Numerical Analysis
GC / SIE
Nonlinear Equations

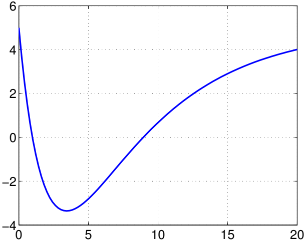
Daniel Kressner
Chair for Numerical Algorithms and HPC
Institute of Mathematics, EPFL
daniel.kressner@epfl.ch

EPFL

Objective

Determine zeros (also called roots) of a function f:

$$f(x) = 0.$$



In practice:

- "solution by hand" or by computer algebra package (Mathematica, Maple, ...) or by ChatGPT only in exceptional cases possible;
- often f very complicated, sometimes only given implicitly as a Python function.

Example 1: Oxygen level in river

- Simple model to measure oxygen level c (mg/L) in a river downstream from a sewage discharge:

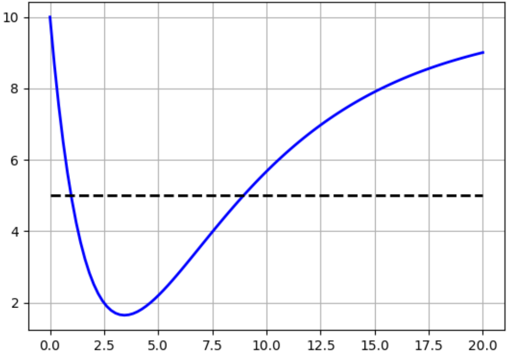
$$c(x) = 10 - 20\left(e^{-0.15x} - e^{-0.5x}\right).$$

  x: downstream distance from discharge (km).

- Oxygen level below 5 mg/L considered harmful.
- Which region of the river is harmful?

```
import numpy as np
import matplotlib.pyplot as plt
f=lambda x: 10 - 20*(np.exp(-0.15*x) - np.exp(-0.5*x));
x=np.linspace(0,20,100)
plt.plot(x,f(x),color='b',linewidth=2)
plt.plot(x,5*np.ones(100),'--k',linewidth=2)
plt.grid(True)
plt.show()
```

Example 1: Graph of $c(x)$



Zoom in to determine zeros of functions visually.

Example 2: Investment fund

- Yearly investment of $v = 1000$ CHF.
- Return: $M = 6000$ CHF after $n = 5$ years.
- Task:
  Compute I = average yearly rate of interest of this investment.

Relation between M, v, I, and n:

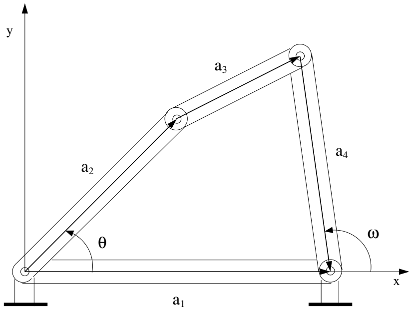
$$M = v \sum_{k=1}^{n} (1 + I)^k = v \frac{1 + I}{I} \left[(1 + I)^n - 1 \right].$$

$\leadsto$ Nonlinear equation:

$$f(I) = M - v \frac{1 + I}{I} \left[(1 + I)^n - 1 \right] = 0.$$

No explicit formula for I available!

Example 3: Statics



- Mechanical system of four rigid rods with planar linkage.
- Constraint: Vector identity

$$\mathbf{a}_1 - \mathbf{a}_2 - \mathbf{a}_3 - \mathbf{a}_4 = 0.$$

- a_i = length of rod $\mathbf{a}_i$.

Then:

$$\frac{a_1}{a_2}\cos(\omega) - \frac{a_1}{a_4}\cos(\theta) - \cos(\omega - \theta) = -\frac{a_1^2 + a_2^2 - a_3^2 + a_4^2}{2a_2a_4}.$$

Task: For given $\omega \in [0, \pi]$, determine corresponding value of θ.

No explicit formula for θ available!

Example 4: State equation of a gas

- Task: Determine volume V occupied by a gas at temperature T and pressure p.
- State equation:

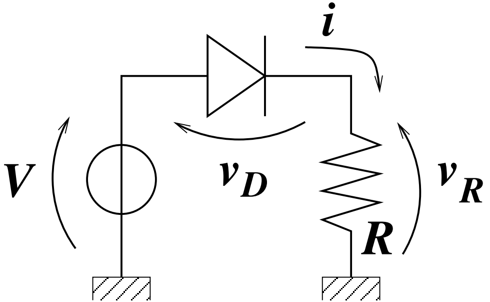
$$\left[p + a\left(\frac{N}{V}\right)^2\right](V - Nb) = kNT \ ,$$

with

  - a, b = gas-specific constants;
  - k = Boltzmann constant;
  - N = number of molecules contained in V.

No explicit formula for V available!

Example 5: Electrical circuit

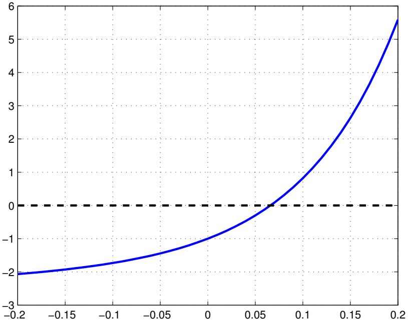


See lecture notes: Voltage v_D across diode is root of the *nonlinear equation*

$$f(x) = 0, \qquad \text{where} \quad f(x) = Ri_0\left(e^{x/v_0} - 1\right) + x - V.$$

Example 5: Electrical circuit

Graph of f for $i_0 = 1$, $v_0 = 0.1$, $R = 1$, $V = 1$



Run bisection with $a = -0.2$, $b = 0.2$, tol $= 10^{-8}$, and maximal iterations 100 (before giving up)

```
from bisection import bisection
import numpy as np
i0=1; v0=0.1; R=1; V=1;
f = lambda x:  R*i0*(np.exp(x/v0)-1)+x-V;
zero,res,niter,inc,err=bisection(f,-0.2,0.2,1e-8,100)
```

Output:

```
zero 0.06596105694770812
res 7.085173225895858e-08
iterations 25
```

What happens when reducing tol to 10^{-100}?

Run bisection with $a = -0.2$, $b = 0.2$, tol $= 10^{-100}$, and maximal iterations 100 (before giving up)

```
from bisection import bisection
import numpy as np
i0=1; v0=0.1; R=1; V=1;
f = lambda x:  R*i0*(np.exp(x/v0)-1)+x-V;
zero,res,niter,inc,err=bisection(f,-0.2,0.2,1e-100,100)
```

Output:

```
zero 0.06596105346440555
res 0.0
iterations 53
```

Is this really the exact root? What is so special about 53?

Fixed point iteration: $f(x) = 1 - xe^x = 0$

$$\Phi_1(x) = e^{-x}, \quad \Phi_2(x) = \frac{1+x}{1+e^x}, \quad \Phi_3(x) = x + 1 - xe^x.$$

First 10 iterates from $x^{(0)} = 0.5$:

k	$x^{(k+1)} := \Phi_1(x^{(k)})$	$x^{(k+1)} := \Phi_2(x^{(k)})$	$x^{(k+1)} := \Phi_3(x^{(k)})$
0	0.500000000000000	0.500000000000000	0.500000000000000
1	0.606530659712633	0.566311003197218	0.675639364649936
2	0.545239211892605	0.567143165034862	0.347812678511202
3	0.579703094878068	0.567143290409781	0.855321409174107
4	0.560064627938902	0.567143290409784	-0.156505955383169
5	0.571172148977215	0.567143290409784	0.977326422747719
6	0.564862946980323	0.567143290409784	-0.619764251895580
7	0.568438047570066	0.567143290409784	0.713713087416146
8	0.566409452746921	0.567143290409784	0.256626649129847
9	0.567559634262242	0.567143290409784	0.924920676910549
10	0.566907212935471	0.567143290409784	-0.407422405542253

Fixed point iteration: $f(x) = 1 - xe^x = 0$

$$\Phi_1(x) = e^{-x}, \quad \Phi_2(x) = \frac{1+x}{1+e^x}, \quad \Phi_3(x) = x + 1 - xe^x.$$

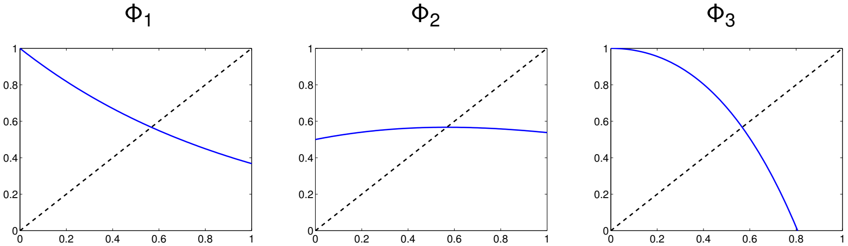
Error for the first 10 iterates from $x^{(0)} = 0.5$:

k	$\lvert x^{(k+1)} - x^\star \rvert$	$\lvert x^{(k+1)} - x^\star \rvert$	$\lvert x^{(k+1)} - x^\star \rvert$
0	0.067143290409784	0.067143290409784	0.067143290409784
1	0.039387369302849	0.000832287212566	0.108496074240152
2	0.021904078517179	0.000000125374922	0.219330611898582
3	0.012559804468284	0.000000000000003	0.288178118764323
4	0.007078662470882	0.000000000000000	0.723649245792953
5	0.004028858567431	0.000000000000000	0.410183132337935
6	0.002280343429460	0.000000000000000	1.186907542305364
7	0.001294757160282	0.000000000000000	0.146569797006362
8	0.000733837662863	0.000000000000000	0.310516641279937
9	0.000416343852458	0.000000000000000	0.357777386500765
10	0.000236077474313	0.000000000000000	0.974565695952037

Fixed point iteration: $f(x) = 1 - xe^x = 0$

- $\Phi_1(x) = e^{-x}$ OK convergence;
- $\Phi_2(x) = \frac{1+x}{1+e^x}$ fantastic convergence;
- $\Phi_3(x) = x + 1 - xe^x$ very disappointing.

What is the difference?



$|\Phi_1'(\alpha)| = 0.5671$ $|\Phi_2'(\alpha)| = 0$ $|\Phi_3'(\alpha)| = 1.7632$

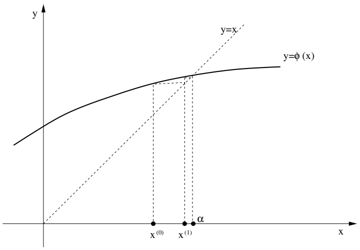
$\alpha = 0.567143\ldots$ is the exact root.

Illustration of Theorem 1.1

Demonstration how $\mid \Phi'(\alpha) \mid$ influences convergences.

Cases of convergence:

$0 < \Phi'(\alpha) < 1,$ $\qquad$ $-1 < \Phi'(\alpha) < 0.$



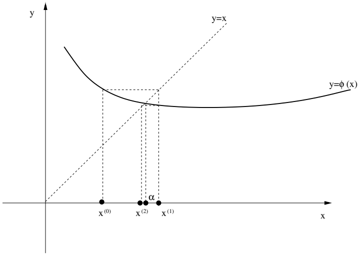
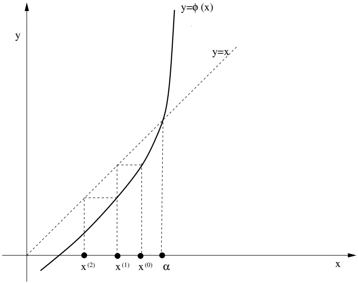


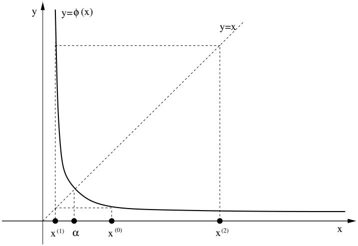
Illustration of Theorem 1.1

Demonstration how $|\ \Phi'(\alpha)\ |$ influences convergences.

Cases of divergence:

$\Phi'(\alpha) > 1$, $\qquad$ $\Phi'(\alpha) < -1$.



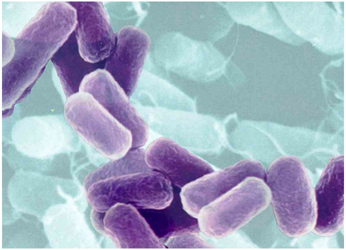


Example 6: Population growth

- x = number of individuals in a population (e.g., bacteria).
- Change of population from one generation x to the next generation x^+ described by

$$x^+ = \Phi(x) = xR(x),$$

where R describes growth or decrease.



Task: Determine whether (and which) stationary point approached by population after many generations.

Example 6: Population growth

Several different models for $R(x)$ available:

- Model of Malthus (Thomas Malthus 1766-1834):

$$x^+ = \Phi_1(x) = xR_1(x) \text{ with } R_1(x) = r, \quad r > 0 \text{ const.}$$

- Model of growth under limited resources (Pierre François Verhulst, 1804-1849):

$$x^+ = \Phi_2(x) = xR_2(x) \text{ with } R_2(x) = r/(1+x/K), \quad r > 0, K > 0 \text{ const.}$$

  Malthus' model takes into account that population growth is limited by available resources.

- Prey-predator model with saturation:

$$x^+ = \Phi_3(x) = xR_3(x) \text{ with } R_3(x) = rx/(1 + (x/K)^2).$$

  Verhulst's model under the presence of an enemy population.

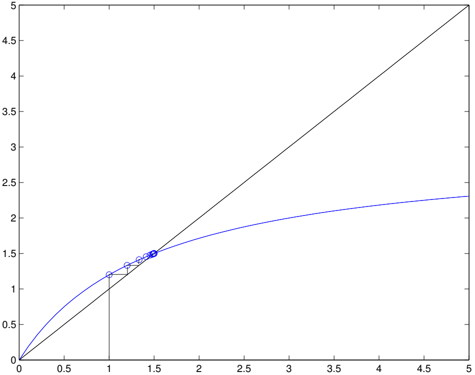
Example 6: Population growth

Example:

- Fixed point iteration for $\Phi_2(x) = rx/(1 + x/K)$ and $\Phi_3(x) = rx^2/(1 + (x/K)^2)$ with $K = 1.5$ and $r = 2$.
- Starting population: $x^{(0)} = 1$.
- Stationary points are $\alpha_2 = 1.5$ and $\alpha_3 = 3.9271$.

Example 6: Population growth

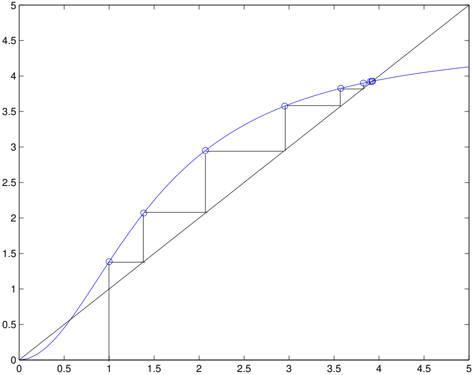
Function $\Phi_2(x)$:



$$
\begin{aligned}
x^{(0)} &= 1.0000, & \qquad |x^{(0)} - \alpha_2| &= 0.5000 \\
x^{(1)} &= 1.2000, & \qquad |x^{(1)} - \alpha_2| &= 0.3000 \\
x^{(2)} &= 1.3333, & \qquad |x^{(2)} - \alpha_2| &= 0.1667 \\
x^{(3)} &= 1.4118, & \qquad |x^{(3)} - \alpha_2| &= 0.0882
\end{aligned}
$$

Example 6: Population growth

Function $\Phi_3(x)$:



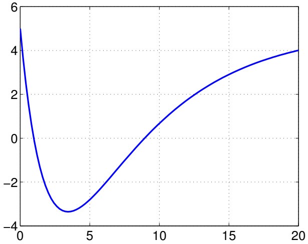
$$
\begin{aligned}
x^{(0)} &= 1.0000, & \qquad |x^{(0)} - \alpha_3| &= 2.9271 \\
x^{(1)} &= 1.3846, & \qquad |x^{(1)} - \alpha_3| &= 2.5424 \\
x^{(2)} &= 2.0703, & \qquad |x^{(2)} - \alpha_3| &= 1.8568 \\
x^{(3)} &= 2.9509, & \qquad |x^{(3)} - \alpha_3| &= 0.9761
\end{aligned}
$$

Convergence of Newton method: oxygen level

- Simple model to measure oxygen level c (mg/L) in a river downstream from a sewage discharge:

$$c(x) = 10 - 20\left(e^{-0.15x} - e^{-0.5x}\right).$$

x: downstream distance from discharge (km).



Graph of $c(x) - 5$

- Oxygen level below 5 mg/L considered harmful.
- Which region of the river is harmful?

Convergence of Newton method: oxygen level

Application of Newton requires:

$$c'(x) = 3e^{-0.15\,x} - 10e^{-0.5\,x}.$$

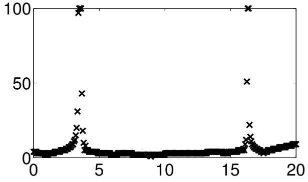
Newton method:

$$x^{(k+1)} = x^{(k)} - \frac{c(x^{(k)}) - 5}{c'(x^{(k)})}.$$

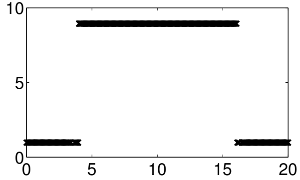
Starting value $x^{(0)} = 0 \rightsquigarrow$

$$\begin{aligned} x^{(1)} &= 0.7143 \\ x^{(2)} &= 0.9527 \\ x^{(3)} &= 0.9760 \\ x^{(4)} &= 0.9762 \end{aligned}$$

$\#$ iterations until $|f(x^{(k)})| \leq 10^{-4}$:



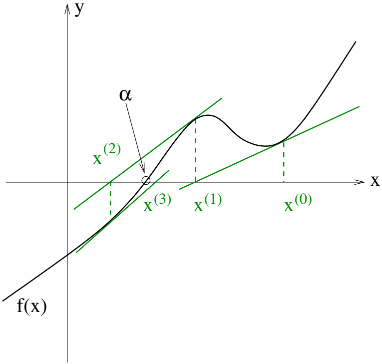
$x^{(k)}$ upon convergence:



Convergence of Newton method

When does Newton converge?

- depends on properties of function;
- depends on starting value.



System of two equations in two unknowns

$$\begin{cases} f_1(x_1, x_2) = x_1^2 + x_1 x_2 - 10 = 0, \\ f_2(x_1, x_2) = x_2 + 3x_1 x_2^2 - 57 = 0. \end{cases}$$

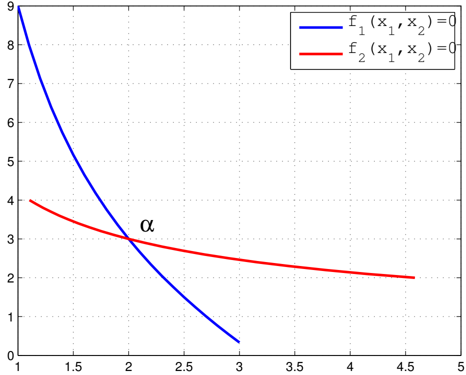
First equation implicitly defines the curve

$$f_1(x_1, x_2) = x_1^2 + x_1 x_2 - 10 = 0 \qquad \Longrightarrow \qquad x_2 = g_1(x_1) = \frac{10 - x_1^2}{x_1},$$

Second equation defines the curve

$$f_2(x_1, x_2) = x_2 + 3x_1 x_2^2 - 57 = 0 \qquad \Longrightarrow \qquad x_1 = g_2(x_2) = \frac{57 - x_2}{3x_2^2}.$$

System of two equations in two unknowns



System of two equations in two unknowns

```
import numpy as np
from newtonsys import *

f=lambda x: np.array([x[0]**2+x[0]*x[1]-10,
                      x[1]+3*x[0]*x[1]**2-57])
df= lambda x: np.array([[2*x[0]+x[1], x[0]],
                        [3*x[1]**2,1+6*x[0]*x[1]]])
x0=[3,4]
x,inc,niter=newtonsys(f,df,x0,1e-8,1000);
#OUTPUT
print('x= '+str(x[-1]))
print('number of iterations '+str(niter))
print('increments=' +str(inc))
#     x= [2. 3.]
# number of iterations= 5
# increments =[1.11487660e+00 3.27957413e-01
#       3.58120064e-02 1.96157782e-04
#  1.10946117e-09]
```