# Analyse Numérique 1

Salem Nafiri

Ecole Hassania des Travaux Publics

# Résolution des équations non linéaires
# $F(x)=0$

DEFINITION 2.1. A function $f\colon \mathbb{R}^n \to \mathbb{R}$ is defined as being *nonlinear* when it does not satisfy the *superposition principle* that is

$$f(x_1 + x_2 + \ldots) \neq f(x_1) + f(x_2) + \ldots$$

Now that we know what the term *nonlinear* refers to we can define a *system of non-linear equations*.

DEFINITION 2.2. A *system of nonlinear equations* is a set of equations as the following:

$$\begin{gathered} f_1(x_1, x_2, \ldots, x_n) = 0, \\ f_2(x_1, x_2, \ldots, x_n) = 0, \\ \vdots \\ f_n(x_1, x_2, \ldots, x_n) = 0, \end{gathered}$$

where $(x_1, x_2, \ldots, x_n) \in \mathbb{R}^n$ and each $f_i$ is a nonlinear real function, $i = 1, 2, \ldots, n$.

# Analytical Solutions

Analytical solutions are available for special equations only.

$$\text{Analytical solution of } \; a\,x^2 + b\,x + c = 0$$

$$roots = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

$$\text{No analytical solution is available for: } \quad x - e^{-x} = 0$$

# Graphical Illustration

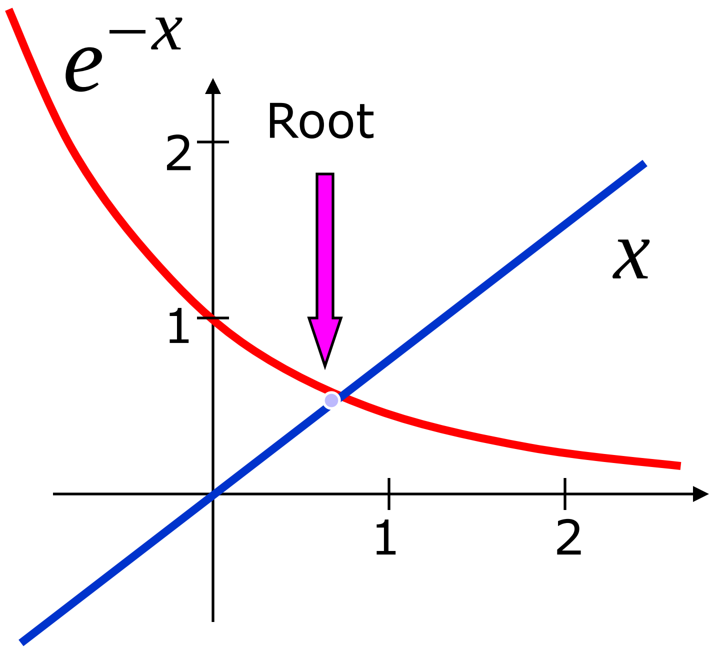
- Graphical illustration are useful to provide an initial guess to be used by other methods

$$Solve$$

$$x = e^{-x}$$

$$The \;\; root \in [0,1]$$

$$root \approx 0.6$$



# Solution Methods

Many methods are available to solve nonlinear equations

- ❑ Bisection Method
- ❑ Newton's Method
- ❑ Fixed point iterations

These will be covered.

- Secant method
- False position Method
- Muller's Method
- Bairstow's Method
- ……….

# Bisection Method

# Bisection Method

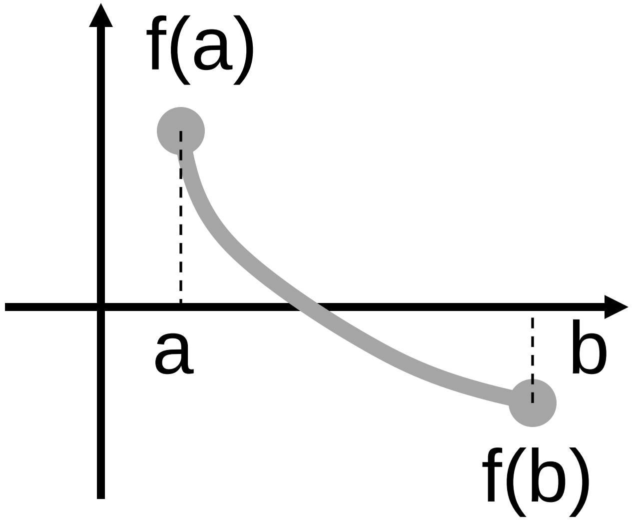
- The **Bisection method** is one of the simplest methods to find a zero of a nonlinear function.
- To use the Bisection method, one needs an initial interval that is known to contain a zero of the function.
- The method systematically reduces the interval. It does this by dividing the interval into two equal parts, performs a simple test and based on the result of the test half of the interval is thrown away.
- The procedure is repeated until the desired interval size is obtained.

# Intermediate Value Theorem

- Let f(x) be defined on the interval [a,b],

- <u>Intermediate value theorem:</u>

  if a function is continuous and f(a) and f(b) have different signs then the function has at least one zero in the interval [a,b]



# Bisection Algorithm

**Assumptions:**

- f(x) is continuous on [a,b]
- f(a) f(b) < 0

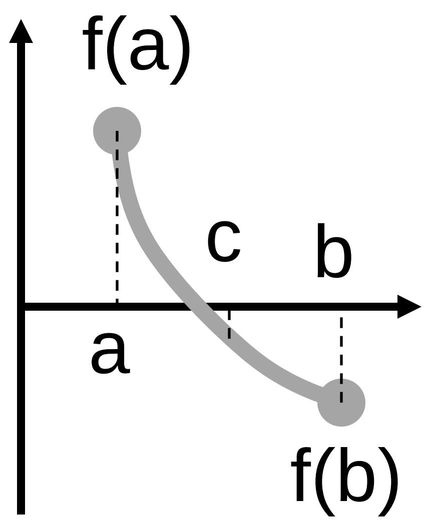
**Algorithm:**

**Loop**

1. Compute the mid point  c=(a+b)/2
2. Evaluate f(c )
3. If    f(a) f(c) < 0  then  new interval [a, c]

   If    f(a) f( c) > 0 then  new interval [c, b]

**End loop**



# Bisection Method

**Assumptions:**

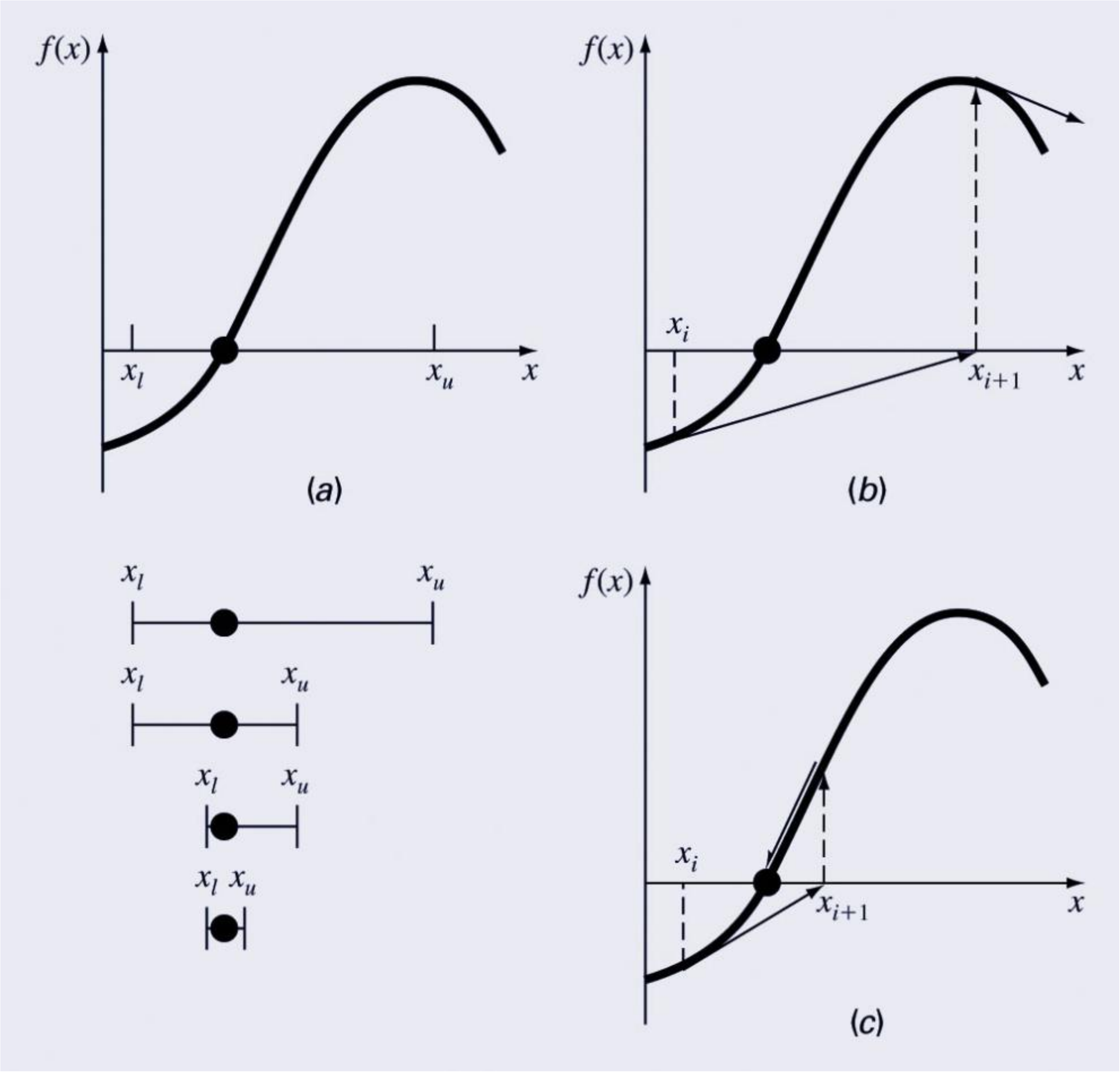
Given an interval [a,b]

f is continuous on [a,b]

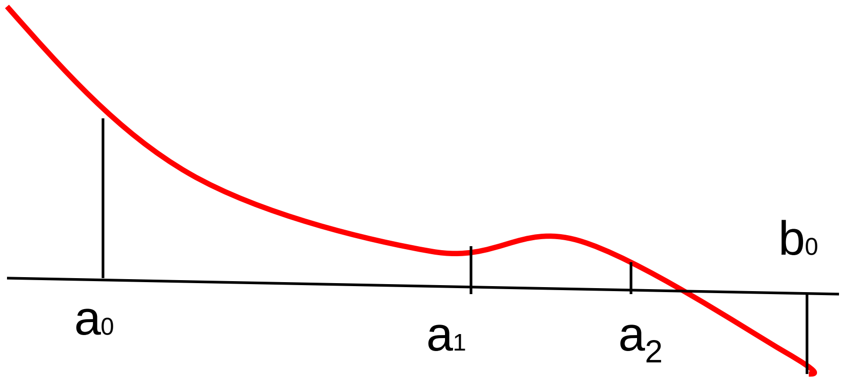
f(a) and f(b) have opposite signs.

These assumptions ensures the existence of at least one zero in the interval [a,b] and the bisection method can be used to obtain a smaller interval that contains the zero.

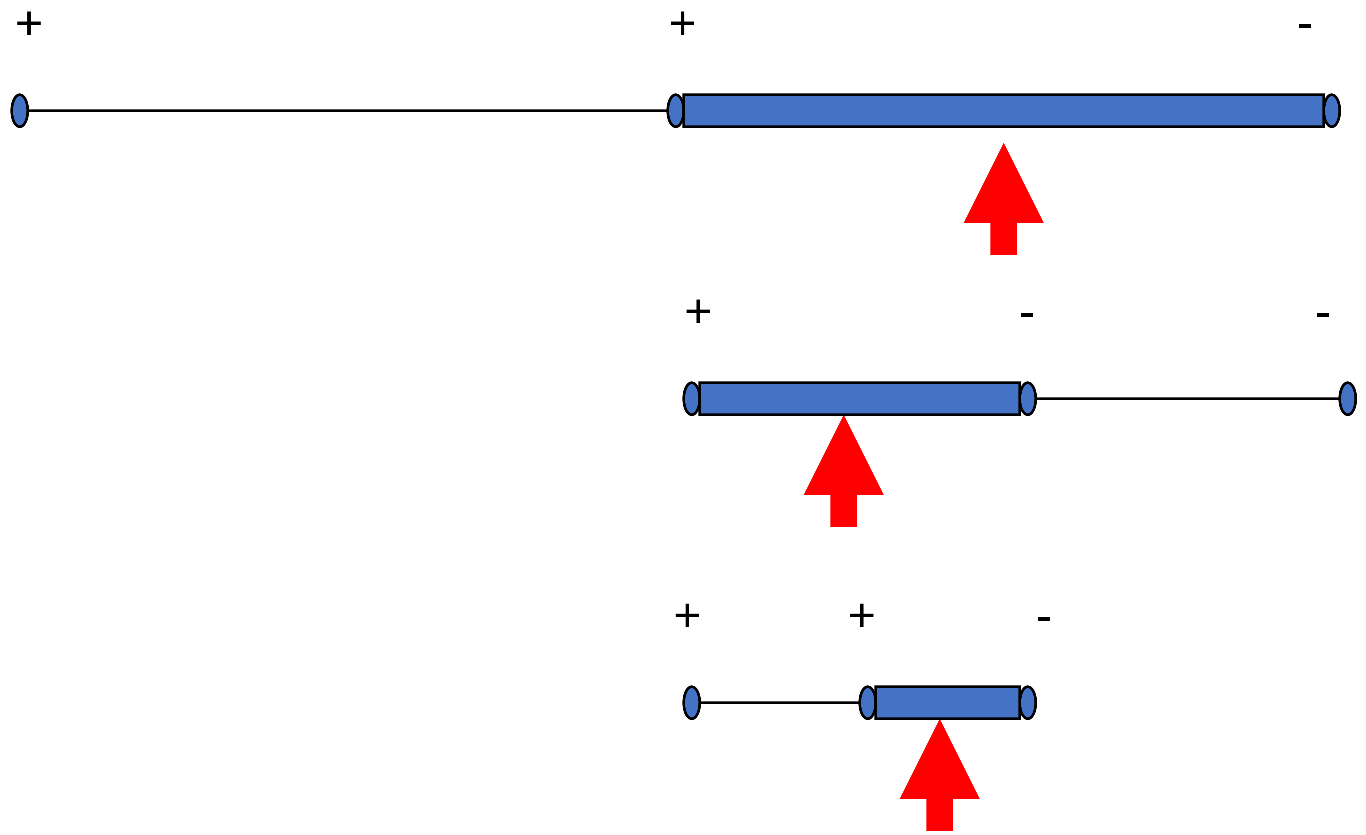
# Bisection Method



# Bisection Method



# Example



# Algorithm of Bisection Method

**Data**: f, a, b, eps

**Output**: alpha **(**approximation of the root of f on [a,b]**)**

**Step 1**: c= (a+b)/2 (generation of the sequence  (c_n))

**Step 2**: If |b-c|<eps then alpha:=c Stop.

**Step 3**: If f(a).f(b)<=0 then a:=c

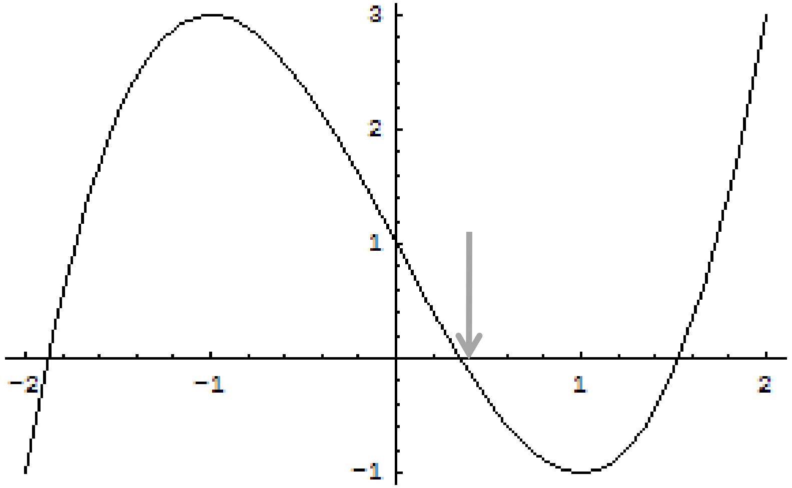
        else If b:=c

**Step 4**: go to step 1

# Example:

Can you use Bisection method to find a zero of

$f(x) = x^3 - 3x + 1$ in the interval [0,1]?




**Answer:**

$f(x)$ is continuous on [0,1]

$f(0) * f(1) = (1)(-1) = -1 < 0$

Assumptions are satisfied

Bisection method can be used

# Best Estimate and Error Level

Bisection method obtains an interval that is guaranteed to contain a zero of the function.

## Questions:

- What is the best estimate of the zero of ***f(x)***?
- What is the error level in the obtained estimate?

# Best Estimate and Error Level

The <u>best estimate</u> of the zero of the function ***f(x)*** after the first iteration of the Bisection method is the mid point of the initial interval:

$$Estimate\ of\ the\ zero:\ r = \frac{b+a}{2}$$

$$Error \leq \frac{b-a}{2}$$

# Stopping Criteria

Two common stopping criteria

1. Stop after a fixed number of iterations
2. Stop when the absolute error is less than a specified value

How are these criteria related?

# Stopping Criteria

$c_n$ : is the midpoint of the interval at the $n^{\text{th}}$ iteration

( $c_n$ is usually used as the estimate of the root).

r: is the zero of the function.

After $n$ iterations:

$$\left|error\right| = \left| r - c_n \right| \leq E_a^n = \frac{b-a}{2^n} = \frac{\Delta x^0}{2^n}$$

# Stopping Criteria

Let TOL >0 be a small number

- One can use either

$$|x_{n+1} - x_n| < TOL \quad \text{or} \quad |x_{n+1} - x_n| / |x_n| < TOL$$

Or

$$f(x_{n+1}) < TOL$$

- However, I recommend to STOP after BOTH

$$|x_{n+1} - x_n| < TOL \text{ and } \quad f(x_{n+1}) < TOL$$

Or

- Number of iterations (iter<=itmax)
- Initial Guess $x_0$

NOTE: most methods for non-linear equations are SENSITIVE w.r.t. the initial guess...

(In particular, Newton's method ...)

# Convergence Analysis

$Given \quad f(x),\ a,\ b,\ and \ \ \varepsilon$

How many iterations are needed such that: $|x - r| \leq \varepsilon$

where $r$ is the zero of $f(x)$ and $x$ is the

bisection estimate (i.e., $x = c_k$)?

$$\boxed{n \geq \frac{\log(b-a) - \log(\varepsilon)}{\log(2)}}$$

# Convergence Analysis – Alternative Form

$$n \geq \frac{\log(b-a) - \log(\varepsilon)}{\log(2)}$$

$$n \geq \log_2\left(\frac{\text{width of initial interval}}{\text{desired error}}\right) = \log_2\left(\frac{b-a}{\varepsilon}\right)$$

# Example

$a = 6,\ \ b = 7,\ \varepsilon = 0.0005$

How many iterations are needed such that: $\left| x - r \right| \le \varepsilon$?

$$n \ge \frac{\log(b-a) - \log(\varepsilon)}{\log(2)} = \frac{\log(1) - \log(0.0005)}{\log(2)} = 10.9658$$

$$\Rightarrow n \ge 11$$

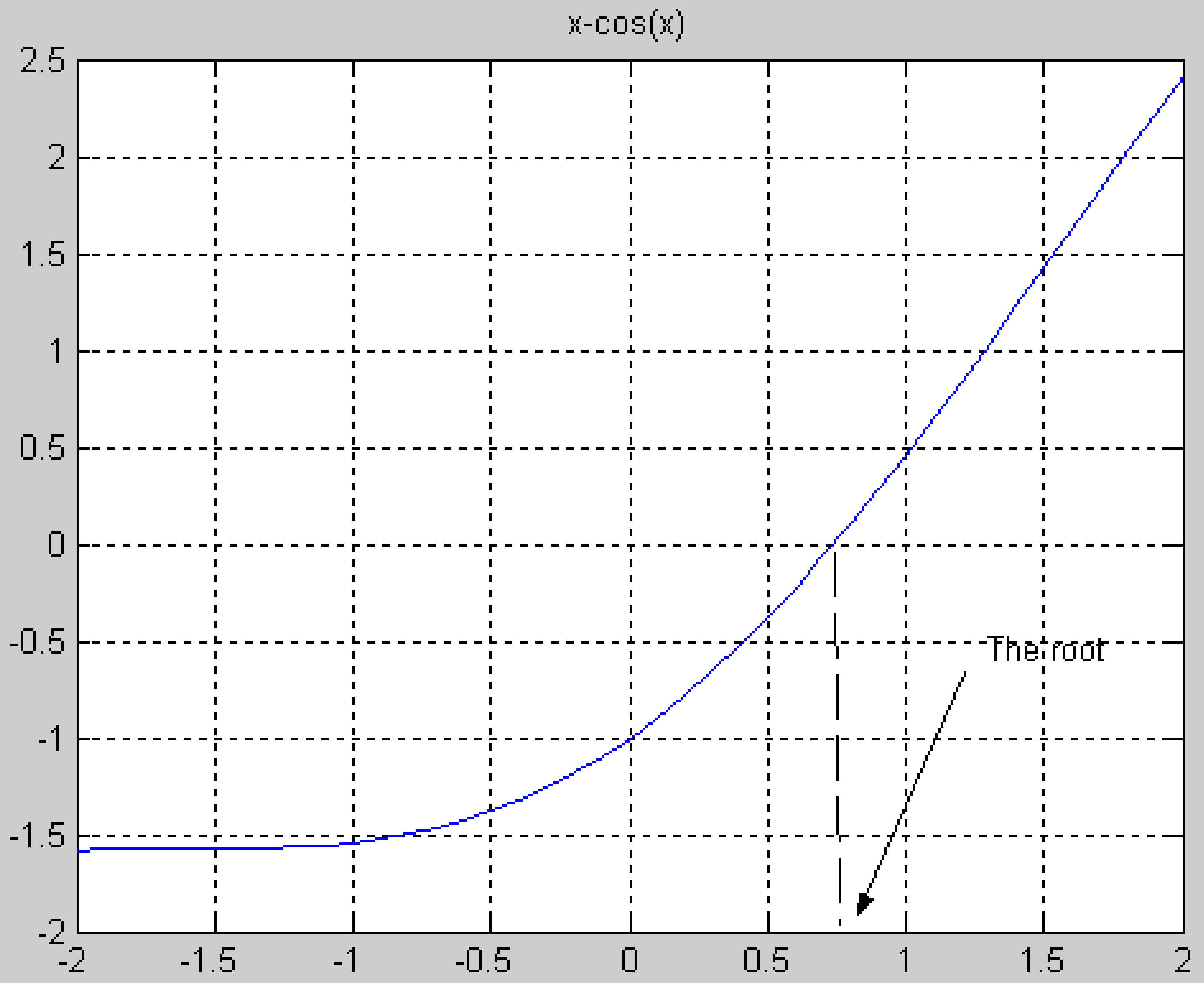
# Example

- Use Bisection method to find a root of the equation $x = \cos(x)$ with $(b-a)/2^n<0.02$

  (assume the initial interval [0.5,0.9])

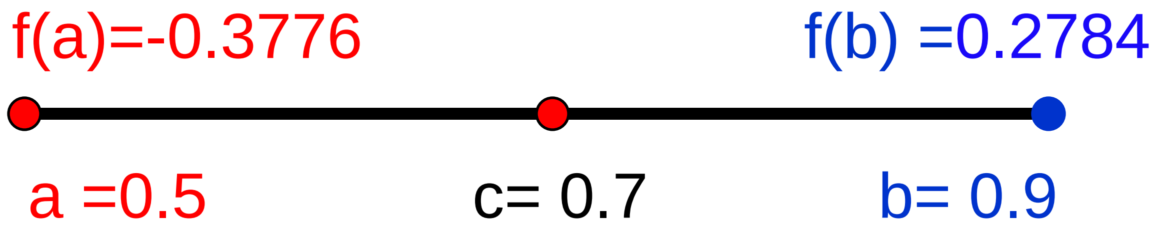
Question 1: What is $f(x)$ ?

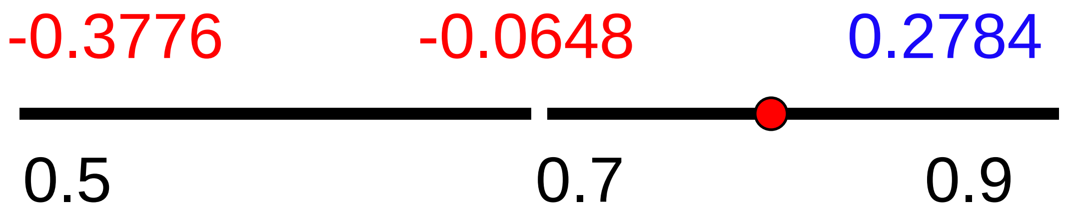
Question 2: Are the assumptions satisfied ?


x-cos(x)
2.5
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-2
-1.5
-1
-0.5
0
0.5
1
1.5
2
The root

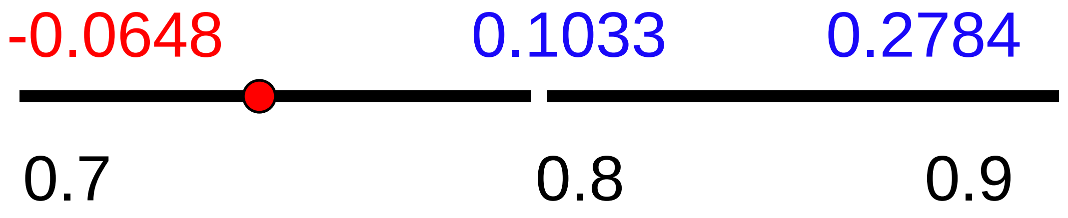
# Bisection Method

## Initial Interval

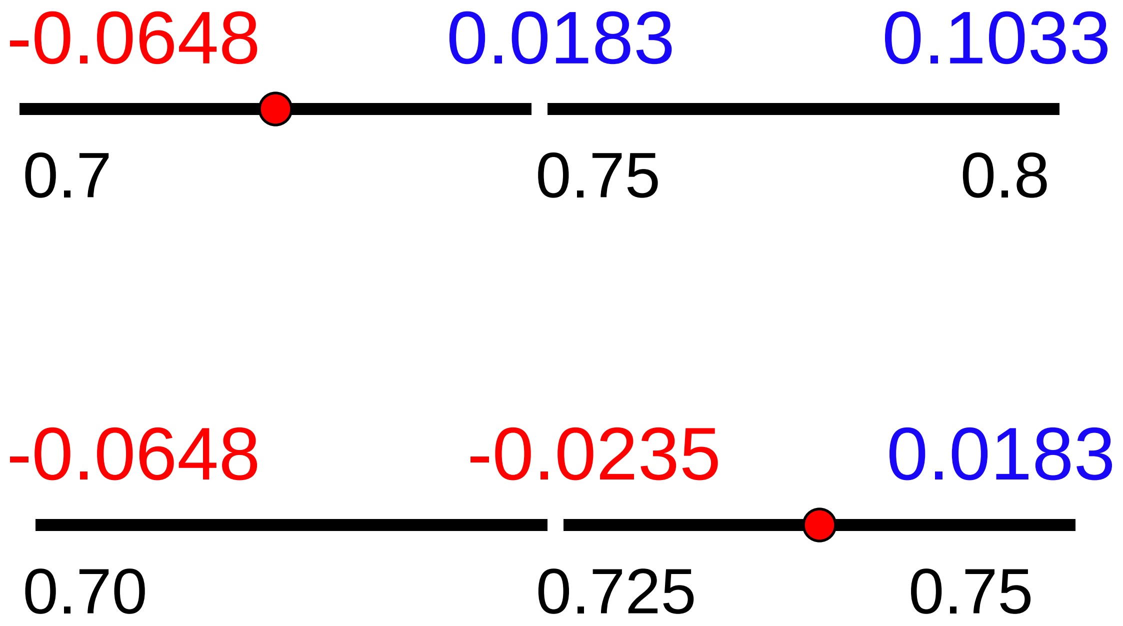





$(0.9-0.7)/2 = 0.1$




$(0.8-0.7)/2 = 0.05$




(0.75-0.7)/2=
0.025

(0.75-0.725)/2=
.0125

# Bisection Method Programming in Scilab

```
a=.5; b=.9;
u=a-cos(a);
v= b-cos(b);
  for i=1:5
      c=(a+b)/2
      fc=c-cos(c)
      if u*fc<0
        b=c ; v=fc;
      else
          a=c; u=fc;
      end
  end
```

```
c =
   0.7000
fc =
  -0.0648
c =
   0.8000
fc =
   0.1033
c =
   0.7500
fc =
   0.0183
c =
   0.7250
fc =
  -0.0235
```

# Bisection Method

- Advantage:
  - A global method: it always converge no matter how far you start from the actual root.
- Disadvantage:
  - It cannot be used to find roots when the function is tangent is the axis and does not pass through the axis.
    - For example:
  - It converges slowly compared with other methods.

$$f(x) = x^2$$

# Newton’s Method

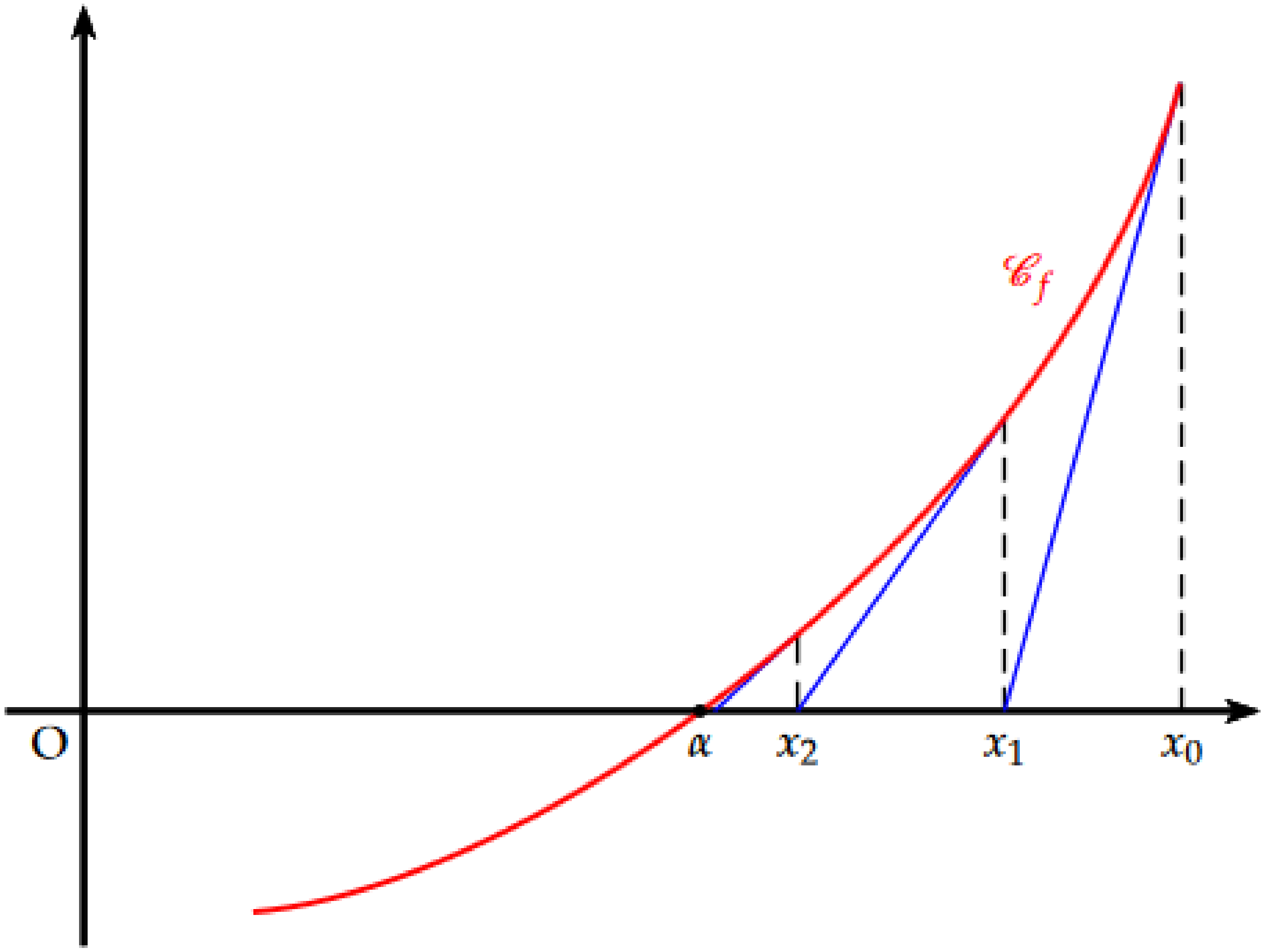
# Newton-Raphson Method
(also known as Newton's Method)

Given an initial guess of the root $x_0$ , Newton-Raphson method uses information about the function and its derivative at that point to find a better guess of the root.

Assumptions:

- $f(x)$ is continuous and first derivative is known
- An initial guess $x_0$ such that $f'(x_0) \neq 0$ is given

# Newton’s Method



# Recurrence formula

$x_{n+1}$ est l'abscisse du point d'intersection de la tangente à $\mathscr{C}_f$ en $x_n$ avec l'axe des abscisses.

L'équation de la tangente en $x_n$ est : $y = f'(x_n)(x - x_n) + f(x_n)$

Cette tangente coupe l'axe des abscisse quand $y = 0$ :

$$f'(x_n)(x - x_n) + f(x_n) = 0 \Leftrightarrow f'(x_n)(x - x_n) = -f(x_n)$$

$$x - x_n = -\frac{f(x_n)}{f'(x_n)} \Leftrightarrow x = x_n - \frac{f(x_n)}{f'(x_n)}$$

On a donc la relation de récurrence suivante : $x_{n+1} = x_n - \dfrac{f(x_n)}{f'(x_n)}$

# Newton’s Method

- Choose some initial guess $x_0$ such that $f'(x_0) \neq 0$
- Generate the sequence by $x_{n+1}=x_n+v_{n+1}$

  where $f(x_n)+ v_{n+1} f'(x_n)=0$

# Example

Use Newton's Method to find a root of

$f(x) = e^{-x} - x, \qquad f'(x) = -e^{-x} - 1$. Use the initial points $x_0 = 1$

Stop after three iterations

$Given \quad f(x),\ f'(x),\ x_0$

$Assumption \quad f'(x_0) \neq 0$

---

$for \quad i = 0 : n$

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}$$

$end$

FN

$function \quad [FN] = FN(X)$

$FN = \exp(-X) - X$

FNP

$function \quad [FNP] = FNP(X)$

$FNP = -\exp(-X) - 1$

```
// Scilab Program
X=1;
For i=1:3
X=X-FN(X)/FNP(X);
FN(X);
end
```

# Results

- X = 0.5379

  FNX =0.0461

- X =0.5670

  FNX =2.4495e-004

- X =  0.5671

  FNX =6.9278e-009

# Newton's Method

- Advantage:
  - Very fast
- Disadvantage:
  - Not a global method
    - For example: Figure 3.3 (root $x$ = 0.5) $f(x) = \frac{4}{3}e^{2-x/2}(1 + x^{-1}\log x)$
    - Another example: Figure 3.4 (root $x$ = 0.05) $f(x) = \frac{20x - 1}{19x}$
    - In these example, the initial point should be carefully chosen.
  - Newton's method will cycle indefinitely.
    - Newton's method will just hop back and forth between two values.
    - For example: Consider (root $x$ = 0)

$$f(x) = \arctan(x).$$

$$x_0 = 1.39174520027\ldots \quad x_1 = -1.39174520027\ldots \quad x_2 = 1.39174520027\ldots \boxed{= x_0}$$

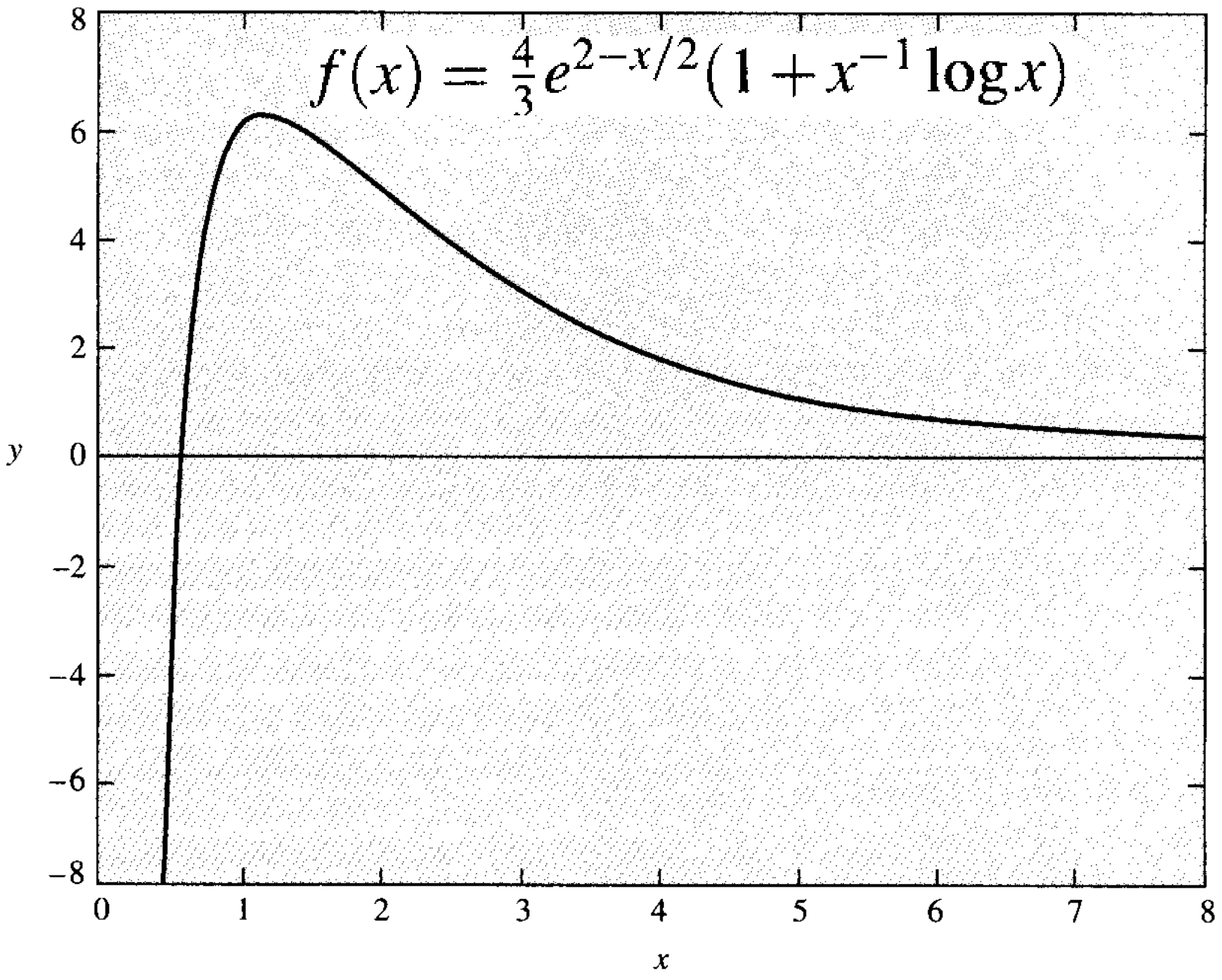



**FIGURE 3.3** A function for which Newton's method will not work well, unless $x_0$ is carefully chosen.

$$f(x) = \frac{20x - 1}{19x}$$

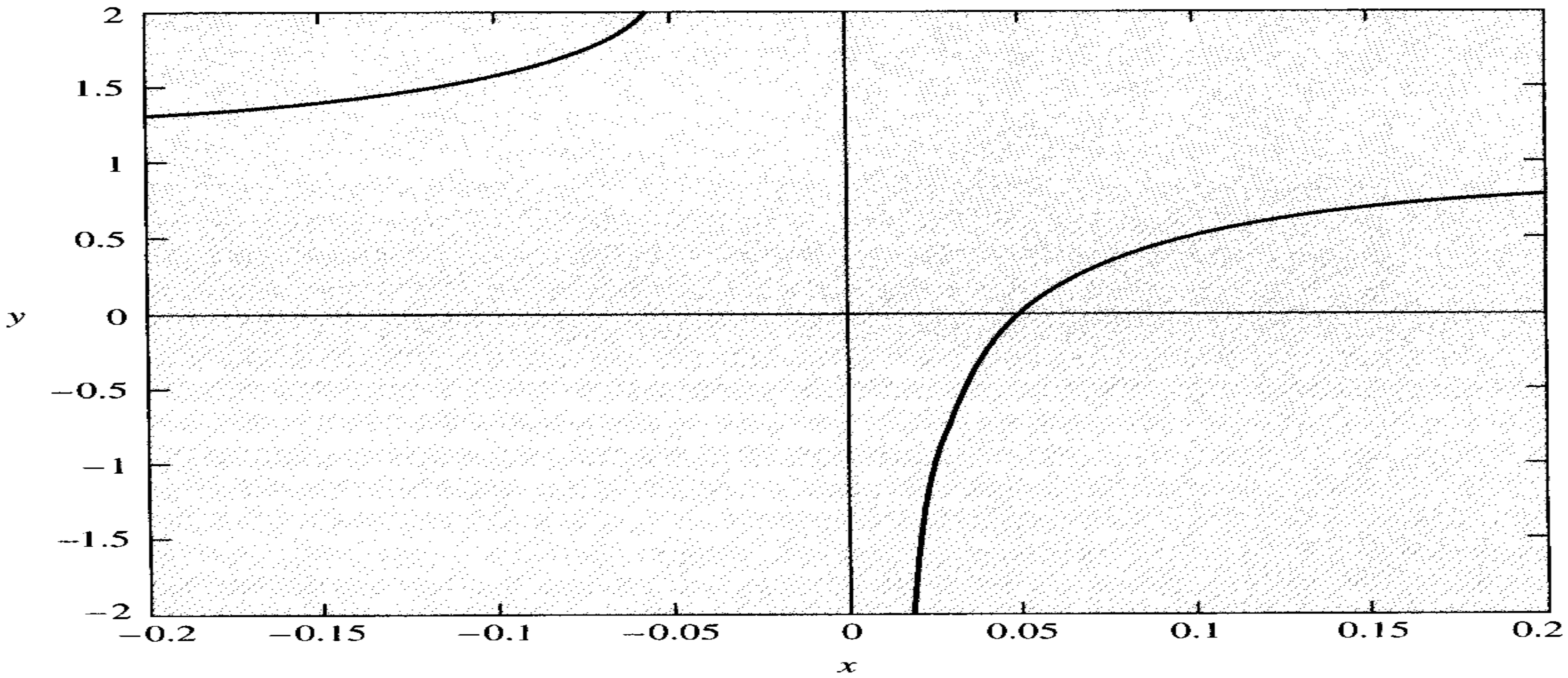



**FIGURE 3.4** A second function for which Newton's method will not work well, unless $x_0$ is very close to $\alpha$.

# How to find the initial value?

- Choose the midpoint of the interval
  - For example:

    If the interval $[\frac{1}{4}, 1]$ take $x_0 = \frac{5}{8}$,

- Using linear interpolation
  - For example:

take our nodes to be $\frac{1}{4}$ and 1 and apply the linear interpolation

$$p_1(x) = \frac{x - 1/4}{3/4}\sqrt{1} + \frac{1-x}{3/4}\sqrt{1/4} = \frac{2x+1}{3}$$

b is known

we take $x_0 = p_1(b)$; therefore $x_0 = \frac{1}{3}(2b+1)$.

# Newton’s Method for n dimensional systems

# Systems of Non-linear Equations: n-dimensional case

- Example:

$$
\begin{aligned}
x + y + z &= 3 \\
x^2 + y^2 + z^2 &= 5 \\
e^x + xy - xz &= 1
\end{aligned}
$$

*Plane intersected with a sphere, intersected with a more complex function.*

- Conservation of mass coupled with conservation of energy, coupled with solution to complex problem.

# Vector Notation

- We can rewrite this using vector notation:

$$\vec{\mathbf{f}}(\vec{\mathbf{x}}) = \vec{\mathbf{0}}$$
$$\mathbf{f} = \left( f_1, f_2, \ldots, f_n \right)$$
$$\mathbf{x} = \left( x_1, x_2, \ldots, x_n \right)$$

# Newton's Method for Non-linear Systems

- Newton's method for non-linear systems can be written as:

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \left[\mathbf{f}'\left(\mathbf{x}^{(k)}\right)\right]^{-1} \mathbf{f}\left(\mathbf{x}^{(k)}\right)$$

$$\textit{where } \mathbf{f}'\left(\mathbf{x}^{(k)}\right) \textit{ is the Jacobian matrix}$$

# The Jacobian Matrix

- The Jacobian contains all the partial derivatives of the set of functions.

$$\mathbf{J} = \begin{bmatrix} \dfrac{\partial f_1}{\partial x_1} & \dfrac{\partial f_1}{\partial x_2} & \cdots & \dfrac{\partial f_1}{\partial x_n} \\ \dfrac{\partial f_2}{\partial x_1} & \dfrac{\partial f_2}{\partial x_2} & \cdots & \dfrac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \dfrac{\partial f_n}{\partial x_1} & \dfrac{\partial f_n}{\partial x_2} & \cdots & \dfrac{\partial f_n}{\partial x_n} \end{bmatrix}$$

- Note, that these are all functions and need to be evaluated at a point to be useful.

# Newton's Method

- If the Jacobian is non-singular, such that its inverse exists, then we can apply this to Newton's method.
- We rarely want to compute the inverse, so instead we look at the problem.

$$\begin{aligned}\mathbf{x}^{(i+1)} &= \mathbf{x}^{(i)} - \left[\mathbf{f}'\left(\mathbf{x}^{(i)}\right)\right]^{-1} \mathbf{f}\left(\mathbf{x}^{(i)}\right) \\ &= \mathbf{x}^{(i)} + \mathbf{h}^{(i)}\end{aligned}$$

# Newton's Method

- Now, we have a linear system and we solve for **h**.

- Repeat until **h** goes to zero.

$$\left[\mathbf{J}\left(\mathbf{x}^{(k)}\right)\right]\mathbf{h}^{(k)} = -\mathbf{f}\left(\mathbf{x}^{(k)}\right)$$

$$\mathbf{x}^{(i+1)} = \mathbf{x}^{(i)} + \mathbf{h}^{(i)}$$

*We will look at solving linear systems later in the course.*

**Theorem 3.** *Assume that the function* $F\colon \mathbb{R}^n \to \mathbb{R}^n$ *is three times differentiable, that* $\vec{s} \in \mathbb{R}^n$ *is a solution of* $F(\vec{s}) = \vec{0}$*, and that* $JF(\vec{s})$ *is an invertible matrix. Then Newton's method converges quadratically to* $\vec{s}$ *whenever* $\vec{x}_0$ *is chosen sufficiently close to* $\vec{s}$*.*

# Initial Guess

- How do we get an initial guess for the root vector in higher-dimensions?
- In 2D, I need to find a region that contains the root.
- Steepest Decent is a more advanced topic not covered in this course. It is more stable and can be used to determine an approximate root.

# Fixed Point Iteration Method

# Fixed Point Iteration Method

- *Fixed point* of given function $g: \mathbb{R} \to \mathbb{R}$ is value $x$ such that

$$x = g(x)$$

- Many iterative methods for solving nonlinear equations use *fixed-point iteration* scheme of form

$$x_{k+1} = g(x_k)$$

  where fixed points for $g$ are solutions for $f(x) = 0$

- Also called *functional iteration*, since function $g$ is applied repeatedly to initial starting value $x_0$
- For given equation $f(x) = 0$, there may be many equivalent fixed-point problems $x = g(x)$ with different choices for $g$

# Other Examples:

$$f(x) = x^2 - x - 2 \qquad\qquad x \succ 0$$

$$g(x) = x^2 - 2$$

$$or$$

$$g(x) = \sqrt{x+2}$$

$$or$$

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

$$\vdots$$

# Example:

Consider Newton's method as applied to $f(x) = x^2 - a$:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}.$$

$$x_{n+1} = \frac{1}{2}\left(x_n + \frac{a}{x_n}\right). \tag{3.32}$$

As $n \to \infty$, we know that $\boxed{x_n \to \alpha = \sqrt{a}.}$ (In this case, convergence occurs for any $x_0 > 0$.) We can write (3.32) more abstractly as

$$x_{n+1} = g(x_n) \tag{3.33}$$

for $\boxed{g(x) = \frac{1}{2}(x + ax^{-1}).}$ Note that

root

$$f(\alpha) = 0 \iff \boxed{\alpha = g(\alpha).}$$

To verify, simply let $x_n = x_{n+1} = x$, and solve for $x$.

1. $x_{n+1} = x_n + \frac{1}{2}(x_n^2 - a)$; $g(x) = x + \frac{1}{2}(x^2 - a)$
2. $x_{n+1} = a/x_n$; $g(x) = a/x$
3. $x_{n+1} = a + x - x^2$; $g(x) = a + x - x^2$

So, for a given function $g$, a number of questions can be raised:

1. Under what conditions does a fixed point exist?
2. Under what conditions does the iteration (3.33) converge?
3. If the iteration converges, how fast does it converge?

# Theorem of FPI

*Theorem 3.5 (Fixed-Point Existence and Convergence Theory)* *Let* $g \in C([a,b])$ *with* $a \leq g(x) \leq b$ *for all* $x \in [a,b]$*; then*

1. $g$ *has at least one fixed point* $\alpha \in [a,b]$.

2. *If there exists a value* $\gamma < 1$ *such that*

$$|g(x) - g(y)| \leq \gamma |x - y| \tag{3.36}$$

*for all* $x$ *and* $y$ *in* $[a,b]$*, then*

(a) $\alpha$ *is unique.*

(b) *The iteration* $x_{n+1} = g(x_n)$ *converges to* $\alpha$ *for any initial guess* $x_0 \in [a,b]$.

(c) *We have the error estimate*

$$|\alpha - x_n| \leq \frac{\gamma^n}{1-\gamma} |x_1 - x_0|. \tag{3.37}$$

# Theorem FPI(con.)

*3. If g is continuously differentiable on [a,b] with*

$$\max_{x\in[a,b]} |g'(x)| = \underline{\gamma < 1}, \tag{3.38}$$

*then*

*(a)* $\alpha$ *is unique.*

*(b) The iteration* $x_{n+1} = g(x_n)$ *converges to* $\alpha$ *for any initial guess* $x_0 \in [a,b]$.

*(c) We have the error estimate*

$$|\alpha - x_n| \leq \frac{\gamma^n}{1-\gamma}|x_1 - x_0|.$$

*(d) The limit*

$$\lim_{n\to\infty} \frac{\alpha - x_{n+1}}{\alpha - x_n} = g'(\alpha)$$

*holds.*

*Theorem 3.6 (Local Convergence for Fixed-Point Iterations)* *Let $g$ be continuously differentiable in an open interval of a fixed point $\alpha$ with $|g'(\alpha)| < 1$; then, for all $x_0$ sufficiently close to $\alpha$, the iteration $x_{n+1} = g(x_n)$ converges,*

$$\lim_{n\to\infty} \frac{\alpha - x_{n+1}}{\alpha - x_n} = g'(\alpha),$$

*and*

$$|\alpha - x_n| \leq \frac{\gamma^n}{1-\gamma}|x_1 - x_0|,$$

*for some $\gamma < 1$.*

*Theorem 3.7* *Consider the fixed-point iteration*

$$x_{n+1} = g(x_n), \tag{3.40}$$

*where $g$ is $p$ times continuously differentiable, and $\alpha = g(\alpha)$. If*

$$g'(\alpha) = g''(\alpha) = \cdots = g^{(p-1)}(\alpha) = 0$$

*but*

$$g^{(p)}(\alpha) \neq 0,$$

*then the iteration (3.40) converges with order $p$ for $x_0$ sufficiently close to $\alpha$.*

# Multiple Roots

- So far our study of root-finding methods has assumed that the derivative of the function does not vanish at the root:

$\Rightarrow$ $f'(\alpha) \neq 0.$

- What happens if the derivative does vanish at the root?

Lemma 3.1 *If $f$ is $k$ times continuously differentiable in a neighborhood of $\alpha$, and*

$$f(\alpha) = f'(\alpha) = \cdots = f^{(k-1)}(\alpha) = 0,$$

*i.e., if the function and first $k-1$ derivatives vanish at $\alpha$, but $f^{(k)}(\alpha) \neq 0$, then we can write*

$$f(x) = (x-\alpha)^k F(x) \tag{3.58}$$

*where $F(\alpha) \neq 0$. Similarly, if we can write $f$ in the form (3.58), where $F(\alpha) \neq 0$, then it follows that the first $k-1$ derivatives vanish at $\alpha$.*

# Example:

Let $f(x) = \cos^2 x$, which has a root at $x = \frac{1}{2}\pi$. Since $f'(x) = -2\sin x \cos x$, it follows that the derivative also vanishes at $x = \frac{1}{2}\pi$, so $f$ has a double root. We can write $f$ in the form called for in the lemma by the simple device of writing

$$f(x) = \cos^2 x = \left(x - \frac{\pi}{2}\right)^2 F(x),$$

where

$$F(x) = \frac{\cos^2 x}{(x - \pi/2)^2}.$$

So long as $x \neq \pi/2$, $F(x)$ is well defined. What happens at $x = \pi/2$? We can use L'Hôpital's rule to determine that

$$\lim_{x \to \pi/2} F(x) = \lim_{x \to \pi/2} \frac{-2\sin x \cos x}{2}(x - \pi/2) = \lim_{x \to \pi/2} \frac{-2\cos^2 x + 2\sin^2 x}{2} = 1.$$

Thus we would define

$$F(x) = \begin{cases} \dfrac{\cos^2 x}{(x - \pi/2)^2} & x \neq \pi/2 \\ 1 & x = \pi/2. \end{cases}$$

# Iterative Solution

Find the root of: $f(x) = \mathbf{e^{-x} - x}$

1. Start with a guess say $x_1=1$,
2. Generate
   a) $x_2=e^{-x1} = e^{-1}= 0.368$
   b) $x_3=e^{-x2}= e^{-0.368} = 0.692$
   c) $x_4=e^{-x3}= e^{-0.692}=0.500$

$$x_{n+1} = e^{-x_n}$$

In general:

After a few more iteration we will get

$$0.567 \approx e^{-0.567}$$

| $n$ | $x_n$ |
|---|---|
| 1 | 1.000 |
| 2 | 0.368 |
| 3 | 0.692 |
| 4 | 0.500 |
| 5 | 0.606 |
| 6 | 0.545 |
| 7 | 0.579 |
| 8 | 0.560 |
| 9 | 0.571 |
| 10 | 0.564 |
| 11 | 0.568 |
| 12 | 0.566 |
| 13 | 0.567 |
| 14 | 0.567 |
| 15 | 0.567 |

# Problem

$$f(x) = 2x^2 - 4x + 1$$

- Find a root near *x=1.0* and *x=2.0*
- Solution:

  - Starting at x=1, x=0.292893 at 15th iteration
  - Starting at x=2, it will not converge
  - Why? Relate to *g'(x)=x*. for convergence *g'(x) < 1*

$$x = g(x) = \tfrac{1}{2}x^2 + \tfrac{1}{4}$$

  - Starting at x=1, x=1.707 at iteration 19
  - Starting at x=2, x=1.707 at iteration 12
  - Why? Relate to

$$g'(x) = \left(2x - \tfrac{1}{2}\right)^{-\frac{1}{2}}$$

$$x = g(x) = \sqrt{2x - \tfrac{1}{2}}$$

# Examples

If $f(x) = x^2 - x - 2$, then fixed points of each of functions

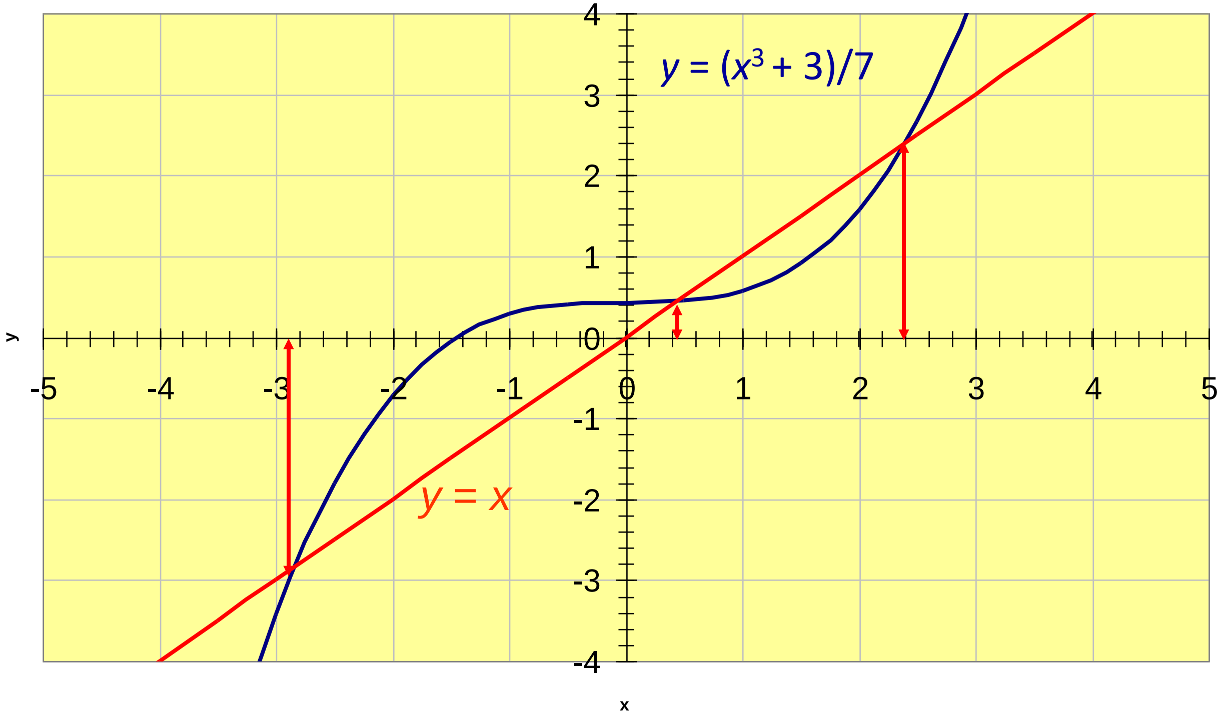
- $g(x) = x^2 - 2$
- $g(x) = \sqrt{x+2}$
- $g(x) = 1 + 2/x$
- $g(x) = \dfrac{x^2+2}{2x-1}$

are solutions to equation $f(x) = 0$

# *Fixed Point Iteration*

The equation f($x$) = 0, where f($x$) = $x^3 - 7x + 3$, may be re-arranged to give $x = (x^3 + 3)/7$.

Intersection of the graphs of $y = x$ and $y = (x^3 + 3)/7$ represent roots of the original equation $x^3 - 7x + 3 = 0$.



# Fixed Point Iteration

The rearrangement $x = (x^3 + 3)/7$ leads to the iteration

$$x_{n+1} = \frac{x_n^{\,3} + 3}{7}, \quad n = 0, 1, 2, 3, \ldots$$

To find the middle root $\alpha$, let initial approximation $x_0 = 2$.

$$x_1 = \frac{x_0^{\,3} + 3}{7} = \frac{2^3 + 3}{7} = 1.57143$$

$$x_2 = \frac{x_1^{\,3} + 3}{7} = \frac{1.57143^3 + 3}{7} = 0.98292$$

$$x_3 = \frac{x_2^{\,3} + 3}{7} = \frac{0.98292^3 + 3}{7} = 0.56423$$

$$x_4 = \frac{x_3^{\,3} + 3}{7} = \frac{0.56423^3 + 3}{7} = 0.45423$$

etc.

The iteration slowly converges to give $\alpha$ = **0.441** (to 3 s.f.)

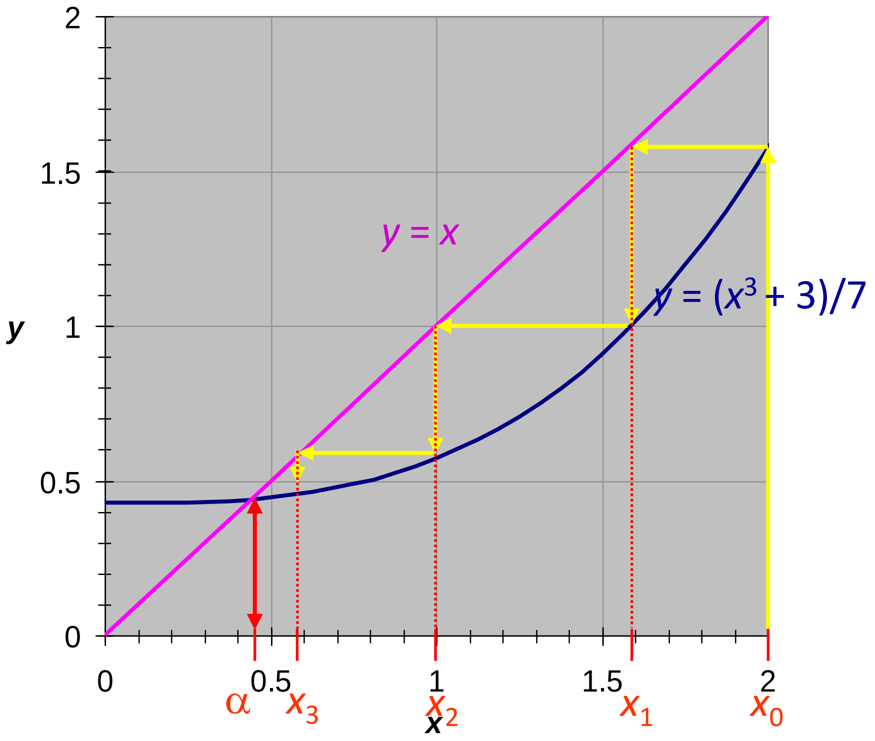
# *Fixed Point Iteration*

The rearrangement $x = (x^3 + 3)/7$ leads to the iteration

$$x_{n+1} = \frac{x_n^{\ 3} + 3}{7}, \quad n = 0, 1, 2, 3, \ldots$$

For $x_0 = 2$ the iteration will converge on the middle root $\alpha$, since $g'(\alpha) < 1$.

| $n$ | $x_n$ |
|---|---|
| 0 | 2 |
| 1 | 1.57143 |
| 2 | 0.98292 |
| 3 | 0.56423 |
| 4 | 0.45423 |
| 5 | 0.44196 |
| 6 | 0.4409 |
| 7 | 0.44082 |
| 8 | 0.44081 |




$\alpha$ = **0.441** (to 3 s.f.)

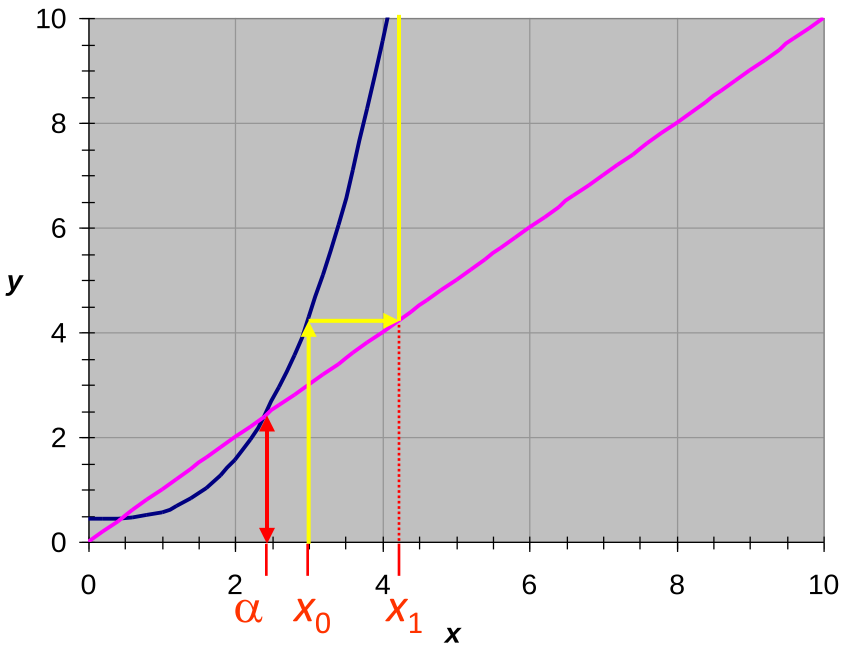
# *Fixed Point Iteration - breakdown*

The rearrangement $x = (x^3 + 3)/7$ leads to the iteration

$$x_{n+1} = \frac{x_n^{\ 3} + 3}{7}, \quad n = 0, 1, 2, 3, \ldots$$

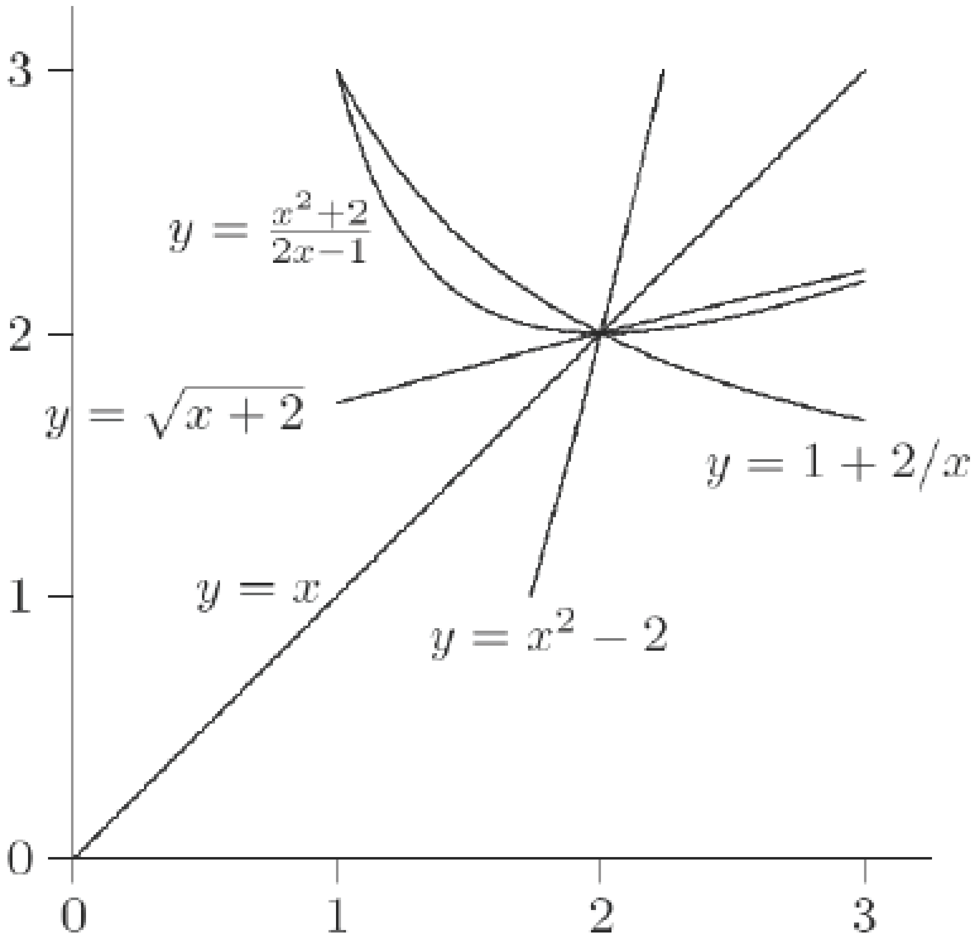
For $x_0 = 3$ the iteration will diverge from the upper root $\alpha$.

| $n$ | $x_n$ |
|---|---|
| 0 | 3 |
| 1 | 4.28571 |
| 2 | 11.6739 |
| 3 | 227.702 |
| 4 | 1686559 |
| 5 | 6.9E+17 |

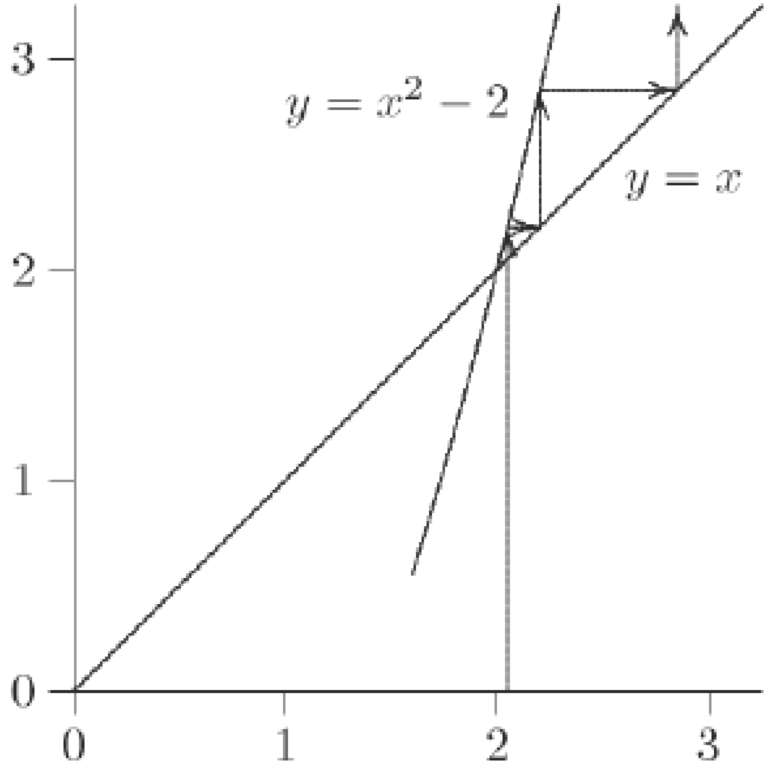


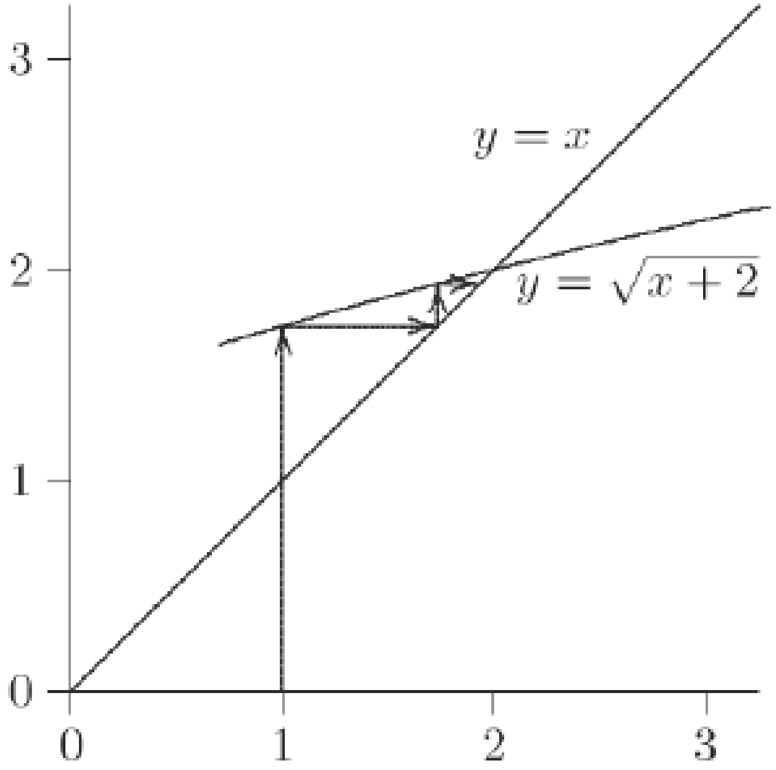

The iteration diverges because $g'(\alpha) > 1$.

# Example: fixed point problems

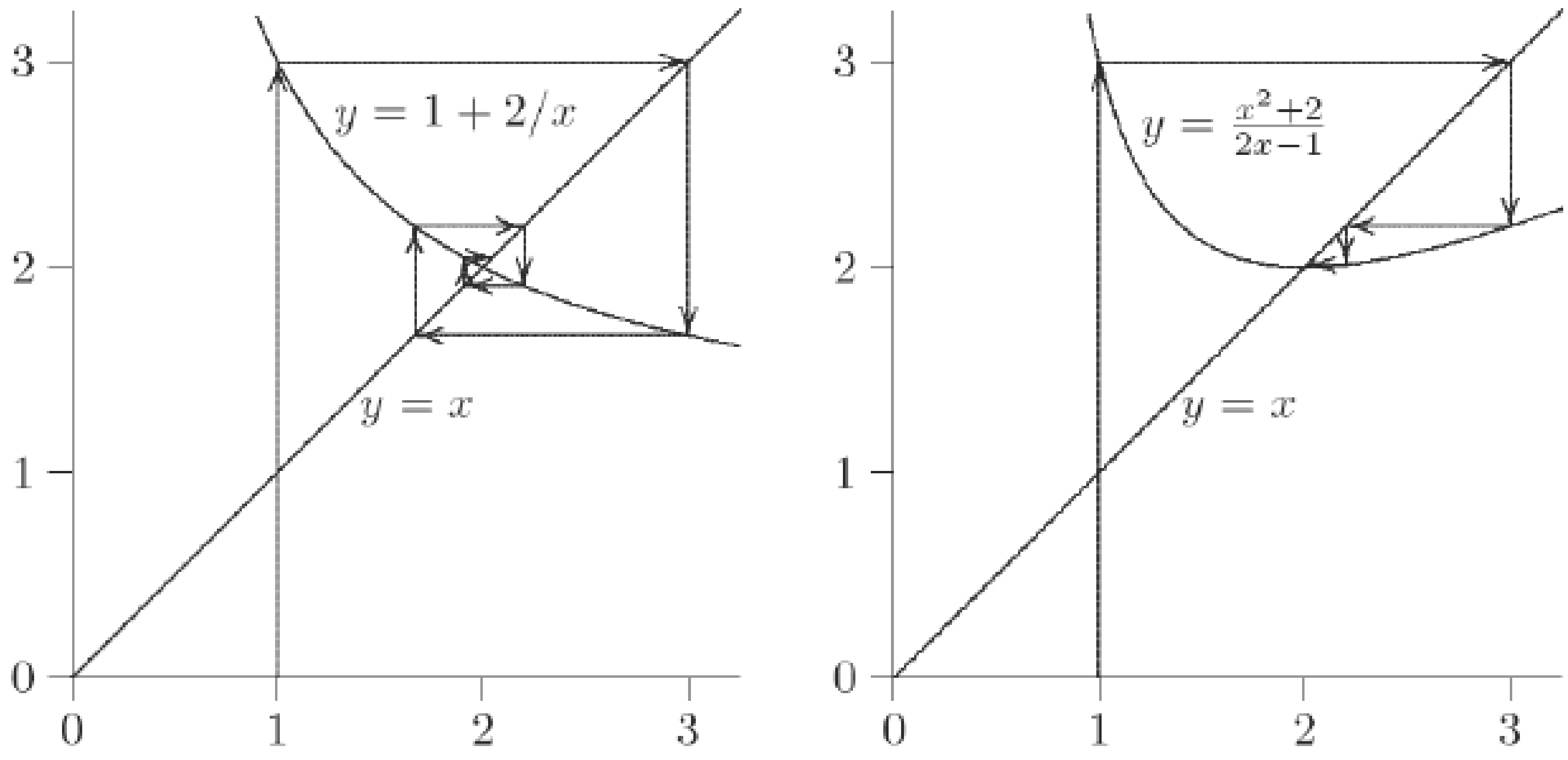


# Examples: FPI



# Example: FPI



# Convergence of FPI

- If $x^* = g(x^*)$ and $|g'(x^*)| < 1$, then there is interval containing $x^*$ such that iteration

$$x_{k+1} = g(x_k)$$

  converges to $x^*$ if started within that interval

- If $|g'(x^*)| > 1$, then iterative scheme diverges
- Asymptotic convergence rate of fixed-point iteration is usually linear, with constant $C = |g'(x^*)|$
- But if $g'(x^*) = 0$, then convergence rate is at least quadratic

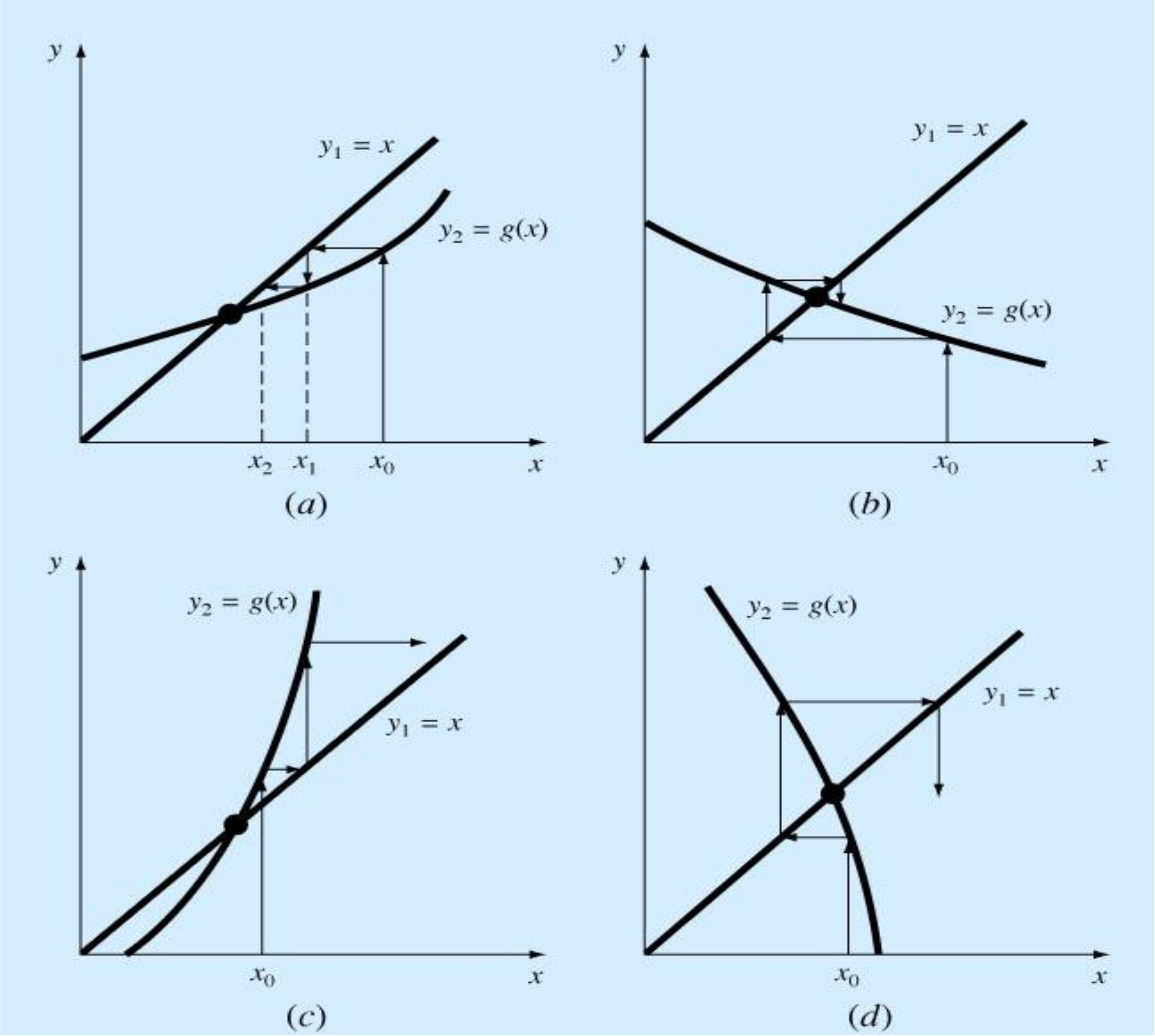
# Simple Fixed-Point Iteration Convergence

- Fixed-point iteration converges if :

$$\left|g'(x)\right| < 1 \qquad \text{(slope of the line } f(x) = x\text{)}$$

- When the method converges, the error is roughly proportional to or less than the error of the previous step, therefore it is called “linearly convergent.”

# Simple Fixed-Point Iteration-Convergence

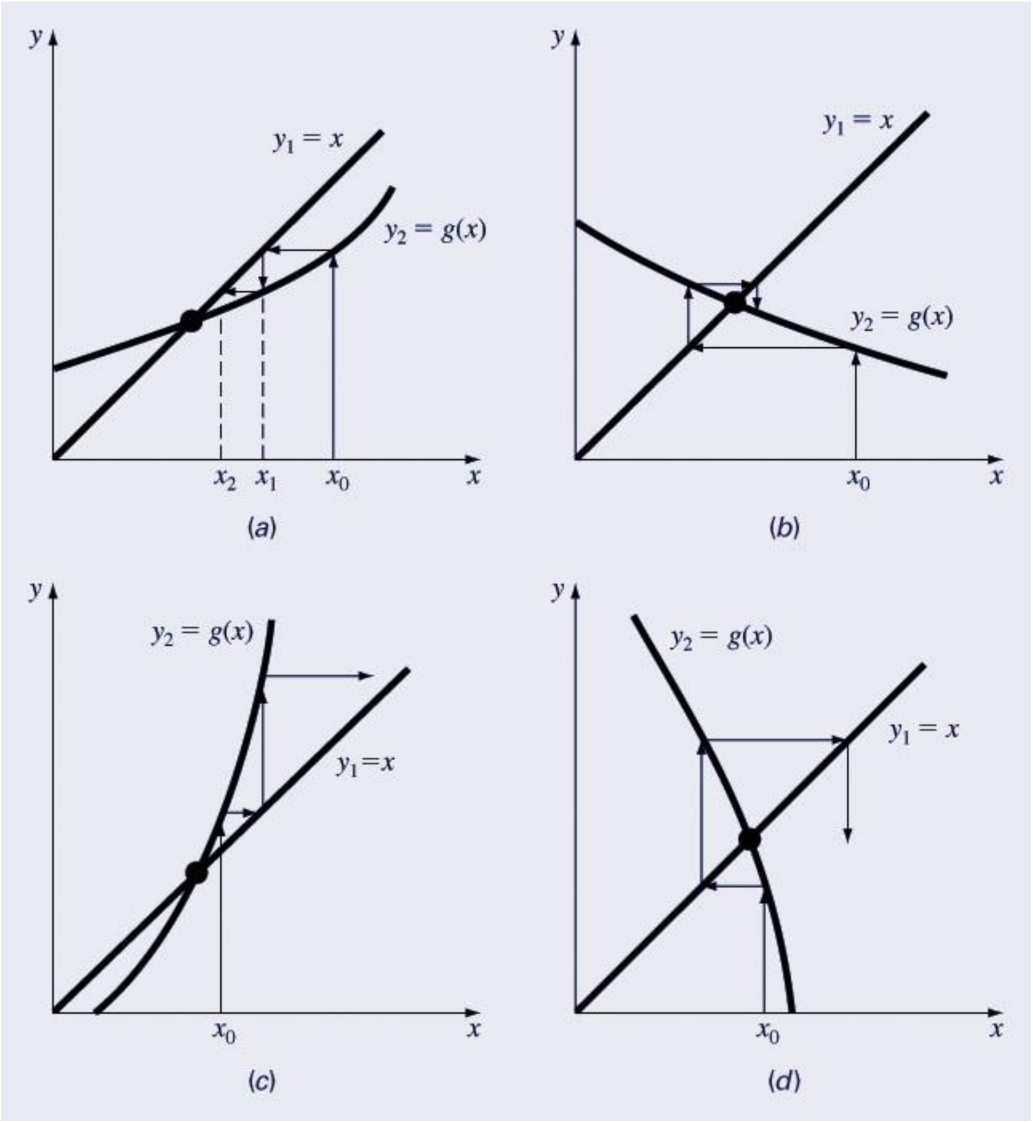


# More on Convergence

- Graphically→ the solution is at the intersection of the two curves. We identify the point on $y_2$ corresponding to the initial guess and the next guess corresponds to the value of the argument x where $y_1(x) = y_2(x)$.

- Convergence of the simple fixed-point iteration method requires that the derivative of g(x) near the root has a magnitude less than 1.

  a) Convergent, $0 \le g' < 1$
  b) Convergent, $-1 < g' \le 0$
  c) Divergent, $g' > 1$
  d) Divergent, $g' < -1$



**Definition 1.21 (Rate of Convergence of an Iterative Method).** Suppose that the sequence $\{x_k\}$ converges to $\xi$. Then the sequence $\{x_k\}$ is said to converge to $\xi$ with the *order of convergence* $\alpha$ if there exists a **positive constant** $p$ such that

$$\lim_{k\to\infty} \frac{|x_{k+1}-\xi|}{|x_k-\xi|^\alpha} = \lim_{k\to\infty} \frac{e_{k+1}}{e_k^\alpha} = p.$$

- If $\alpha = 1$, the convergence is **linear**
- If $\alpha = 2$, the convergence is **quadratic**
- If $1 < \alpha < 2$, the convergence is **superlinear**

Using the above definition, we will now show:

- The rate of convergence of the fixed-point iteration is usually linear,
- The rate of convergence of the Newton method is quadratic,
- The rate of convergence of the Secant method is superlinear [**Exercise**].

# Summary

| Method | Pros | Cons |
| --- | --- | --- |
| Bisection | - Easy, Reliable, Convergent<br>- One function evaluation per iteration<br>- A global method: it always converge no matter how far you start from the actual root.<br>- No knowledge of derivative is needed | - Slow<br>- Needs an interval [a,b] containing the root, i.e., $f(a)f(b)<0$.<br>- It cannot be used to find roots when the function is tangent to the axis and does not pass through the axis ($f(x)=x^2$) |
| Newton | - Very Fast (if near the root)<br>- Two function evaluations per iteration | - May diverge<br>- Not a global method<br>- Needs derivative and an initial guess $x_0$ such that $f'(x_0)$ is nonzero |
| Fixed Iteration method | - Fast (depends on the choice of g)<br>- One function evaluation per iteration<br>- Convergent when $g'<1$<br>-No knowledge of derivative is needed | - Divergent when $g'>1$<br>-Needs an initial guess of $x_0$ |

# Summary

| Method | Pros | Cons |
| --- | --- | --- |
| Bisection | - Easy, Reliable, Convergent<br>- One function evaluation per iteration<br>- No knowledge of derivative is needed | - Slow<br>- Needs an interval [a,b] containing the root, i.e., $f(a)f(b)<0$ |
| Newton | - Fast (if near the root)<br>- Two function evaluations per iteration | - May diverge<br>- Needs derivative and an initial guess $x_0$ such that $f'(x_0)$ is nonzero |
| Fixed Iteration method | - Fast (depends on the choice of g)<br>- One function evaluation per iteration<br>- Convergent when $g'<1$<br>-No knowledge of derivative is needed | - Divergent when $g'>1$<br>-Needs an initial guess of $x_0$ |
| Secant | - Fast (slower than Newton)<br>- One function evaluation per iteration<br>- No knowledge of derivative is needed | - May diverge<br>- Needs two initial points guess $x_0$, $x_1$ such that $f(x_0)- f(x_1)$ is nonzero |

There also

# Scilab TP: Resolution of $f(x)=0$

# Algorithm of Bisection Method

**Data**: f, a, b, eps

**Output**: alpha **(**approximation of the root of f on [a,b]**)**

**Step 1**: c= (a+b)/2 (generation of the sequence  (c_n))

**Step 2**: If |b-c|<eps then alpha:=c Stop.

**Step 3**: If f(a).f(b)<=0 then a:=c

else If b:=c

**Step 4**: go to step 1

# TP1: Exercice 1

Programmer la méthode de Bissection pour f(x)=x-cos(x) sur [a,b]=[0.5,0.9]

```
a=.5; b=.9;
u=a-cos(a);
v= b-cos(b);
   for i=1:5
        c=(a+b)/2
        fc=c-cos(c)
        if u*fc<0
          b=c ; v=fc;
        else
            a=c; u=fc;
        end
   end
```

```
c =
   0.7000
fc =
  -0.0648
c =
   0.8000
fc =
   0.1033
c =
   0.7500
fc =
   0.0183
c =
   0.7250
fc =
  -0.0235
```

# Exercice 2: Resolve $f(x)=0$ for

- $f(x)=x^2-2$, $[a,b]=[1,2]$, $TOL= 10^{-5}$
- $f(x)=\exp(x)-4x$, $[a,b]=[1,2.5]$, $TOL=10^{-8}$
- $f(x)=(x-1)^{20}$, $[a,b]=[1,2]$, $TOL= 10^{-6}$
- $f(x)=x^3+4x^2-10$, $[a,b]=[1,2]$, $TOL= 10^{-5}$

# Algorithm of Newton Method

**Data**: f, f’, x0,eps,iter, itmax

**Output**: x1 (approximation of the root of f on [a,b])

1. iter=1
2. fpm:=f’(x0)
3. fpm=0, iter=2 et sortir
4. x1=x0-f(x0)/fpm
5. If |x1-x0|<eps then iter=0 root:=x1 stop.
6. If iter=itmax the method is divergent, iter=1
7. Iter=iter+1; x0=x1 go to step 2.

# TP2:

1. Montrer que l’équation $f(x)=x^6-x-1$ admet une racine alpha dans $]1,2[$.
2. Expliciter l’itération de Newton-Raphston.
3. Ecrire un programme permettant d’approcher la racine alpha par la méthode de Newton.
4. Pour $x_0=2$, donner les résultats sur un tableau avec les colonnes suivantes:

   $x_n$; $f(x_n)$;alpha- $x_n$; $x_{n+1}$- $x_n$.

Comparer avec la méthode de la bissection et interpréter les résultats obtenus.

# Algorithm of Fixed Point Iteration Method

**Data**: g, x0,eps,iter, itmax

**Output**: alpha (approximation of the FPT of g on [a,b])

1. iter=1; err=1+eps
2. while (iter<=itmax and err>eps) do
   a. xk=g(x0)
   b. err=|xk-x0|
   c. iter=iter+1
   d. x0=xk
3. If (iter>itmax) then the method is divergent
   else if alpha=xk.

# TP3:

On considère les deux fonctions $f(x)=x-x^3$ et $g(x)=x+x^3$

1. Déterminer les points fixes de f et g dans l'intervalle [-1,1]
2. Etudier la convergence de l'itération du point fixe pour f puis pour g pour un point initial $x_0$ dans [-1,1].
3. Ecrire un programme réalisant l'itération du point fixe et illustrer les résultats obtenus dans la question précédente.