksenini kesen **c** noktası **kök** değerine daha yakındır.

f(a) ·f(c) < 0

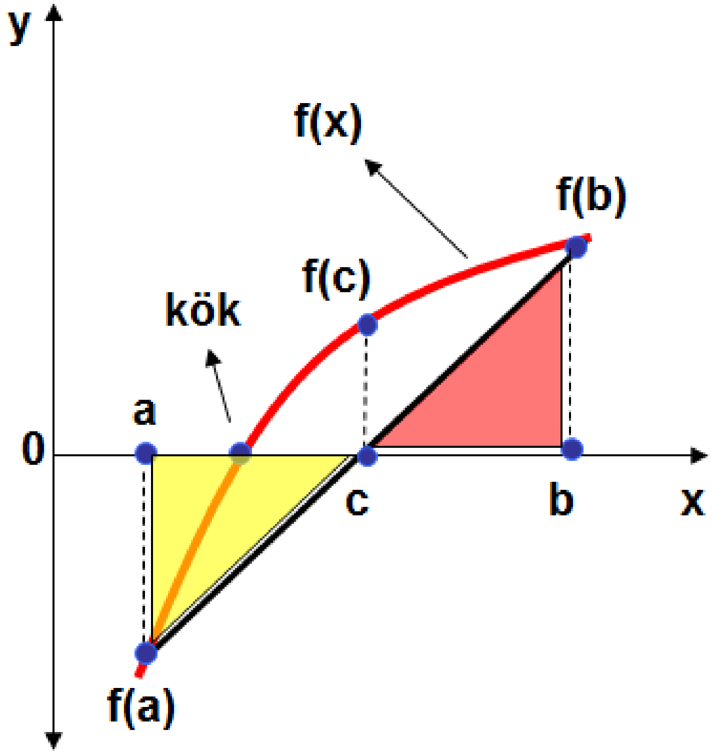


c ile **b** aynı tarafta (**f(a) ·f(c) < 0**) ise **kök** **a** ile **c** arasında aranır.

Regula Falsi (Yer Değiştirme Yöntemi)

c noktasının hesabı

$f(a) \cdot f(c) < 0$



a, c, **f(a)** üçgeni ile b, c, **f(b)** üçgeni **benzer**dir.

$$\boxed{\frac{|ac|}{|bc|} = \frac{|c\mathrm{f}(a)|}{|c\mathrm{f}(b)|} = \frac{|a\mathrm{f}(a)|}{|b\mathrm{f}(b)|}}$$

$$\boxed{\frac{c-a}{b-c} = \frac{-\mathrm{f}(a)}{\mathrm{f}(b)}}$$

$$\boxed{c = \frac{b \cdot \mathrm{f}(a) - a \cdot \mathrm{f}(b)}{\mathrm{f}(a) - \mathrm{f}(b)}}$$

Yöntemin uygulanmasında izlenecek yol aşağıdaki gibi özetlenebilir.

1) Kökün bulunduğu aralık için **alt** (**a**) ve **üst** (**b**) değerler tahmin edilir ve **$f(a).f(b) < 0$** şartı aranır.

2) **c** değeri hesaplanır.

3) **f(c)** değeri hesaplanır
   Eğer **$f(c) = 0$** ise **kök c** 'dir.
   Eğer **$f(c) \neq 0$** ise **işleme devam** edilir

4) **$f(a) \cdot f(c) > 0$ ise** **$a = c$**
   $f(a) \cdot f(c) < 0$ ise **$b = c$**

alınarak 1. basamağa geri dönülür.

Yer Değiştirme yönteminde iterasyona iki şekilde son verilir.

1) **f(c)=0** olunca işleme son verilir . Kök **c** 'dir.

2) **$| E_b | < \varepsilon$** ise işleme son verilir.

$$\boxed{E_b = \frac{\text{Son deger - Bir önceki deger}}{\text{Son deger}} = \frac{x_{0,k+1} - x_{0,k}}{x_{0,k+1}}}$$

```
clear all;close all;clc
fprintf('Regula Falsi yöntemini kullanarak f(x)=x^3-2 denkleminin köklerini bulma');
a=1;
b=2;
tol=0.00001;
fprintf('\n iter    a           b          xr      y(xr)\n');
for i=1:50
fonka=a^3-2;
fonkb=b^3-2;
xr=(a*fonkb-b*fonka)/(fonkb-fonka);
fonkxr=xr^3-2;
fprintf('%4.1f  %7.4f   %7.4f  %7.4f   %7.4f \n',i,a,b,xr,fonkxr);
if abs(fonkxr)<tol
    break;
end
if fonka*fonkxr<0
b=xr;
fonkb=fonkxr;
else
a=xr;
fonka=fonkxr;
end
end
disp('Iterasyon sayısı')
i
disp('Denklemin kökü');
format long
xr
disp('Fonksiyonun kökteki değeri')
fonkxr
```

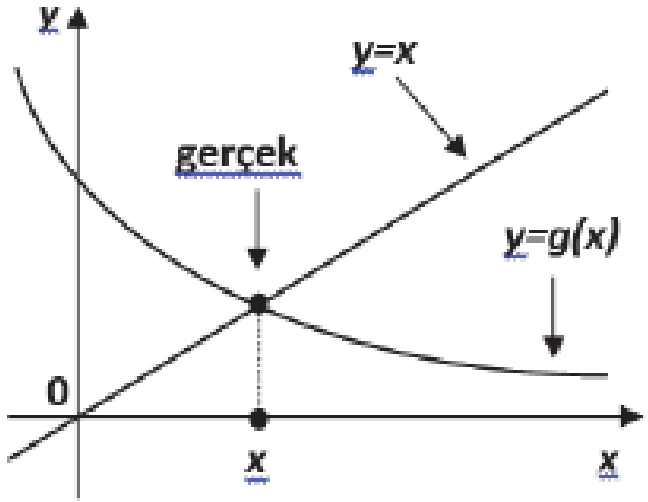
Program Çıktısı

```
Regula Falsi yöntemini kullanarak f(x)=x^3-2 denkleminin köklerini bulma
 iter    a          b          xr        y(xr)
 1.0    1.0000     2.0000     1.1429    -0.5073
 2.0    1.1429     2.0000     1.2097    -0.2299
 3.0    1.2097     2.0000     1.2388    -0.0987
 4.0    1.2388     2.0000     1.2512    -0.0414
 5.0    1.2512     2.0000     1.2563    -0.0172
 6.0    1.2563     2.0000     1.2584    -0.0071
 7.0    1.2584     2.0000     1.2593    -0.0029
 8.0    1.2593     2.0000     1.2597    -0.0012
 9.0    1.2597     2.0000     1.2598    -0.0005
10.0    1.2598     2.0000     1.2599    -0.0002
11.0    1.2599     2.0000     1.2599    -0.0001
12.0    1.2599     2.0000     1.2599    -0.0000
13.0    1.2599     2.0000     1.2599    -0.0000
14.0    1.2599     2.0000     1.2599    -0.0000
```

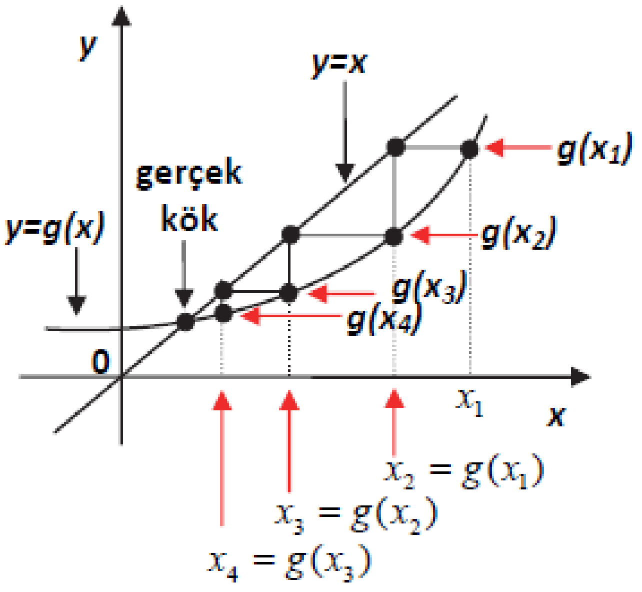
```
Iterasyon sayısı

i =

    14

Denklemin kökü

xr =

   1.259919790896880

Fonksiyonun kökteki değeri

fonkxr =

    -5.995598227226395e-06
```

FİXED POINT-SABİT NOKTA ITERASYON

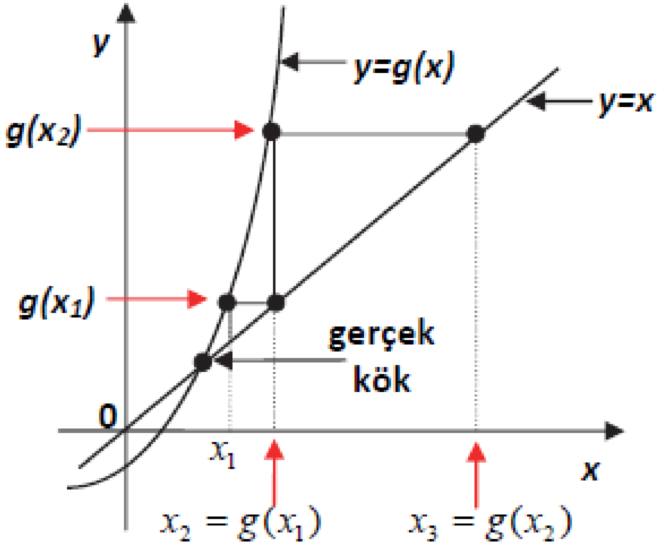
Sabit noktalı iterasyon f(x)=0 eşitsizliğinin çözülmesi için kullanılan bir yöntemdir. Şekilde yöntemin grafiksel gösterimi verilmektedir. Bu yöntemde eşitlik x=g(x) $x_{i+1}=g(x_i)$ formunda yazılır. Çözüm f(x)=0 eşitliğini sağlayan x değeridir. Şekil incelenirse y=x ve y=g(x) eğrilerinin 'sabit nokta' olarak isimlendirilen kesişme noktası aranan çözümü vermektedir.



FİXED POINT-SABİT NOKTA ITERASYON



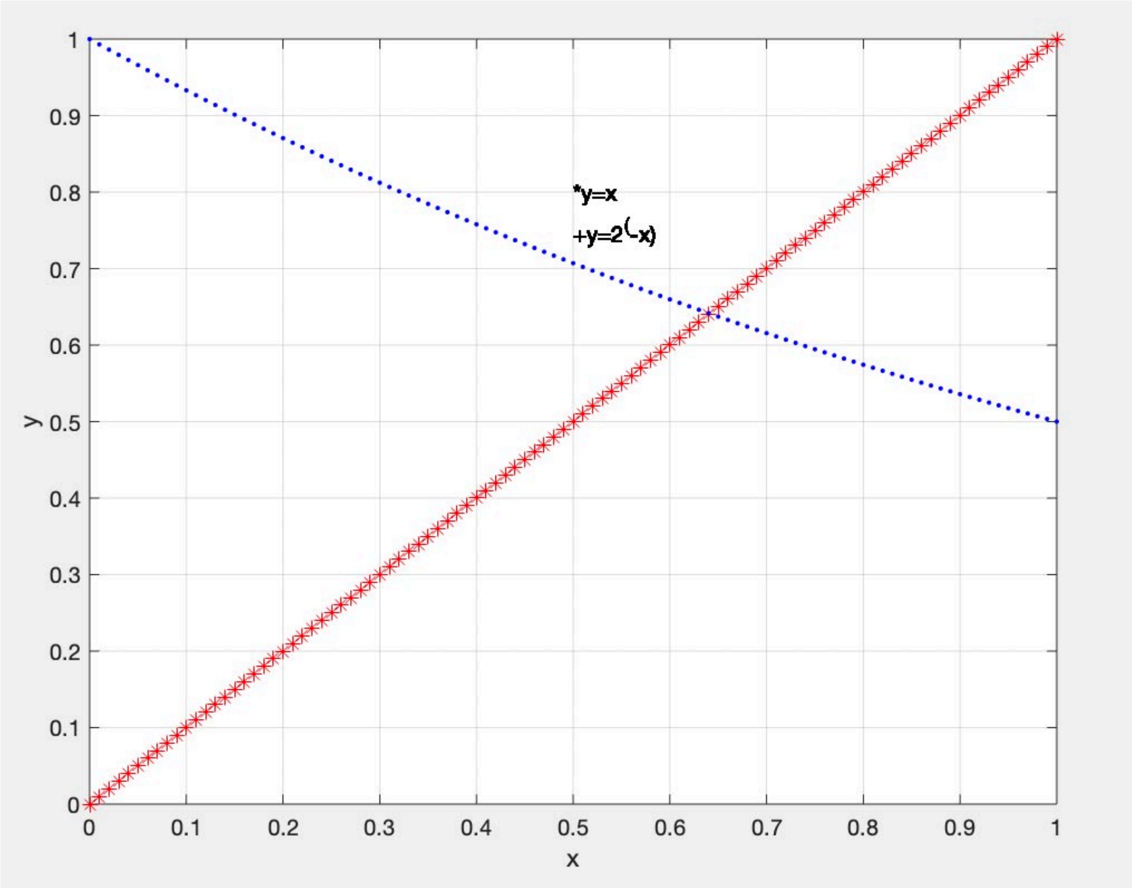
a) Yakınsama



b) Iraksama

Fixed Point Yöntemi Matlab Çözümü

```
clear all;close all;clc
fprintf('Fixed Point yöntemini kullanarak f(x)=x-2^(-x) denkleminin köklerini bulma \n');
x1=0;
tol=0.1;

for x1=0:0.01:1
    y=x1;
    yy=2^(-x1);
plot(x1,y,'r*',x1,yy,'b.')
hold on
grid on
xlabel('x')
ylabel('y')
text(0.5,0.8,'*y=x')
text(0.5,0.75,'+y=2^(-x)')
end
 x1=0;
 fprintf('Iter      x1       x2         Ea         ear    \n')
 for i=1:50
     x2=2^(-x1);
     Ea=abs(x2-x1);
     ear=Ea/abs(x2);
     fprintf('%4.1f   %7.4f   %7.4f   %7.4f   %7.4f\n',i,x1,x2,Ea,ear);
     if abs(x2-x1)<tol
         break;
     else
         x1=x2;
     end
 end
 disp('Denklemin koku');
 disp([x2])
```




```
Fixed Point yöntemini kullanarak f(x)=x-2^(-x) denkleminin köklerini bulma
Iter       x1        x2          Ea         ear
 1.0     0.0000    1.0000    1.0000    1.0000
 2.0     1.0000    0.5000    0.5000    1.0000
 3.0     0.5000    0.7071    0.2071    0.2929
 4.0     0.7071    0.6125    0.0946    0.1544
Denklemin koku
   0.612547326536066
```