Numerical Algorithms for Machine Learning

Session 2

Numerical solutions of Equations in one variable

Dr. Ahmed Abdelreheem

Content

Objective

Learn several numerical algorithms for solving Root-Finding Problem

The Root-Finding Problem

- **Root-finding problem**

$$f(-4) = 16 - 3 * 4 - 4 = 0$$

$$f(x) = X^2 + 3x - 4$$

$$f(1) = 1 + 3 * 1 - 4 = 0$$

A Zero of function $f(x)$

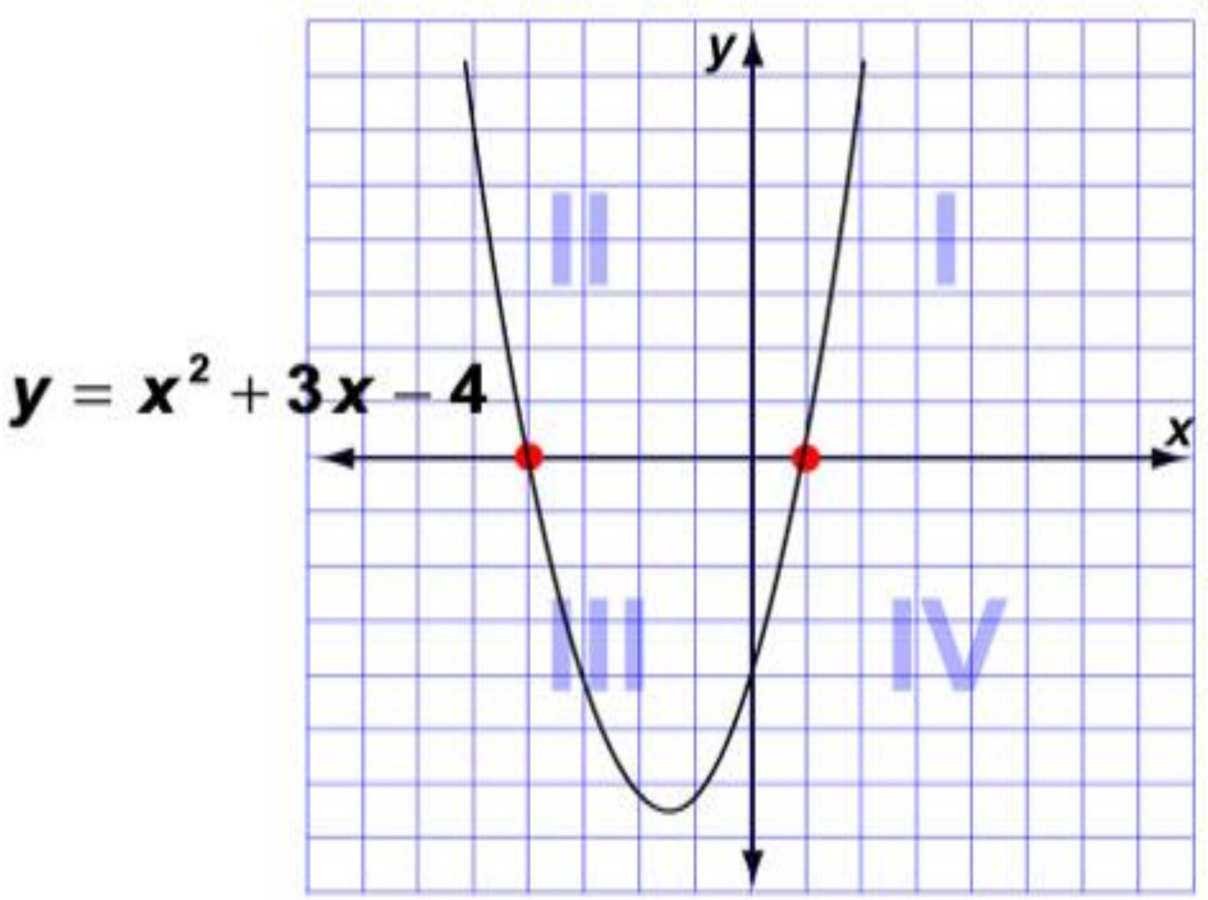
$$(x + 4)(x - 1) = 0$$

$$\Rightarrow (x + 4) = 0$$

$$\&$$

$$\Rightarrow (x - 1) = 0$$

$$x = -4 \ \& \ x = 1$$



The Root-Finding Problem

Numerical Methods for Root Finding Problem

$$f(x) = X^4 + 5x^3 + 1$$

$$f(x) = \left(\frac{x}{2}\right)^2 - \sin(x)$$

$$f(x) = x^2 - \sin(x) - 0.5$$

Intermediate Mean-Value Theorem

Intermediate Mean-Value Theorem

Let $f(x)$ be a continuous function defined on $[a, b]$, such that

- Suppose f is continuous on the interval (a_0, b_0) and

f($f(a_0) \cdot f(b_0) < 0$

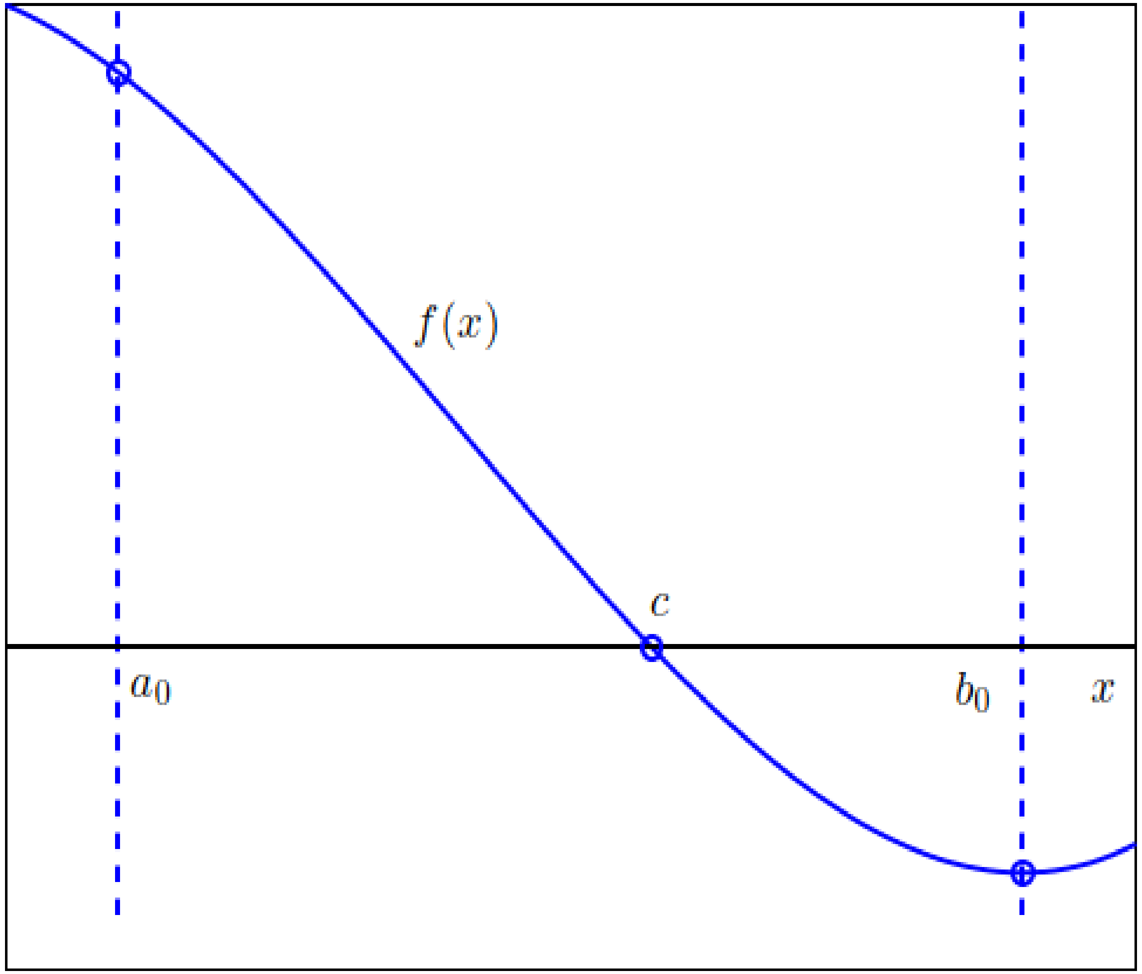
  - This means the function changes sign at least once in the interval

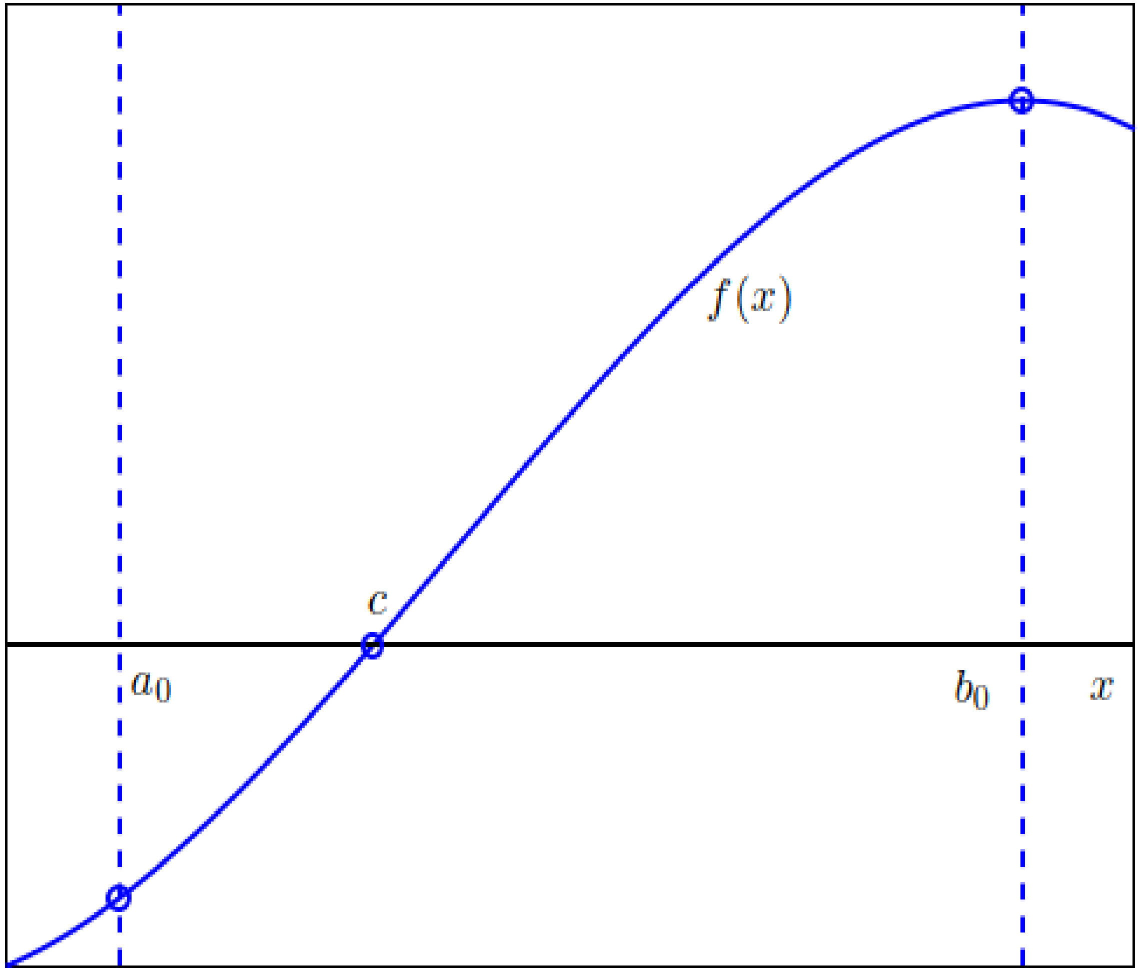
T]

- The **Intermediate Value Theorem** guarantees the existence of $c \in (a_0, b_0)$ such that $f(c) = 0$ (could be more than one)

Intermediate Mean-Value Theorem

$$\boxed{f(a_0) \cdot f(b_0) < 0}$$





Numerical Methods for Root Finding Problem

- **Bisection Method**
- **Newton’s method**
- **Fixed Point Iteration**

Numerical Methods for Root Finding Problem

- **Bisection Method**
- **Newton's method**
- **Fixed Point Iteration**

Bisection Method (Binary Search)

Basic-Idea: Let $f(x)$ be a continuous function defined on $[a_0, b_0]$

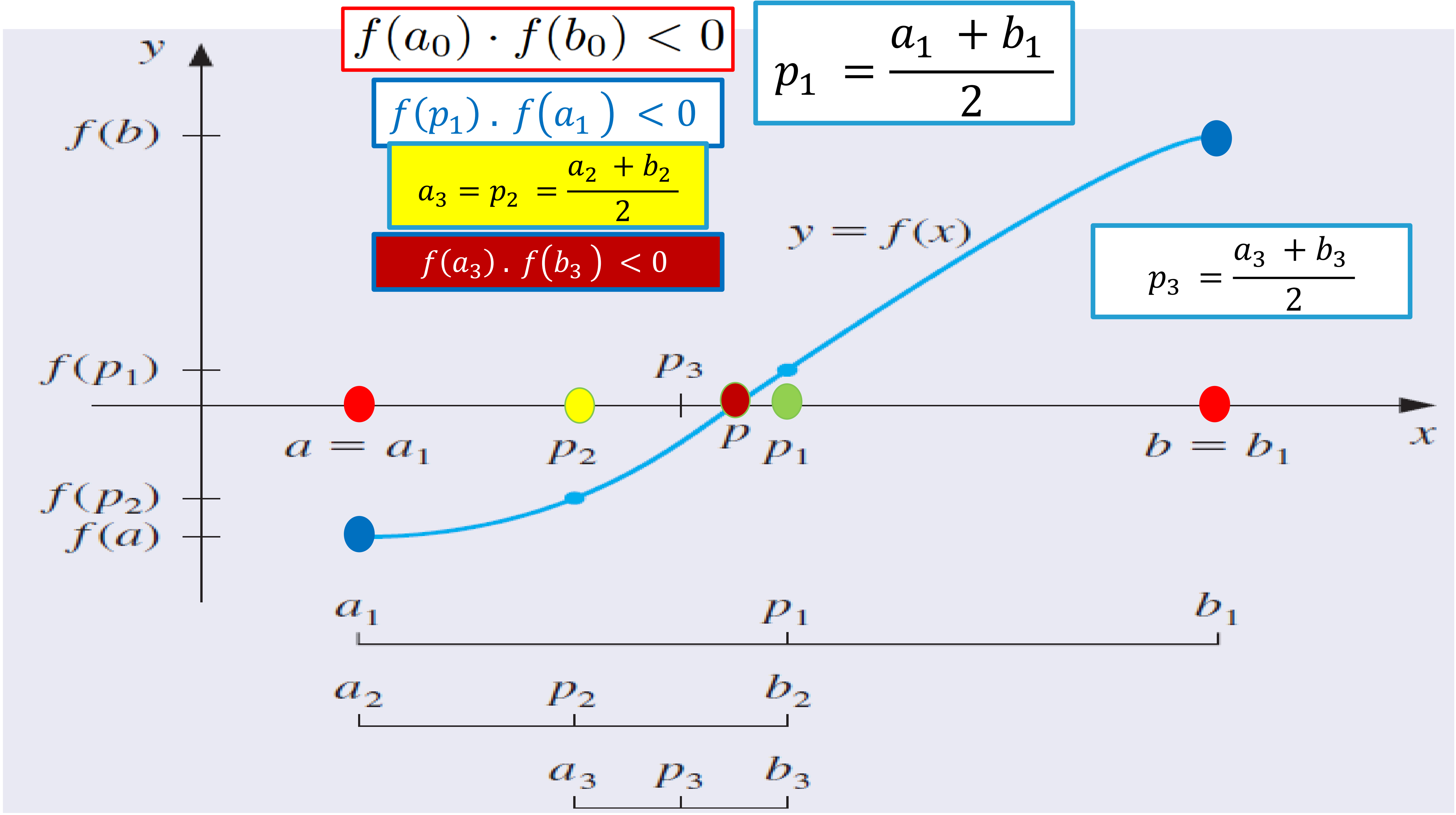
$$\boxed{f(a_0) \cdot f(b_0) < 0}$$

let $c_0 = \frac{a_0+b_0}{2}$ be the middle point of $[a_0, b_0]$:

If $(f(c_0) = 0)$: c_0 is the root, then we are done

Else:

1. If $f(c_0) . f(a_0) < 0$: the root in intervals $[a_0, c_0]$
2. If $f(c_0) . f(b_0) < 0$: the root in intervals $[c, b_0]$



$f(a_0) \cdot f(b_0) < 0$
$p_1 = \frac{a_1 + b_1}{2}$
$f(p_1) . f(a_1) < 0$
$a_3 = p_2 = \frac{a_2 + b_2}{2}$
$f(a_3) . f(b_3) < 0$
$p_3 = \frac{a_3 + b_3}{2}$
y
x
$y = f(x)$
$f(b)$
$f(p_1)$
$f(p_2)$
$f(a)$
p_3
p
$a = a_1$
p_2
p_1
$b = b_1$
a_1 p_1 b_1
a_2 p_2 b_2
a_3 p_3 b_3

Bisection Method (Binary Search)

Algorithm 1.1 The Bisection Method for Rooting-Finding

Inputs: (i) $f(x)$ - The given function

(ii) a_0, b_0 - The two numbers such that $f(a_0)f(b_0) < 0$.

Output: An approximation of the root of $f(x) = 0$ in $[a_0, b_0]$.
For $k = 0, 1, 2, \cdots,$ do until satisfied:

- Compute $c_k = \dfrac{a_k + b_k}{2}$.
- Test if c_k is the desired root, if so, stop.
- If c_k is not the desired root, test if $f(c_k)f(a_k) < 0$. If so, set $b_{k+1} = c_k$ and $a_{k+1} = a_k$. Otherwise, set $a_{k+1} = c_k$, $b_{k+1} = b_k$.

Example 1.1 $f(x) = x^3 - 6x^2 + 11x - 6.$

Let $a_0 = 2.5,\ b_0 = 4.$ *Then* $f(a_0)f(b_0) < 0.$ *Then there is a root in* $[2.5, 4]$

Iteration 1. $k = 0:$

$$c_0 = \frac{a_0 + b_0}{2} = \frac{4 + 2 \cdot 5}{2} = \frac{6 \cdot 5}{2} = 3 \cdot 25.$$

Since $f(c_0)f(a_0) = f(3 \cdot 25)f(2 \cdot 5) < 0$, set $b_1 = c_0,\ a_1 = a_0$

Iteration 2. $k = 1:$

$$c_1 = \frac{3 \cdot 25 + 2 \cdot 5}{2} = 2 \cdot 8750.$$

Since $f(c_1)f(a_1) > 0$, set $a_2 = 2 \cdot 875,\ b_2 = b_1$

Iteration 3. $k = 2:$

$$c_2 = \frac{a_2 + b_2}{2} = \frac{2 \cdot 875 + 3 \cdot 250}{2} = 3 \cdot 0625$$

Iteration 4. $k = 3$

$$c_3 = \frac{a_3 + b_3}{2} = \frac{2 \cdot 875 + 3 \cdot 0625}{2} = 2.9688.$$

It is clear that the iterations are converging towards the root $x = 3$

$$f(x) = \left(\frac{x}{2}\right)^2 - \sin(x) = 0$$

with $(a_0, b_0) = (1.5,\ 2.0)$, and $(f(a_0), f(b_0)) = (-0.4350, 0.0907)$ gives:

k	a_k	b_k	m_k	$f(m_k)$
0	1.5000	2.0000	1.7500	−0.2184
1	1.7500	2.0000	1.8750	−0.0752
2	1.8750	2.0000	1.9375	0.0050
3	1.8750	1.9375	1.9062	−0.0358
4	1.9062	1.9375	1.9219	−0.0156
5	1.9219	1.9375	1.9297	−0.0054
6	1.9297	1.9375	1.9336	−0.0002
7	1.9336	1.9375	1.9355	0.0024
8	1.9336	1.9355	1.9346	0.0011
9	1.9336	1.9346	1.9341	0.0004

Bisection Method (Binary Search)

Stopping Criteria: Since this is an iterative method, we must determine some stopping criteria that will allow the iteration to stop. Here are some commonly used stopping criteria

Let ϵ be the tolerance; that is, we would like to obtain the root with an error of at most of ϵ

When do we stop?

We can (1) keep going until successive iterates are close: $|m_k - m_{k-1}| < \epsilon$

or (2) close in relative terms

$$\frac{|m_k - m_{k-1}|}{|m_k|} < \epsilon$$

or (3) the function value is small enough

$$|f(m_k)| < \epsilon$$

Rate of Convergence

- Theorem: The number of iterations N needed in the Bisection Method to obtain an accuracy of ϵ is given by

$$N \geq \frac{[\log_{10}(b_0 - a_0) - \log_{10}(\epsilon)]}{\log_{10} 2}$$

Example *Suppose we would like to determine a priori the minimum number of iterations needed in the Bisection Algorithm, given* $a_0 = 2.5, b_0 = 4,$ *and* $\epsilon = 10^{-3}$.

$$N \geq \frac{\log_{10}(1.5) - \log_{10}(10^{-3})}{\log_{10} 2} = \frac{\log_{10}(1.5) + 3}{\log_{10}(2)} = 10.5507$$

Thus, a minimum of 11 iterations will be needed to obtain the desired accuracy using the Bisection Method.
Remarks: Since the number of iterations N needed to achieve a certain accuracy depends upon the initial length of the interval containing the root, it is desirable to choose the initial interval $[a_0 , b_0]$ as small as possible.

Example

Use the Bisection Method to approximate the real root of $x^4+5x^3+1=0$ *in the interval* $[-1,0]$, *with an error less than* 0.01.

Solution The given function is continuous, and $f(-1)\cdot f(0) = -3\cdot 1 < 0$, so the Bisection Method applies.

n	h_n	m_n	$f(m_n)$
1	0.5	-0.5	0.4375
2	0.25	-0.75	-0.79296875
3	0.125	-0.625	-0.06811523440
4	0.0625	-0.5625	0.2102203369
5	0.03125	-0.59375	0.07768344877
6	0.015625	-0.609375	0.006471693480
7	0.0078125	-0.6171875	-0.03039621937

Since $h_7 < 0.01$, we stop the iteration, and conclude that the root of the equation in the interval $[-1,0]$ is approximately -0.6171875.

Numerical Methods for Root Finding Problem

- Bisection Method

- **Newton's method**

- Fixed Point Iteration

Newton's method

Newton's Method for Root Finding: **Newton's (or the Newton-Raphson) method is one of the most powerful and well-known numerical methods for solving a root-finding problem**

we are looking for x^* so that $f(x^*) = 0$

Newton's Method for root finding is based on the approximation $x^* \approx x - \frac{f(x)}{f'(x)}$

which is valid when x is close to x^*

Strategy: Newton's Method

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

Newton's method

Algorithm : ***The Newton-Raphson Method***

Inputs: $f(x)$ - The given function
x_0 - The initial approximation
ϵ - The error tolerance
N - The maximum number of iterations

Output: An approximation to the root $x = \xi$ or a message of failure.

Assumption: $x = \xi$ is a simple root of $f(x)$.

For $k = 0, 1, \cdots$ do until convergence or failure.

- Compute $f(x_k), f'(x_k)$.
- Compute $x_{k+1} = x_k - \dfrac{f(x_k)}{f'(x_k)}$.
- Test for convergence or failure:

$$\left.\begin{array}{ll} \text{If} & |f(x_k)| < \epsilon \\ \text{or} & \dfrac{|x_{k+1} - x_k|}{|x_k|} < \epsilon \end{array}\right] \text{stopping criteria.}$$

or $k > N$, Stop.

End

- If none of these criteria has been satisfied within a predetermined, say N iterations, then the method has failed after the prescribed number of iterations. In that case, one could try the method again with a different x_0.

- A good choice of x_0 can sometimes be obtained by drawing the graph of $f(x)$, if possible. However, there does not seem to exist a clear-cut guideline of how to choose a right starting point x_0 that guarantees the convergence of the Newton-Raphson Method to a desired root

Use Newton's method to find the roots of $f(x) = x^2 - 2$, $x_0 = 2$, $N = 4$

$f'(x) = 2x$

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)} = 2 - \frac{2}{4} = \quad 1.5$$

$$x_2 = x_1 - \frac{f(x_1)}{f'(x_1)} = \frac{3}{2} - \frac{\frac{1}{4}}{3} = \quad 1.41666667$$

$$x_3 = x_2 - \frac{f(x_2)}{f'(x_2)} = \frac{17}{12} - \frac{\frac{1}{144}}{\frac{17}{6}} = \frac{577}{408} = 1.41421569$$

$$x_4 = x_2 - \frac{f(x_2)}{f'(x_2)} = \frac{665{,}857}{470{,}832} = \quad 1.414213562375$$

Example: Newton's Method

Consider the function

$$f(x) = \cos x - x = 0$$

Approximate a root of f using Newton's Method

- To apply Newton's method to this problem we need

$$f'(x) = -\sin x - 1$$

- Starting again with $p_0 = \pi/4$,

$$p_n = p_{n-1} - \frac{f(p_{n-1})}{f(p'_{n-1})} = p_{n-1} - \frac{\cos p_{n-1} - p_{n-1}}{-\sin p_{n-1} - 1}$$

Newton's Method for $f(x) = \cos(x) - x$, $x_0 = \frac{\pi}{4}$

n	p_{n-1}	$f(p_{n-1})$	$f'(p_{n-1})$	p_n	$\lvert p_n - p_{n-1} \rvert$
1	0.78539816	-0.078291	-1.707107	0.73953613	0.04586203
2	0.73953613	-0.000755	-1.673945	0.73908518	0.00045096
3	0.73908518	-0.000000	-1.673612	0.73908513	0.00000004
4	0.73908513	-0.000000	-1.673612	0.73908513	0.00000000

Some Familiar Computations Using the Newton-Raphson Method

Computing the Square Root of a Positive Number A: Compute $\sqrt{A}$, where $A > 0$.

Computing $\sqrt{A}$ is equivalent to solving $x^2 - A = 0$. The number $\sqrt{A}$, thus, may be computed by applying the Newton-Raphson Method to $f(x) = x^2 - A$.

Since $f'(x) = 2x$, we have the following Newton iterations to generate $\{x_k\}$:

Newton-Raphson Iterations for Computing $\sqrt{A}$

Input: A - A positive number
Output: An approximation to $\sqrt{A}$.
Step 1. Guess an initial approximation x_0 to $\sqrt{A}$.
Step 2. Compute the successive approximations $\{x_k\}$ as follows:

For $k = 0, 1, 2, \dots$, do until convergence

$$x_{k+1} = x_k - \frac{x_k^2 - A}{2x_k} = \frac{x_k^2 + A}{2x_k}$$

End

Example 1.5 *Let* $A = 2,\ x_0 = 1.5$

Iteration 1. $x_1 = \dfrac{x_0^2 + A}{2x_0} = \dfrac{(1.5)^2 + 2}{3} = 1.4167$

Iteration 2. $x_2 = \dfrac{x_1^2 + A}{2x_1} = 1.4142$

Iteration 3. $x_3 = \dfrac{x_2^2 + A}{2x_2} = 1.4142$

Some Familiar Computations Using the Newton-Raphson Method

Computing the nth Root

It is easy to see that the above Newton-Raphson Method to compute $\sqrt{A}$ can be easily generalized to compute $\sqrt[n]{A}$. In this case $f(x) = x^n - A, \quad f'(x) = nx^{n-1}$.

Thus Newton's Iterations in this case are:

$$\boxed{x_{k+1} = x_k - \frac{x_k^n - A}{nx_k^{n-1}} = \frac{(n-1)x_k^n + A}{nx_k^{n-1}}.}$$

Newton-Raphson Method

Taylor Series

$$f(x) = \sum_{n=0}^{\infty} \frac{f^{(n)}(a)}{n!}(x-a)^n$$
$$= f(a) + f'(a)(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \frac{f'''(a)}{3!}(x-a)^3 + \cdots$$

Consider the first Taylor polynomial for $f(x)$ expanded about p_0 and evaluated at $x = p$.

$$f(p) = f(p_0) + (p - p_0)f'(p_0) + \frac{(p-p_0)^2}{2} f''(\xi(p)),$$

where $\xi(p)$ lies between p and p_0.

Since $f(p) = 0$, this equation gives

$$0 = f(p_0) + (p - p_0)f'(p_0) + \frac{(p-p_0)^2}{2} f''(\xi(p)).$$

Newton-Raphson Method

$$0 = f(p_0) + (p - p_0)f'(p_0) + \frac{(p - p_0)^2}{2} f''(\xi(p)).$$

- Newton's method is derived by assuming that since $|p - p_0|$ is small, the term involving $(p - p_0)^2$ is much smaller, so

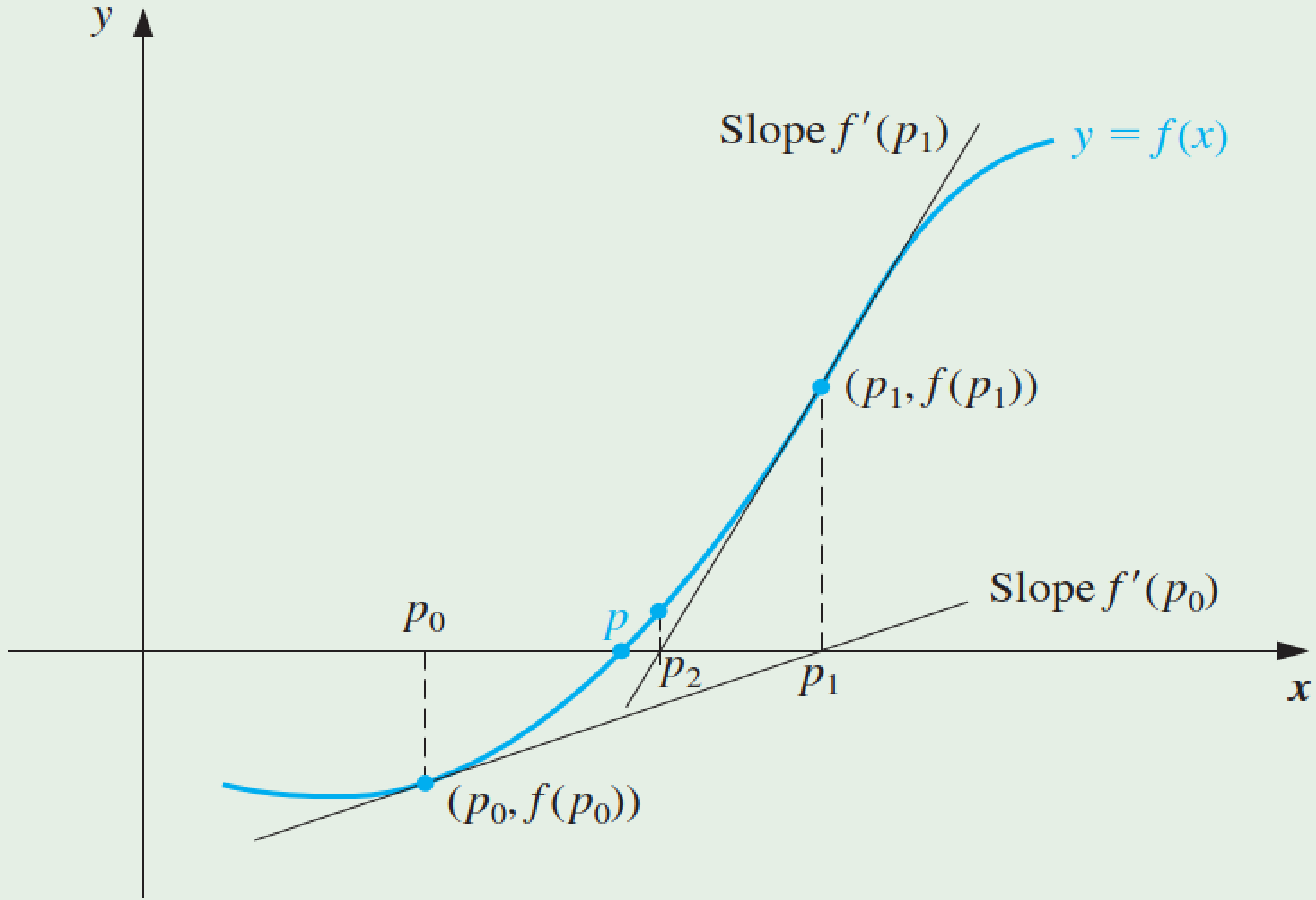
$$0 \approx f(p_0) + (p - p_0)f'(p_0).$$

- Solving for p gives

$$p \approx p_0 - \frac{f(p_0)}{f'(p_0)} \equiv p_1.$$

This sets the stage for Newton's method, which starts with an initial approximation p_0 and generates the sequence $\{p_n\}_{n=0}^{\infty}$, by

$$p_n = p_{n-1} - \frac{f(p_{n-1})}{f'(p_{n-1})} \qquad \text{for } n \geq 1$$



y
Slope $f'(p_1)$
$y = f(x)$
$(p_1, f(p_1))$
Slope $f'(p_0)$
p_0
p
p_2
p_1
x
$(p_0, f(p_0))$

Theoretical importance of the choice of p_0

- The Taylor series derivation of Newton's method points out the importance of an accurate initial approximation.
- The crucial assumption is that the term involving $(p - p_0)^2$ is, by comparison with $|p - p_0|$, so small that it can be deleted.
- This will clearly be false unless p_0 is a good approximation to p.
- If p_0 is not sufficiently close to the actual root, there is little reason to suspect that Newton's method will converge to the root.
- However, in some instances, even poor initial approximations will produce convergence.

- # Finding a Starting Point for Newton's Method

Recall our initial argument that when $|x - x^*|$ is small, then $|x - x^*|^2 \ll |x - x^*|$, and we can neglect the second order term in the Taylor expansion.

In order for Newton's method to converge we need a ***good starting point!***

Problems:

- Difficult to determine the range of initial conditions for which Newton's method converges
- Algorithm often fails to converge
- Problems if $f'(x) = 0$
- Computing the derivative can be "expensive"
- If the zero of $f(x)$ isn't simple, then convergence is ***linear***

Main weakness of **Newton's Method** is computing the derivative

- Computing the derivative can be difficult
- Derivative often needs many more arithmetic operations

One solution is to **Approximate the Derivative**
By definition,

$$f'(x_{n-1}) = \lim_{x \to x_{n-1}} \frac{f(x) - f(x_{n-1})}{x - x_{n-1}}$$

Numerical Methods for Root Finding Problem

- Bisection Method
- Newton’s method
- **Fixed Point Iteration**

Fixed-point Iteration

A *fixed point* for a function is a number at which the value of the function does not change when the function is applied.

The number p is a **fixed point** for a given function g if $g(p) = p$. ■

In this section we consider the problem of finding solutions to fixed-point problems and the connection between the fixed-point problems and the root-finding problems we wish to solve. Root-finding problems and fixed-point problems are equivalent classes in the following sense:

- Given a root-finding problem $f(p) = 0$, we can define functions g with a fixed point at p in a number of ways, for example, as

$$g(x) = x - f(x) \quad \text{or as} \quad g(x) = x + 3f(x).$$

- Conversely, if the function g has a fixed point at p, then the function defined by

$$f(x) = x - g(x)$$

Fixed-point Iteration

Fixed-Point Iteration

A number ξ is a **fixed point** of a function $g(x)$ if $g(\xi) = \xi$.

Suppose that the equation $f(x) = 0$ is written in the form $x = g(x)$; that is, $f(x) = x - g(x) = 0$. Then any fixed point ξ of $g(x)$ is a root of $f(x) = 0$; because $f(\xi) = \xi - g(\xi) = \xi - \xi = 0$. *Thus a root of $f(x) = 0$ can be found by finding a fixed point of $x = g(x)$, which correspond to $f(x) = 0$.*

Finding a root of $f(x) = 0$ by finding a fixed point of $x = g(x)$ immediately suggests an iterative procedure of the following type.

Start with an initial guess x_0 of the root, and form a sequence $\{x_k\}$ defined by

$$x_{k+1} = g(x_k), \; k = 0, 1, 2, \cdots$$

If the sequence $\{x_k\}$ converges, then $\lim_{k\to\infty} x_k = \xi$ will be a root of $f(x) = 0$.

Fixed-point Iteration

The question therefore rises:

> Given $f(x) = 0$, how to write $f(x) = 0$ in the form $x = g(x)$,
>
> so that starting with any x_0 in $[a,\ b]$, the sequence $\{x_k\}$ defined
>
> by $x_{k+1} = g(x_k)$ is guaranteed to converge?

The simplest way to write $f(x) = 0$ in the form $x = g(x)$ is to add x on both sides, that is,

$$x = f(x) + x = g(x).$$

But it does not very often work.

$$Let\ f(x) = x^3 - 6x^2 + 11x - 6 = 0.\ Choose\ a = 2.5,\ b = 4$$

Let's write $f(x) = 0$ *in the form*

$$x = g(x)\ \text{as follows:}\ x = \frac{1}{11}(-x^3 + 6x^2 + 6) = g(x).$$

the iteration $x_{k+1} = g(x_t)$ should converge to the root

$$\begin{aligned}
x_0 & & &= 3.5\\
x_1 &= g(x_0) & &= 3.3295\\
x_2 &= g(x_1) & &= 3.2368\\
x_3 &= g(x_2) & &= 3.1772\\
x_4 &= g(x_x) & &= 3.1359\\
x_5 &= g(x_4) & &= 3.1059\\
x_6 &= g(x_5) & &= 3.0835\\
x_7 &= g(x_6) & &= 3.0664\\
x_8 &= g(x_7) & &= 3.0531\\
x_9 &= g(x_8) & &= 3.0426\\
x_{10} &= g(x_9) & &= 3.0344\\
x_{11} &= g(x_{10}) & &= 3.0778
\end{aligned}$$

The sequence is clearly converging to the root x = 3

$$Find\ a\ root\ of\ f(x) = x - \cos x = 0\ \ in\ [0, \frac{\pi}{2}].$$

$$Let's\ write\ x - \cos x = 0\ \ as\ x = \cos x = g(x).\ \ Then\ |g'(x)| = |\sin x| < 1\ \ in\ (0, \frac{\pi}{2}).$$

$$\begin{aligned}
x_0 &= 0 \\
x_1 &= \cos x_o = 1 \\
x_2 &= \cos x_1 = 0.54 \\
x_3 &= \cos x_2 = 0.86 \\
&\vdots \\
x_{17} &= 0.73956 \\
x_{18} &= \cos x_{17} = 0.73956
\end{aligned}$$

$$f(x) = x^3 - 6x^2 + 11x - 6 = 0.$$

a root of $f(x)$ in $[2.5,\ 4]$

Define $g(x) = x + f(x) = x^3 - 6x^2 + 12x - 6.$

Let's start the iteration $x_{k+1} = g(x_k)$ with $x_0 = 3.5$,

then we have: $x_1 = g(x_0) = g(3.5) = 5.3750$

$$x_2 = g(x_1) = g(5.3750) = 40.4434$$

$$x_3 = g(x_2) = g(40.4434) = 5.6817 \times 10^4$$

$$x_4 = g(x_3) = g(5.6817 \times 10^4) = 1.8340 \times 10^{14}$$

The sequence $\{x_k\}$ is clearly diverging.

Example

The equation $x^3 + 4x^2 - 10 = 0$ has a unique root in $[1, 2]$. There are many ways to change the equation to the fixed-point form $x = g(x)$ using simple algebraic manipulation. For example, to obtain the function g described in part (c), we can manipulate the equation $x^3 + 4x^2 - 10 = 0$ as follows:

$$4x^2 = 10 - x^3, \quad \text{so} \quad x^2 = \frac{1}{4}(10 - x^3), \quad \text{and} \quad x = \pm\frac{1}{2}(10 - x^3)^{1/2}.$$

To obtain a positive solution, $g_3(x)$ is chosen. It is not important for you to derive the functions shown here, but you should verify that the fixed point of each is actually a solution to the original equation, $x^3 + 4x^2 - 10 = 0$.

(a) $x = g_1(x) = x - x^3 - 4x^2 + 10$

(b) $x = g_2(x) = \left(\frac{10}{x} - 4x\right)^{1/2}$

(c) $x = g_3(x) = \frac{1}{2}(10 - x^3)^{1/2}$

(d) $x = g_4(x) = \left(\frac{10}{4 + x}\right)^{1/2}$

(e) $x = g_5(x) = x - \frac{x^3 + 4x^2 - 10}{3x^2 + 8x}$

Example

With $p_0 = 1.5$, Table   lists the results of the fixed-point iteration for all five choices of g.

n	(a)	(b)	(c)	(d)	(e)
0	1.5	1.5	1.5	1.5	1.5
1	−0.875	0.8165	1.286953768	1.348399725	1.373333333
2	6.732	2.9969	1.402540804	1.367376372	1.365262015
3	−469.7	$(-8.65)^{1/2}$	1.345458374	1.364957015	1.365230014
4	1.03×10^8		1.375170253	1.365264748	1.365230013
5			1.360094193	1.365225594	
6			1.367846968	1.365230576	
7			1.363887004	1.365229942	
8			1.365916734	1.365230022	
9			1.364878217	1.365230012	
10			1.365410062	1.365230014	
15			1.365223680	1.365230013	
20			1.365230236		
25			1.365230006		
30			1.365230013		

Prime Objective

- In what follows, it is important not to lose sight of our prime objective:
- Given a function $f(x)$ where $a \leq x \leq b$, find values p such that

$$f(p) = 0$$

- Given such a function, $f(x)$, we now construct an auxiliary function $g(x)$ such that

$$p = g(p)$$

  whenever $f(p) = 0$ (this construction is not unique).
- The problem of finding p such that $p = g(p)$ is known as the fixed point problem.

A Fixed Point

If g is defined on $[a, b]$ and $g(p) = p$ for some $p \in [a, b]$, then the function g is said to have the fixed point p in $[a, b]$.

Note

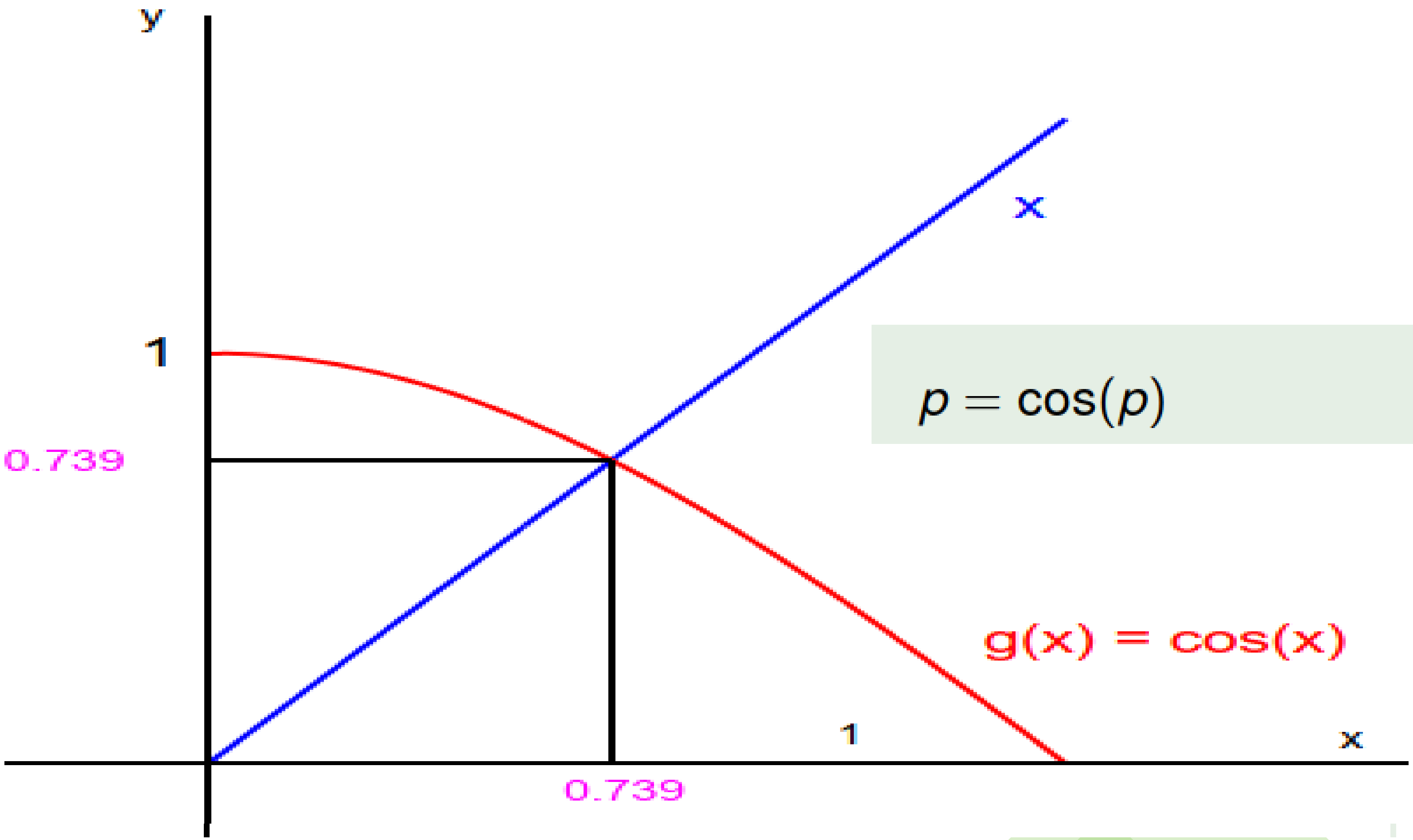
- The fixed-point problem turns out to be quite simple both theoretically and geometrically.
- The function $g(x)$ will have a fixed point in the interval $[a, b]$ whenever the graph of $g(x)$ intersects the line $y = x$.

The Equation $f(x) = x - \cos(x) = 0$

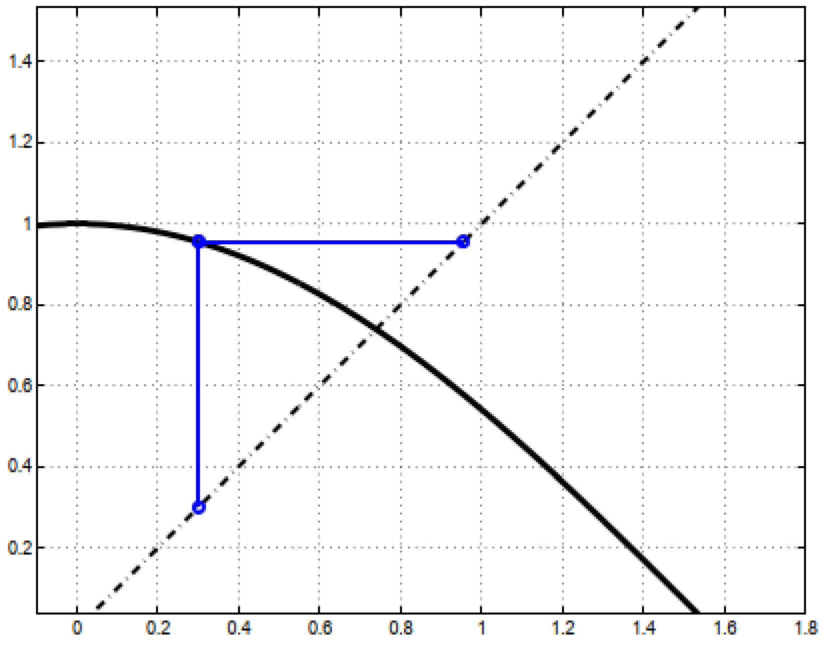
If we write this equation in the form:

$$x = \cos(x)$$

then $g(x) = \cos(x)$.

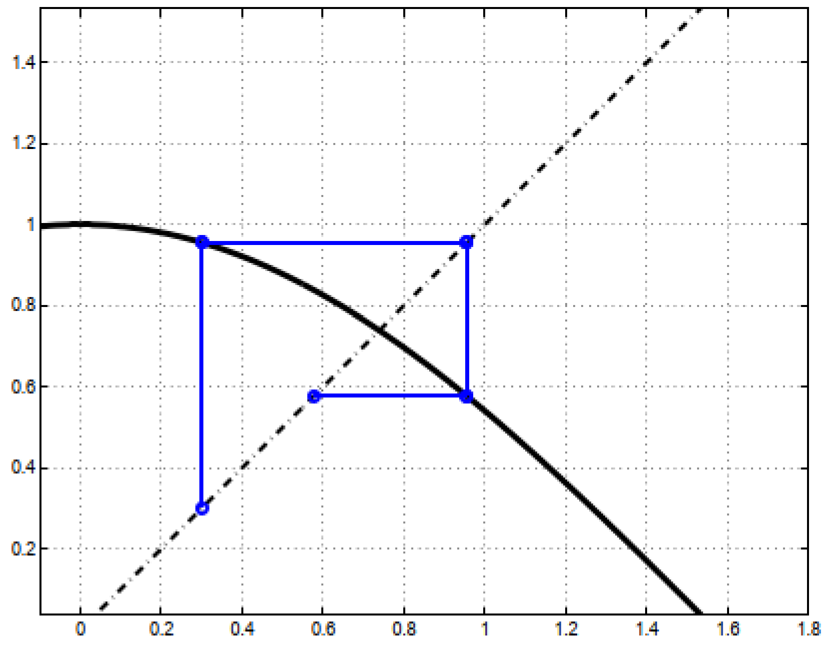


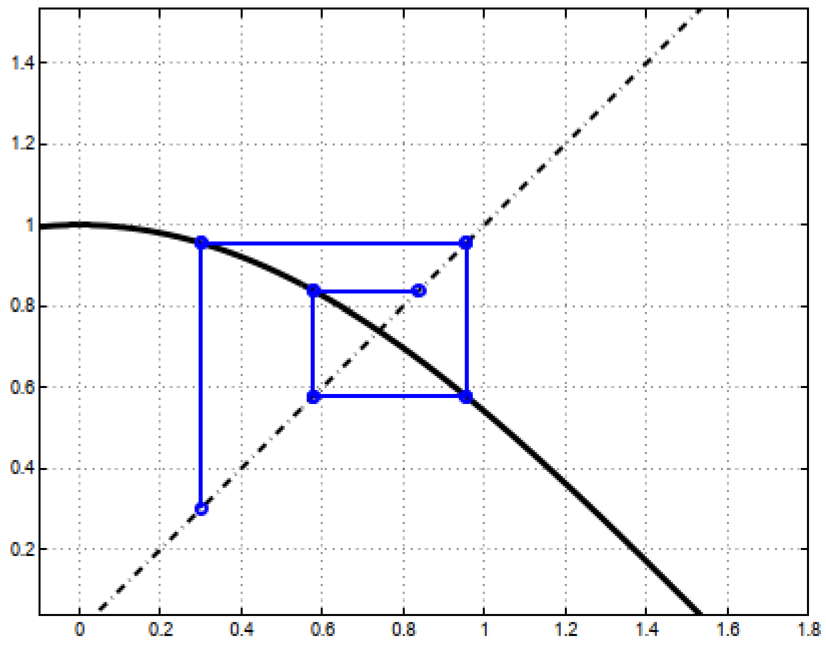
$p = \cos(p)$ $\quad$ $p \approx 0.739$



Iteration #1:
$p = 0.3,$
$\cos(p) = 0.92106099400289$
$p = 0.92106099400289$

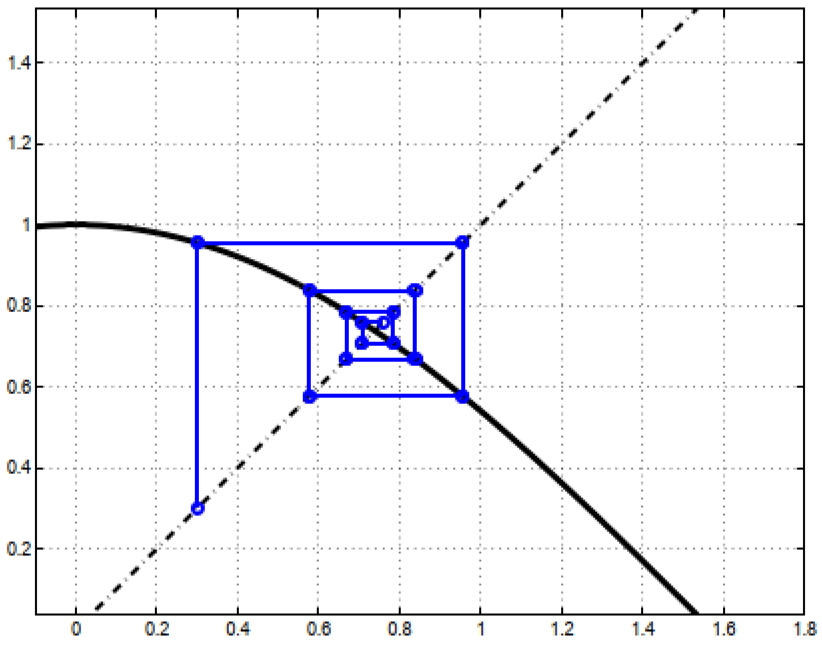
Iteration #2:
$p = 0.92106099400289,$
$\cos(p) = 0.60497568726594$
$p = 0.60497568726594$





Iteration #3:
$p = 0.60497568726594,$
$\cos(p) = 0.82251592555039$
$p = 0.82251592555039$

Iteration #7:
$p = 0.71278594551835,$
$\cos(p) = 0.75654296195845$
$p = 0.75654296195845$



Proof of the Fixed-Proof Theorem

Existence:
If $a = g(a)$, then a is a fixed point. If $b = g(b)$, then b is a fixed point. If not, **because of the assumption 1**, $g(a) > a$, and $g(b) < b$. Thus, $f(a) = g(a) - a > 0$, and $f(b) = g(b) - b < 0$. Also $f(x) = g(x) - x$ is continuous on $[a, b]$.
Thus, by the **Intermediate Mean Value Theorem**, there is a root of $f(x)$ in $[a,\ b]$. This proves the existence.

Corollary *Let $g(x)$ be continuously differentiable in some open interval containing the fixed point ξ, and let $|g'(\xi)| < 1$, so that the iteration $x_{k+1} = g(x_k)$ converges whenever*

Example *Let $f(x) = x^3 - 6x^2 + 11x - 6 = 0$. Choose $a = 2.5$, $b = 4$. Let's write $f(x) = 0$ in the form $x = g(x)$ as follows: $x = \frac{1}{11}(-x^3 + 6x^2 + 6) = g(x)$.*

Then

$$g'(x) = \frac{1}{11}(-3x^2 + 12x).$$

It is easy to verify graphically and analytically that $|g'(x)|$ is less than 1 for all x is $(2.5, 4)$, (note that $g'(4) = 0$, and $g'(2.5) = 1.0227$).

Order of Convergence

In the previous section we discussed four different algorithms for finding the root of f (x) = 0. why one method would be faster than another...

Now, we are going to look at the error analysis of iterative methods, and we will quantify the speed of our methods.

The simplex method for solving linear optimization problems stops after a finite number of steps. In general, however, algorithms for solving optimization problems will generate an infinite sequence and one hopes that, as tends to infinity, an acceptable solution will be produced. In case of convergence, it is natural to ask how fast the sequence actually converges. There are several orders of convergence

Order of Convergence

Order of Convergence

Suppose $\{p_n\}_{n=0}^{\infty}$ is a sequence that converges to p, with $p_n \neq p$ for all n. If positive constants λ and α exist with

$$\lim_{n\to\infty} \frac{|p_{n+1} - p|}{|p_n - p|^{\alpha}} = \lambda,$$

then $\{p_n\}_{n=0}^{\infty}$ **converges to p of order α, with asymptotic error constant λ.** ■

In general, a sequence with a high order of convergence converges more rapidly than a sequence with a lower order. The asymptotic constant affects the speed of convergence but not to the extent of the order. Two cases of order are given special attention.

(i) If $\alpha = 1$ (and $\lambda < 1$), the sequence is **linearly convergent**.

(ii) If $\alpha = 2$, the sequence is **quadratically convergent**.

Bottom line: High order (α) $\Rightarrow$ Faster convergence (more desirable).
λ has an effect, but is less important than the order.

Order of Convergence

When $\alpha = 1$ the sequence is **linearly convergent.**

When $\alpha = 2$ the sequence is **quadratically convergent.**

When $\alpha < 1$ the sequence is **sub-linearly convergent** (very undesirable, or "painfully slow.")

When $((\alpha = 1 \text{ and } \lambda = 0) \text{ or } 1 < \alpha < 2)$, the sequence is **super-linearly convergent**.

Illustration Suppose that $\{p_n\}_{n=0}^{\infty}$ is linearly convergent to 0 with

$$\lim_{n\to\infty} \frac{|p_{n+1}|}{|p_n|} = 0.5$$

and that $\{\tilde{p}_n\}_{n=0}^{\infty}$ is quadratically convergent to 0 with the same asymptotic error constant,

$$\lim_{n\to\infty} \frac{|\tilde{p}_{n+1}|}{|\tilde{p}_n|^2} = 0.5.$$

For simplicity we assume that for each n we have

$$\frac{|p_{n+1}|}{|p_n|} \approx 0.5 \quad \text{and} \quad \frac{|\tilde{p}_{n+1}|}{|\tilde{p}_n|^2} \approx 0.5.$$

For the linearly convergent scheme, this means that

$$|p_n - 0| = |p_n| \approx 0.5|p_{n-1}| \approx (0.5)^2|p_{n-2}| \approx \cdots \approx (0.5)^n|p_0|,$$

whereas the quadratically convergent procedure has

$$\begin{aligned}
|\tilde{p}_n - 0| = |\tilde{p}_n| &\approx 0.5|\tilde{p}_{n-1}|^2 \approx (0.5)[0.5|\tilde{p}_{n-2}|^2]^2 = (0.5)^3|\tilde{p}_{n-2}|^4 \\
&\approx (0.5)^3[(0.5)|\tilde{p}_{n-3}|^2]^4 = (0.5)^7|\tilde{p}_{n-3}|^8 \\
&\approx \cdots \approx (0.5)^{2^n-1}|\tilde{p}_0|^{2^n}.
\end{aligned}$$

Table illustrates the relative speed of convergence of the sequences to 0 if $|p_0| = |\tilde{p}_0| = 1$.

n	Linear Convergence Sequence $\{p_n\}_{n=0}^{\infty}$ $(0.5)^n$	Quadratic Convergence Sequence $\{\tilde{p}_n\}_{n=0}^{\infty}$ $(0.5)^{2^n-1}$
1	5.0000×10^{-1}	5.0000×10^{-1}
2	2.5000×10^{-1}	1.2500×10^{-1}
3	1.2500×10^{-1}	7.8125×10^{-3}
4	6.2500×10^{-2}	3.0518×10^{-5}
5	3.1250×10^{-2}	4.6566×10^{-10}
6	1.5625×10^{-2}	1.0842×10^{-19}
7	7.8125×10^{-3}	5.8775×10^{-39}

The quadratically convergent sequence is within 10^{-38} of 0 by the seventh term. At least 126 terms are needed to ensure this accuracy for the linearly convergent sequence.

Quadratically convergent sequences are expected to converge much quicker than those that converge only linearly

Suppose we have two sequences converging to zero:

$$\lim_{n\to\infty} \frac{|p_{n+1}|}{|p_n|} = \lambda_p, \qquad \lim_{n\to\infty} \frac{|q_{n+1}|}{|q_n|^2} = \lambda_q$$

Roughly this means that

$$|p_n| \approx \lambda_p |p_{n-1}| \approx \lambda_p^n |p_0|, \qquad |q_n| \approx \lambda_q |q_{n-1}|^2 \approx \lambda_q^{2^n-1} |q_0|^{2^n}$$

Now, assume $\lambda_p = \lambda_q = 0.9$ and $p_0 = q_0 = 1$, we get the following

n	p_n	q_n
0	1	1
1	0.9	0.9
2	0.81	0.729
3	0.729	0.4782969
4	0.6561	0.205891132094649
5	0.59049	0.0381520424476946
6	0.531441	0.00131002050863762
7	0.4782969	0.00000154453835975
8	0.43046721	0.00000000000021470

Table (Linear vs. Quadratic): A dramatic difference! After 8 iterations, q_n has 11 correct decimals, and p_n still *none*. q_n roughly doubles the number of correct digits in every iteration. Here p_n needs more than 20 iterations/digit-of-correction.

Convergence of General Fixed Point Iteration

Theorem

Let $g \in C[a,b]$ be such that $g(x) \in [a,b]$ for all $x \in [a,b]$. Suppose, in addition that $g'(x)$ is continuous on (a,b) and there is a positive constant $k < 1$ so that

$$|g'(k)| \leq k, \quad \forall x \in (a,b)$$

If $\mathbf{g}'(\mathbf{p}^*) \neq \mathbf{0}$, then for any number p_0 in $[a,b]$, the sequence

$$p_n = g(p_{n-1}), \quad n \geq 1$$

converges **only linearly** *to the unique fixed point p^* in $[a,b]$.*

In a sense, this is bad news since we like fast convergence...

Speeding up Convergence of Fixed Point Iteration

Bottom Line: The theorem tells us that if we are looking to design rapidly converging fixed point schemes, we must design them so that $g'(p^*) = 0$...

We state the following without proof:

Theorem

Let p^* be a solution of $p = g(p)$. Suppose $g'(p^*) = 0$, and $g''(x)$ is continuous and strictly bounded by M on an open interval I containing p^*. Then there exists a $\delta > 0$ such that, for $p_0 \in [p^* - \delta, p^* + \delta]$ the sequence defined by $p_n = g(p_{n-1})$ converges **at least quadratically** *to p^*. Moreover, for sufficiently large n*

$$|p_{n+1} - p^*| < \frac{M}{2}|p_n - p^*|^2$$

Practical Application of the Theorems

The theorems tell us:

"Look for quadratically convergent fixed point methods among functions whose derivative is zero at the fixed point."

We want to solve: $f(x) = 0$ using fixed point iteration. We write the problem as an equivalent fixed point problem:

$g(x) = x - f(x)$	Solve: $x = g(x)$	
$g(x) = x - \alpha f(x)$	Solve: $x = g(x)$	α a constant
$g(x) = x - \Phi(x) f(x)$	Solve: $x = g(x)$	$\Phi(x)$ differentiable

We use the most general form (the last one).
Remember, we want $g'(p^*) = 0$ when $f(p^*) = 0$.

Newton's Method for $\sqrt{a}$ converges quadratically if $p_0 > 0$

- $\{p_k\}_{k=1}^{\infty}$ bounded below by $\sqrt{a}$:

$$p_k - \sqrt{a} = \frac{1}{2}\left(p_{k-1} + \frac{a}{p_{k-1}}\right) - \sqrt{a} = \frac{\left(p_{k-1} - \sqrt{a}\right)^2}{2p_{k-1}} \geq 0, \quad k = 1, 2, \cdots,$$

- $\{p_k\}_{k=1}^{\infty}$ monotonically decreasing:

$$p_{k+1} - p_k = \frac{1}{2}\left(p_k + \frac{a}{p_k}\right) - p_k = \frac{a - p_k^2}{2p_k} \leq 0, \quad k = 1, 2, \cdots,$$

- $\lim_{k\to\infty} p_k \stackrel{def}{=} p$ exists and satisfies Newton Iteration:

$$p = \frac{1}{2}\left(p + \frac{a}{p}\right) \geq \sqrt{a}, \quad \text{therefore} \quad p = \sqrt{a}.$$

- Newton's Method quadratically convergent

$$\lim_{k\to\infty} \frac{|p_k - \sqrt{a}|}{|p_{k-1} - \sqrt{a}|^2} = \lim_{k\to\infty} \frac{1}{2p_{k-1}} = \frac{1}{2\sqrt{a}}.$$

Accelerating Convergence

Accelerating Convergence: Aitken's Δ^2 Method

Accelerating Convergence: Steffensen's Method

Homework

The equation $x^2 - 10\cos x = 0$ has two solutions, ± 1.3793646. Use Newton's method to approximate the solutions to within 10^{-5} with the following values of p_0.

a. $p_0 = -100$ **b.** $p_0 = -50$ **c.** $p_0 = -25$

d. $p_0 = 25$ **e.** $p_0 = 50$ **f.** $p_0 = 100$

Use the Bisection method to find solutions accurate to within 10^{-5} for the following problems.

a. $x - 2^{-x} = 0$ for $0 \le x \le 1$

b. $e^x - x^2 + 3x - 2 = 0$ for $0 \le x \le 1$

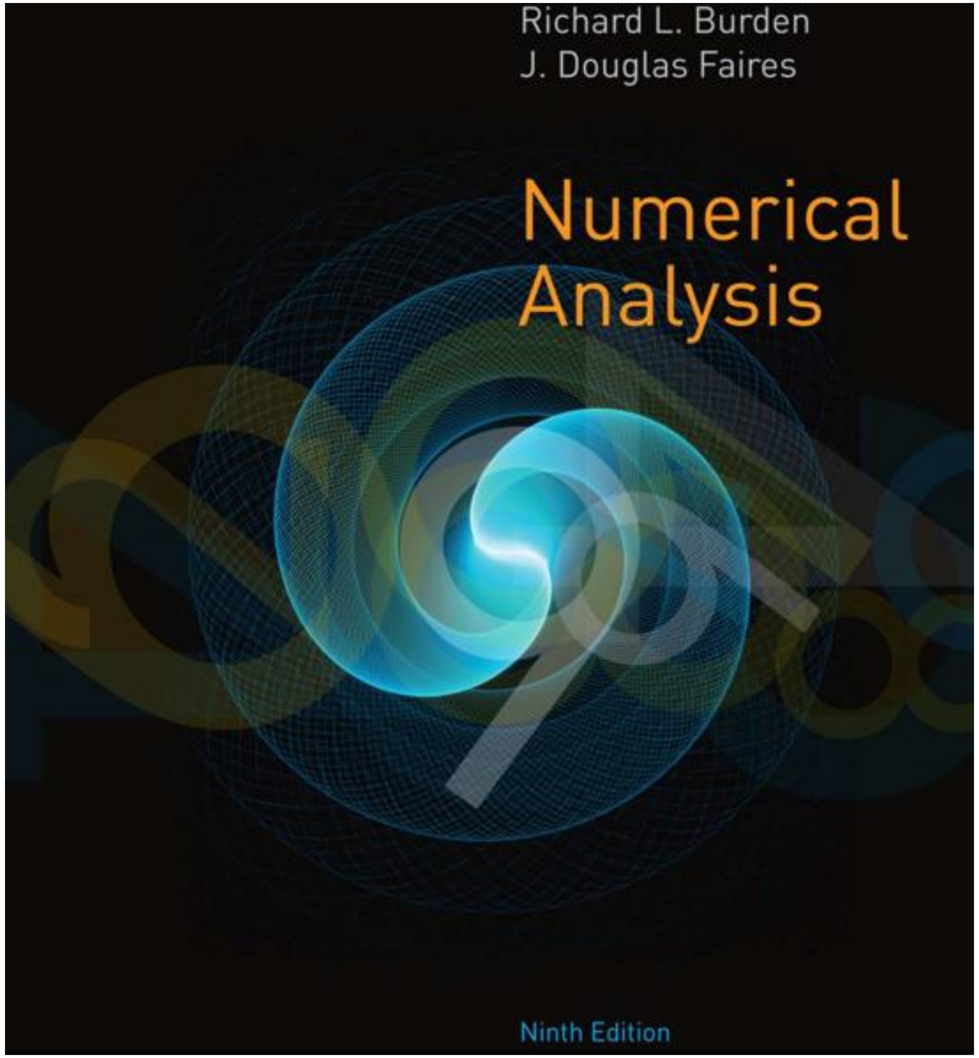
c. $2x\cos(2x) - (x+1)^2 = 0$ for $-3 \le x \le -2$ and $-1 \le x \le 0$

d. $x\cos x - 2x^2 + 3x - 1 = 0$ for $0.2 \le x \le 0.3$ and $1.2 \le x \le 1.3$

Find an approximation to $\sqrt[3]{25}$ correct to within 10^{-4} using the Bisection Algorithm.

Use a fixed-point iteration method to determine a solution accurate to within 10^{-2} for $x^3 - x - 1 = 0$ on $[1, 2]$. Use $p_0 = 1$.

References


Richard L. Burden
J. Douglas Faires
Numerical
Analysis
Ninth Edition