PART 1

Solving Equations

1

Solving Nonlinear Equations

1.1. Introduction

Let $F\colon \mathbb{R} \to \mathbb{R}$ be a function defined on a subset $D \subset \mathbb{R}$. Suppose that we wish to find the roots of the equation $F(x) = 0$, if they exist. Most theorems that guarantee the existence of roots for equations of the form $F(x) = 0$ do not specify a way to construct these roots, and we are usually not capable of solving the problem analytically, except in a few special cases. Therefore, we often need to resort to numerical methods to find approximate solutions of the equation $F(x) = 0$ [RAD 09, BAK 76].

There are many possible numerical methods to choose from, and each has its own advantages and disadvantages (we will only present the most common methods in this chapter). Before trying to solve a problem numerically, we need to know that the solution exists and is unique [QUA 04, RAD 10]. Therefore, the process of numerically solving the equation $F(x) = 0$ can be divided into two parts:

1) First, we must show the existence of real-valued solutions and separate each solution that we wish to approximate in an interval $[a, b] \subset D$.

2) Second, we must choose a numerical method to approximate the isolated roots.

1.2. Separating the roots

DEFINITION.– *We say that the root $\bar{x}$ of the equation $F(x) = 0$ is separable if there exists an interval $[a, b] \subset D$, such that $\bar{x}$ is the only root contained in $[a, b]$.*

If so, the root $\bar{x} \in [a, b]$ is said to be separated or isolated. In practice, we can separate the roots of the equation $F(x) = 0$ either graphically (by looking for the points at which the graph of F intersects with the $(o\vec{x})$ axis) or analytically, by applying the intermediate value theorem (see the appendix on standard results from analysis, Appendix 2).

EXAMPLE.–

1) The equation $x^3 - 6x + 2 = 0$ has three real roots that are separated by the intervals $[-3, -2]$, $[0, 1]$ and $[2, 3]$.

2) The equation $e^x \sin x - 1 = 0 \Leftrightarrow \sin x = e^{-x}$ has two roots that are separated by the intervals $[0, 1]$ and $[3, 4]$.

3) The equation $x \ln x - 1 = 0$ has a single root in $[1, 2]$.

4) The equation $2x^4 + 3x^3 - 4x - 5 = 0$ has two real roots separated by the intervals $[-2, -1]$ and $[1, 2]$.

5) The equation $(1+x)e^{1-x} - 3/2 = 0$ has two real roots separated by the intervals $[-1, 0]$ and $[1, 2]$.

1.3. Approximating a separated root

In this section, we will assume that the root $\bar{x}$ has already been isolated by the interval $[a, b]$.

We will present some of the standard methods that can be used to find an approximate value for $\bar{x}$, and we will study their properties and rates of convergence. Concretely, when we speak of approximating $\bar{x}$ up to ϵ, we mean finding an approximate value $\widetilde{x}$ that is known to satisfy $|\bar{x} - \widetilde{x}| \leq \epsilon$ (where ϵ is a small real number; the smaller this ϵ, the more precise the algorithm).

1.3.1. *Bisection method (or dichotomy method)*

Suppose that $\bar{x}$ is a simple separable root in $[a, b]$ ($F(a)F(b) < 0$). The bisection method constructs a sequence of intervals $I_n = [a_n, b_n]$ from the initial interval $[a, b]$, such that each I_n contains $\overline{x}$ and has a width equal to half of the width of I_{n-1}.

The underlying idea of the method is to construct the following three sequences (a_n), (b_n), and (x_n):

Define $a_0 = a$; $b_0 = b$, $x_0 = \frac{1}{2}(a_0 + b_0)$, and for $n \geq 1$:

– If $F(a_{n-1})F(x_{n-1}) < 0$, define $a_n = a_{n-1}$, $b_n = x_{n-1}$, $x_n = \frac{1}{2}(a_n + b_n)$.

– If $F(a_{n-1})F(x_{n-1}) > 0$, define $a_n = x_{n-1}$, $b_n = b_{n-1}$, $x_n = \frac{1}{2}(a_n + b_n)$.

– If $F(a_{n-1})F(x_{n-1}) = 0$, then $\overline{x} = x_{n-1}$: this means that we have found the exact solution, hence the process is terminated.

In this way, at each iteration, we select the half-interval that contains the root $\bar{x}$. After n iterations, we have found an interval containing $\bar{x}$ with width $\dfrac{b-a}{2^n}$.

THEOREM.– *The sequence (x_n) constructed by the bisection algorithm converges to the root $\bar{x}$.*

An upper bound for the error committed by approximating $\bar{x}$ by x_n is given by: $|x_n - \overline{x}| \leq \dfrac{b-a}{2^{n+1}}$.

REMARK.–

1) Similarly, the sequences (a_n) and (b_n) converge to $\bar{x}$, and $|\bar{x}-a_n| \leq (b-a)/2^n$; $|\bar{x} - b_n| \leq (b-a)/2^n$.

2) To guarantee a precision of ϵ when approximating $\bar{x}$ by x_n, we can simply pick n, such that: $n \geq \ln \left[\dfrac{(b-a)}{\epsilon} \right] / \ln 2 - 1$.

This method has the following properties:

– straightforward algorithm (based on the intermediate value theorem);

– can be applied to non-analytic functions;

– cannot be applied to double roots;

– cannot be applied to multiple equations;

– very slow (to achieve high precision, we need a high number of iterations).

Owing to these limitations, we need other methods in some situations.

EXAMPLE.– Suppose that we wish to find a zero of the following function using the bisection method:

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

This function satisfies $f(0) = -1$, $f(2) = 0.818595$. It is continuous on $[0, 2]$, and $f(0)f(2) < 0$. By the intermediate value theorem, the equation $f(x) = 0$ has at least one root $\alpha \in [0, 2]$. We can therefore use this interval to initialize the bisection method. Applying this method to f on the interval $[0, 2]$, we find:

i	x_i	i	x_i
1	1.000000	7	1.109375
2	1.500000	8	1.117188
3	1.250000	9	1.113281
4	1.125000	10	1.115234
5	1.062500	11	1.114258
6	1.093750	12	1.114258

An approximate value for the root of f is therefore given by $x_{12} = 1.114258$.

1.3.2. *Fixed-point method*

DEFINITION.– *The solutions of the equation $f(x) = x$ are said to be the fixed points of the function f.*

THEOREM.– *If f is continuous on $[a, b]$ and $f([a, b]) \subset [a, b]$, then f has at least one fixed point between a and b (in other words, $\exists c \in]a, b[\,/\, f(c) = c$).*

Suppose that $\bar{x} \in [a, b]$ is an isolated root of the equation $F(x) = 0$. The fixed-point method defines and studies a function f such that $\bar{x}$ is a fixed point of f (i.e. such that $F(x) = 0 \Leftrightarrow f(x) = x$).

EXAMPLE.– For example, for the function $F(x) = (1 + x)e^{1-x} - \frac{3}{2}$, we can define $f(x) = \frac{3}{2}e^{x-1} - 1$.

Now, instead of searching for $\bar{x}$ as a root of $F(x)$, we can find it as a fixed point of $f(x)$ by taking advantage of the following property:

If the recursive sequence (x_n) defined by x_0 arbitrary; $x_n = f(x_{n-1})$, $n \geq 1$, converges, then its limit is a fixed point of the function f (see Figure 1.1).

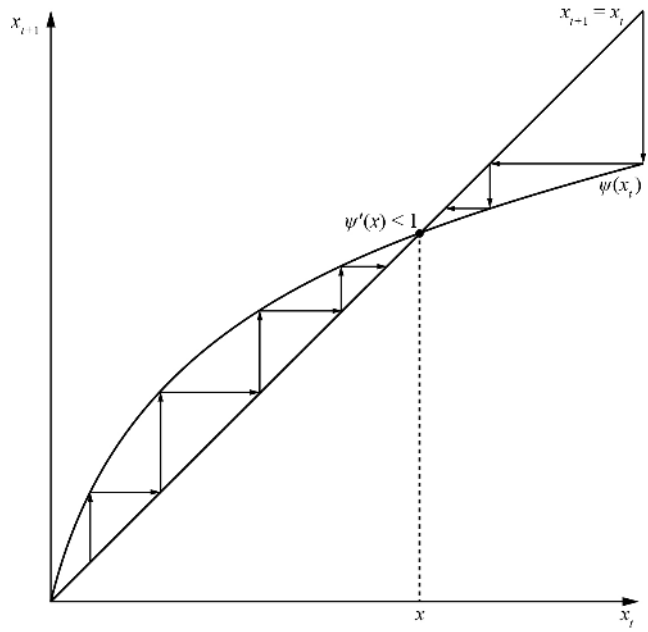

Figure 1.1. *Fixed-point method*

The idea of the method is to choose a function f for the original function F, such that:

– $F(x) = 0 \rightarrow f(x) = x$.

– The sequence (x_n) defined by $x_0 \in [a, b]$; $x_n = f(x_{n-1})$ for $n \geq 1$ converges to $\overline{x}$.

The problem of approximating $\bar{x}$ up to ϵ is therefore reduced to that of computing x_n, such that $|x_n - \bar{x}| \leq \epsilon$.

REMARK.– For any given function $F(x) = 0$, there are an infinite number of choices of f such that $F(x) = 0$ is equivalent to $f(x) = x$ (e.g. $f(x) = x + \lambda F(x)$; $\lambda \in \mathbb{R}^*$). The challenge of the fixed-point method lies in choosing f in such a way that the sequence (x_n) converges, and does so as quickly as possible. There are two natural criteria for selecting the best function: convergence and rate of convergence. The choice of the initial point x_0 also influences both of these criteria.

EXAMPLE.– Suppose that we wish to compute the value of $\sqrt{2}$. This is equivalent to finding the positive root $\alpha = \sqrt{2}$ of the function

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

that is, solving a nonlinear equation.

It is easy to check that $\alpha = \sqrt{2}$ is a fixed point of the function $\phi(x) = -\frac{1}{4}x^2 + x + \frac{1}{2}$ simply by noting that $\phi(\sqrt{2}) = \sqrt{2}$. Moreover, $\forall x \in [1, 2]$, $|\phi'(x)| \leq \frac{1}{2} = C$. We can therefore define the sequence $x_0 \in [1, 2]$ and $x_{n+1} = \phi(x_n)$, and apply the mean value theorem to deduce that $\exists \eta \in [1, 2]$, such that

$$|x_{n+1} - \alpha| = |\phi(x_n) - \phi(\alpha)| = |\phi'(\eta)||x_n - \alpha|.$$

Therefore,

$$|x_n - \alpha| \leq C^n |x_0 - \alpha|, \ \forall n \geq 0.$$

This shows that the sequence x_n converges to the root α. We need to perform 34 iterations of the fixed-point method to compute an approximate value for $\sqrt{2}$ that is accurate to ten digits after the decimal point.

1.3.3. *First convergence criterion*

THEOREM.– *Let $\overline{x} \in [a, b]$ be an isolated root of $F(x)$, and suppose that $f(x)$ is a continuous and differentiable function on $[a, b]$ satisfying $f(\overline{x}) = \overline{x}$.*

If there exists a real number k, $0 < k < 1$, *such that:*

i) $f([a,b]) \subset [a,b]$;

ii) $\forall x \in [a,b]$, $|f'(x)| \leq k$;

then the recursive sequence (x_n) *defined by* $x_0 \in [a,b]; x_n = f(x_{n-1})$ *converges to* $\overline{x}$.

An upper bound for the error committed by approximating $\overline{x}$ *with* x_n *is given by:*

$$|\overline{x} - x_n| \leq k^n |\overline{x} - x_0|.$$

1.3.4. *Iterative stopping criteria*

The idea of this method is to find $\overline{x}$ by computing the limit of a convergent numerical sequence. In practice, we need to find an approximation of $\overline{x}$ up to a certain accuracy ϵ in a finite number of iterations (as few as possible). As $\bar{x}$ is unknown, we cannot directly apply the stopping criterion $|\overline{x} - x_n| \leq \epsilon$.

1.3.4.1. *Choosing the number of iterations*

To guarantee an accuracy of ϵ when approximating $\bar{x}$ by x_n, we can simply choose the smallest integer n, such that $k^n|b-a| \leq \epsilon$. Let $n \geq \ln\left(\dfrac{\epsilon}{b-a}\right) / \ln k$.

The smaller the number k, the faster the convergence.

The condition $k^n|b-a| \leq \epsilon$ is an extremely strong condition that is sufficient but not necessary! Before we can apply this condition, we need to know the value of k. If this is not possible, we will need an alternative stopping condition.

1.3.4.2. *Testing the absolute value*

In some cases, we can use a stopping condition based on the absolute value of the difference between two consecutive approximations of $\overline{x}$, which can be expressed in the form of the inequality $|x_n - x_{n-1}| \leq \epsilon$.

Of course, this condition is not sufficient to guarantee that x_n approximates the exact solution $\bar{x}$ up to ϵ, since $|x_n - x_{n-1}| \leq \epsilon$ does not imply that $|x_n - \overline{x}| \leq \epsilon$ in general. Whether or not the latter inequality holds depends on how the sequence (x_n) converges to $\bar{x}$.

We can write $|x_n - \bar{x}|$ as a function of $|x_n - x_{n-1}|$ as follows:

CASE 1.– $-1 < f'(x) < 0$ on $[a, b]$.

In this case, the sequences (x_{2n}) and (x_{2n+1}) are adjacent and their limit $\bar{x}$ is always located between any two consecutive terms of the sequence (x_n). Therefore, in this specific case, $|x_n - \overline{x}| \leq |x_n - x_{n-1}|$, and the stopping condition $|x_n - x_{n-1}| \leq \epsilon$ is sufficient to guarantee that x_n and x_{n-1} approximate $\bar{x}$ up to ϵ, one from above and the other from below. Therefore, we can approximate $\bar{x}$ up to ϵ without needing to find k.

CASE 2.– $0 < f'(x) < 1$ (or if the sign of f' is not known on $[a, b]$).

In this case, the sequence (x_n) converges monotonically, and we can show that

$$|x_n - \bar{x}| \leq \frac{1}{1-k}|x_{n+1} - x_n|.$$

Therefore, to guarantee a precision of ϵ when approximating $\bar{x}$ by x_n, we can simply apply the stopping condition $|x_{n+1} - x_n| \leq \epsilon(1 - k)$.

1.3.5. *Second convergence criterion (local criterion)*

THEOREM.– *Let $\bar{x} \in [a, b]$ be an isolated root of $F(x)$, and let $f(x)$ be a $\mathcal{C}^1$ function, such that $f(\bar{x}) = \bar{x}$.*

1) If $|f'(\bar{x})| < 1$, then there exists a neighborhood V of $\bar{x}$ such that the recursive sequence (x_n) associated with f converges to $\bar{x}$, $\forall x_0 \in V$.

Furthermore, (x_n) converges monotonically if $0 \leq f'(\bar{x}) < 1$, and converges while oscillating around $\overline{x}$ if $-1 < f'(\bar{x}) < 0$.

The set V is called the domain of attraction of $\bar{x}$.

2) If $|f'(\bar{x})| > 1$, then, $\forall x_0$, the sequence $x_n = f(x_{n-1})$ does not converge to $\bar{x}$.

3) If $|f'(\bar{x})| = 1$, then we cannot draw any conclusions about the convergence behavior of the sequence (x_n), which is unstable (either convergent or divergent) and very sensitive to rounding errors.

REMARK.–

1) The convergence criterion $|f'(\bar{x})| < 1$ only holds in the neighborhood of $\bar{x}$ (we might not have convergence for every x_0 in $[a, b]$). We therefore need to choose the right x_0 (in the domain of attraction) to apply this criterion.

2) The smaller the value of $|f'(\bar{x})|$, the faster the convergence ($|f'(\bar{x})|$ is known as the rate of convergence).

3) In order to use this criterion, we need to compute $f'(\bar{x})$. This is not always possible (since $\bar{x}$ is an unknown in the problem).

The properties of this method can be summarized as follows:

– mostly only useful in theoretical contexts;

– constructive method;

– need a strategy to choose f and x_0.

1.3.6. *Newton's method (or the method of tangents)*

Suppose that we have an isolated root $\overline{x}$ of the equation $F(x) = 0$ in the interval $[a, b]$, and suppose that $F'(x)$ does not vanish on $[a, b]$.

Newton's method chooses the following specific function f to study the roots of F (see Figure 1.2):

$$f(x) = x - \frac{F(x)}{F'(x)}. \qquad [1.1]$$

Clearly, $\bar{x}$ is a fixed point of f. What can we say about the recursive sequence (x_n) defined by $x_0 \in [a, b]$, $x_{n+1} = f(x_n)$?

THEOREM.– *Let $\bar{x}$ be an isolated root of the equation $F(x) = 0$ in $[a, b]$. Suppose that F has continuity class $\mathcal{C}^2$ and that F' does not vanish on $[a, b]$.*

Then there exists a neighborhood V of $\bar{x}$ on which the sequence (x_n) constructed by Newton's method converges to $\bar{x}$, $\forall x_0 \in V$.

THEOREM.– *Let $\bar{x}$ be an isolated root of the equation $F(x) = 0$ in $[a, b]$. Suppose that F' does not vanish on $[a, b]$, and that F'' is continuous on $[a, b]$. Let $M = \max |F''(x)|/2$ on $[a, b]$. If $x_0 \in [a, b]$ and $|x_0 - \bar{x}| < \frac{1}{M}$, then the sequence (x_n) constructed by Newton's method converges to the solution $\bar{x}$.*

An upper bound for the error committed by approximating $\bar{x}$ with x_n is given by:

$$|x_n - \bar{x}| \leq \frac{1}{M}(M|x_0 - \bar{x}|)^{2^n}.$$

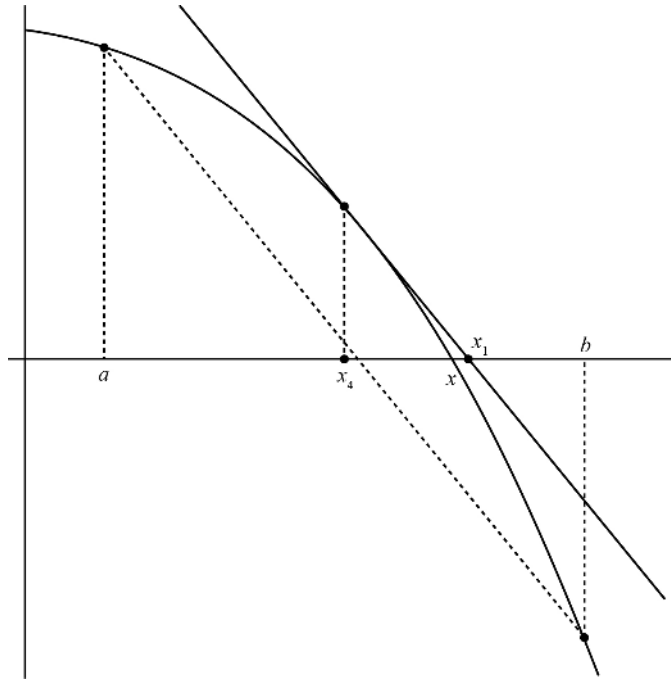

Figure 1.2. *Newton's method*

REMARK.–

1) Note that $f(\bar{x}) = 0$, so convergence is guaranteed on some neighborhood V of $\bar{x}$.

2) The convergence condition of Newton's process is expressed in terms of the choice of initial value $|x_0 - \bar{x}| < \frac{1}{M}$. This condition is satisfied $\forall x_0 \in [a, b]$ whenever $|b - a| < \frac{1}{M}$.

Therefore, the challenge of Newton's method is to choose the right x_0.

3) The following expression gives an alternative upper bound for the error:

$$|x_n - \overline{x}| \leq \frac{1}{L}(L|x_0 - \overline{x}|)^{2^n},$$

where $L = \max \left| \dfrac{F''(x)}{2F'(x)} \right|$.

4) Theorem 1.3 gives a sufficient condition on the choice of x_0 to guarantee convergence, but this condition is not necessary. The assumptions of the theorem

are often difficult to satisfy (since $\bar{x}$ is unknown). Therefore, in practice, we work backwards, as follows:

We define the sequence $x_{n+1} = f(x_n)$, then pick an arbitrary $x_0 \in [a, b]$; if this sequence converges, the limit is the desired solution; otherwise, we choose another x_0 in $[a, b]$.

5) Newton's method achieves quadratic convergence, which is faster than the previous method.

1.3.7. *Secant method*

In cases where we cannot compute the value of $F'(x)$ explicitly, we can instead use the approximation:

$$\frac{F(x_n) - F(x_{n-1})}{x_n - x_{n-1}}. \qquad [1.2]$$

This transforms the recurrence formula into:

$$x_{n+1} = x_n - F(x_n)\frac{x_n - x_{n-1}}{F(x_n) - F(x_{n-1})}, \ x_0, x_1 \in [a, b]. \qquad [1.3]$$

Conceptually, we are replacing the tangent to the curve $F(x)$ at the point $(x_n, F(x_n))$ with the straight line that passes through this point and the point $(x_{n-1}, F(x_{n-1}))$.

EXAMPLE.– In 1225, Leonardo di Pisa studied the equation

$$f(x) = x^3 + 2x^2 + 10x - 20 = 0 \qquad [1.4]$$

and found the solution $x \simeq 1.368808107$. Nobody knows how he found his result, but it is an astonishing achievement given the state of knowledge at the time.

There are several ways in which to rewrite equation [1.4] in the form $x = F(x)$. If we define

$$x = \frac{20}{x^2 + 2x + 10} = F(x),$$

then

$$\lim_{x \to +\infty} f(x) = +\infty \qquad \lim_{x \to -\infty} f(x) = -\infty.$$

By the intermediate value theorem, there exists at least one root on $]-\infty,+\infty[$. Computing the derivative of the function f gives

$$\begin{aligned} f'(x) &= 3x^2+4x+10 \\ &= (\sqrt{3}x+\frac{2}{\sqrt{3}})^2+\frac{26}{3}. \end{aligned}$$

Therefore, $f'(x) > 0$ for every $x \in \mathbb{R}$, so the equation $f(x) = 0$ has a unique root in $\mathbb{R}$. More precisely, $f(1) = -7$ and $f(2) = 16$ so the unique root of f on $\mathbb{R}$ is contained in the interval $[1, 2]$.

Hence, to find this root, we can simply restrict attention to the interval $[1, 2]$. The solution satisfies

$$x = \frac{20}{x^2+2x+10} = F(x) = \frac{20}{(x+1)^2+3^2}.$$

It can easily be shown that the image of $[1, 2]$ under F is $[1, 2]$. The derivative of F is

$$F'(x) = \frac{-40(x+1)}{((x+1)^2+9)^2},$$

so, for all $x \geq 0$,

$$|F'(x)| = \frac{40(x+1)}{((x+1)^2+9)^2} \leq K < 1.$$

We define

$$g(x) = \frac{40(x+1)}{((x+1)^2+9)^2}.$$

After performing a few calculations, we find that

$$g'(x) = \frac{120((x+1)^2+9)(3-(x+1)^2)}{((x+1)^2+9)^4}.$$

The following table shows the variations of g:

x	0		$\sqrt{3}-1$		$+\infty$
g'		$+$	0	$-$	
g	$\frac{2}{5}$	$\nearrow$	$\frac{5\sqrt{3}}{18}$	$\searrow$	0

On $[1,2]$, the function g is decreasing, so, $\forall x \in [1,2]$,

$$g(x) \leq g(1) \simeq 0.47 = K < 1.$$

Therefore,

$$|F'(x)| \leq 0.48 \simeq K < 1 \quad K = \frac{80}{169},$$

which means that F is a contraction mapping. This implies that the fixed-point algorithm converges, and $\lim_{n\to+\infty} x_n = \alpha$ for $x_0 = 1$, where $\alpha = F(\alpha)$.

Note that:

$$|\alpha - x_{24}| = |F(\alpha) - F(x_{23}|.$$

Now, let

$$e_{24} = \alpha - x_{24} = F(\alpha) - F(x_{23}) = F'(\alpha - x_{23}).$$

Then:

$$|e_{24}| \leq \frac{80}{169}|e_{23}| \leq \cdots \leq \left(\frac{80}{169}\right)^{24} |\alpha - x_0| < \left(\frac{80}{169}\right)^{24}.$$

To finish, observe that

$$\left(\frac{80}{169}\right)^{24} \simeq 1.6 \times 10^{-8}.$$

We can now apply the fixed-point method to the function $f(x) = x^3 + 2x^2 + 10x - 20$.

We have the values $y_0 = 1$, $c = 2$, $f(\frac{3}{2}) = \frac{23}{8}$ and $f(2) = 16$, so

$$y_{n+1} = y_n - \frac{f(y_n)}{f(y_n) - f(c)}(y_n - c).$$

Setting

$$\begin{aligned} G(y) &= y - \frac{f(y)}{f(y) - f(c)}(y - c) \\ &= \frac{-yf(c) + cf(y)}{f(y) - f(c)} \end{aligned}$$

gives

$$G'(y) = \frac{[-f(c) + cf'(y)][f(y) - f(c)] - [-yf(c) + cf(y)]f'(y)}{(f(y) - f(c))^2}$$
$$= \frac{-f(c)f(y) + f(c)yf'(y) - cf(c)f'(y) + f(-c)^2}{(f(y) - f(c))^2}.$$

Replacing $f(y)$ with its formula, we find

$$G'(y) = \frac{32(y^3 - 2y^2 - 4y + 8)}{(f(y) - 16)^2}.$$

It can be shown that $|G'(y)| \leq K < 1$ in the neighborhood of α. This implies that

$$|G'(y)| \leq \frac{32|y^3 - y^2 - 4y + 8|}{(f(\frac{3}{2} - 16)^2} < \frac{32.3^3}{13.125^2} \simeq 0.557.$$

The following computations are performed with $c = \frac{3}{2}$. In this case,

$$y_1 = G(y_0) = 1 + \frac{28}{79} = \frac{107}{79} \simeq 1.354430797$$

and $f(y_1) \simeq -0.302055212$;

$$y_2 = G(y_1) = y_1 - \frac{f(y_1)}{f(y_1) - \frac{23}{8}}(\frac{107}{79} - \frac{3}{2})$$
$$\simeq 1.36827067688$$

and $f(y_2) \simeq -0.011685720$;

$$y_3 = G(y_2) = y_2 - \frac{f(y_2)}{f(y_2) - \frac{23}{8}}(y_2 - \frac{3}{2})$$
$$\simeq 1.36878803936$$

and $f(y_3) \simeq -0.000088007$;

$$y_4 = G(y_3) = y_3 - \frac{f(y_3)}{f(y_3) - \frac{23}{8}}(y_3 - \frac{3}{2})$$
$$\simeq 1.3688079520.$$

Therefore,

$$\begin{aligned}|\alpha - y_n| &\leq |\alpha - x_{24}| + |x_{24} - y_4| \\ &\leq 1.6 \times 10^{-8} + 2.5 \times 10^{-7} \\ &\leq 2.2 \times 10^{-7}.\end{aligned}$$

With the secant method, an order of $n = 4$ leads to an accuracy of the order of 10^{-7}, which is much better than the fixed-point method, which needed an order of $n = 24$ to achieve an accuracy of the order of 10^{-8}. Thus, the secant method converges more quickly than the fixed-point method.

Newton's method can be stated as

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

where $f(x) = x^3 + 2x^2 + 10x - 20$ and $f'(x) = 3x^2 + 4x + 10$, which gives

$$x_{n+1} = x_n - \frac{x_n^3 + 2x_n^2 + 10x_n - 20}{3x_n^2 + 4x_n + 10}.$$

Choosing $x_0 = 1$, we find:

$$x_1 \simeq 1.411764706$$
$$x_2 \simeq 1.369336471$$
$$x_3 \simeq 1.368808189$$
$$x_4 \simeq 1.368898108.$$

In this example, we almost achieved the same accuracy as Leonardo in just four iterations, with an error of 10^{-8}.

Newton's method is significantly better than the secant method, but in a sense they are similar. We can show that the error of Newton's method decays quadratically.

From the second-order Taylor expansion of f about α, we can write that:

$$f(\alpha) = f(x_{n-1}) + (\alpha - x_{n-1})f'(x_{n-1}) + \frac{1}{2}(\alpha - x_{n-1})^2 f''(\xi).$$

Newton's formula gives

$$0 = f(x_{n-1}) + (x_n - x_{n-1})f'(x_{n-1})$$

which implies that

$$f(\alpha) = 0 = (\alpha - x_n)f'(x_{n-1}) + \frac{1}{2}(\alpha - x_{n-1})^2 f''(\xi).$$

By setting $e_n = \alpha - x_n$, we find that

$$0 = e_n f'(x_{n-1}) + \frac{1}{2}e_{n-1}^2 f''(\xi),$$

hence

$$|e_n| \leq \frac{1}{2}\frac{M}{m}|e_{n-1}|^2,$$

where $M = \sup_{[1,\frac{3}{2}]} |f''(x)|$ and $m = \inf_{[1,\frac{3}{2}]} |f'(x)|$. We deduce that

$$|e_n| \leq \left(\frac{M}{2m}\right)^{2^n-1} |e_0|^{2^n}.$$

1.3.8. Regula falsi *method (or false position method)*

If we approximate $F'(x)$ by the expression:

$$\frac{F(x_n) - F(x_0)}{x_n - x_0}, \qquad [1.5]$$

then the recurrence formula becomes

$$x_{n+1} = x_n - F(x_n)\frac{x_n - x_0}{F(x_n) - F(x_0)}, \ x_0, x_1 \in [a, b]. \qquad [1.6]$$

This time, we are replacing the tangent to the curve of $F(x)$ at the point $(x_n, F(x_n))$ with the straight line that passes through this point and $(x_0, F(x_0))$.

EXAMPLE.– Consider the function

$$f(x) = e^{-2x} - \cos(x) - 3.$$

The function f is decreasing on $[-1, 0]$, $f(-1) = 3.848754$ and $f(0) = -3$, hence f has a unique zero in $[-1, 0]$.

By applying the *regula falsi* method to f on $[-1, 0]$, we find:

i	x_i	i	x_i
1	-0.438036	8	-0.6656762
2	-0.595945	9	-0.665706
3	-0.645201	10	-0.665714
4	-0.659764	11	-0.665717
5	-0.663996	12	-0.665717
6	-0.665221	13	-0.665718
7	-0.665574	14	-0.665718

An approximate value for the root of f is given by $x_{13} = -0.665718$.

By applying Newton's method to f with initial point $x_0 = 0$, we find that

i	x_i
1	-1.500000
2	-1.086704
3	-0.798386
4	-0.681373
5	-0.665953
6	-0.665718

This also leads to the approximation $x_6 = -0.665718$.

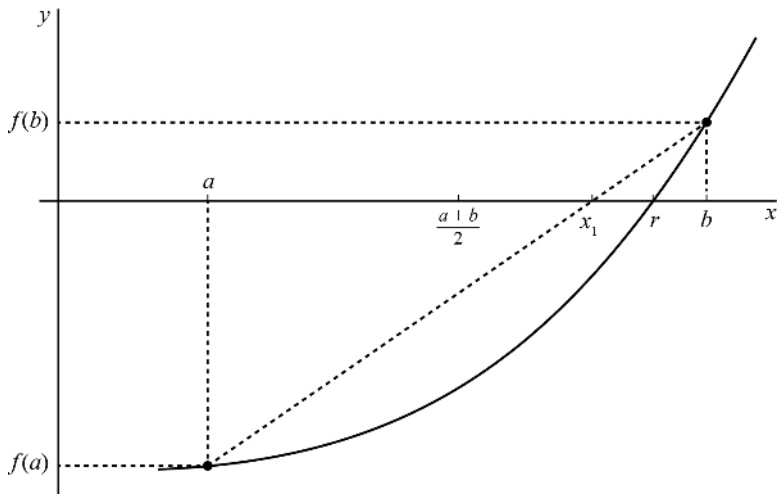

Figure 1.3. Regula falsi *method*

1.4. Order of an iterative process

DEFINITION.– *Let $\bar{x}$ be a root of the equation $F(x) = 0$, and let (x_n) be a numerically constructed sequence that converges to $\bar{x}$.*

We say that the method associated with this sequence is of order k if the following condition holds:

$$x_{n+1} - \bar{x} = (x_n - \bar{x})^k [a_k + \epsilon(x_n)],$$

where a_k is a non-zero constant and $\epsilon(x)$ is a function that satisfies $\lim \epsilon(x) = 0$ (i.e. $\lim_{n \to +\infty} \epsilon(x_n) = 0$). In other words:

$$\lim_{n \to =\infty} \frac{x_{n+1} - \overline{x}}{(x_n - \overline{x})^k} = a_k.$$

REMARK.–

1) In practice, the sequence (x_n) is not explicitly known. We say that the method is of at least order k if we can find a constant $M > 0$ and a given rank n_0, such that, for $n \geq n_0$, $|x_{n+1} - \bar{x}| \leq M|x_n - \bar{x}|^k$.

If so, an upper bound for the error of the method is given by:

$$|x_n - \bar{x}| \leq \frac{1}{\sqrt[k-1]{M}} \left(\sqrt[k-1]{M} |x_0 - \bar{x}| \right)^{k^n}.$$

2) If the iterative method is defined using a function with continuity class $\mathcal{C}^p$, the order k can be defined as the smallest integer m, such that $f^{(m)}(\bar{x}) \neq 0$ and $a_k = \dfrac{f^{(k)}(\bar{x})}{k!}$ (or alternatively, $M = \sup \left| \dfrac{f^{(k)}(x)}{k!} \right|$).

3) Higher order iterative methods converge faster.

1.5. Using *Matlab*

1.5.1. *Finding the roots of polynomials*

Suppose that we wish to compute the roots of the equation:

$$x^5 - 2x^4 + 2x^3 + 3x^2 + x + 4 = 0.$$

We can use the *roots* function to do this in *Matlab*. First, we enter the coefficients of the polynomial:

```
>> c= [1 -2 2 3 1 4];
```

Next, we can find the solution by running the following command:

```
>> solution = roots(c)

solution =
         1.5336 + 1.4377i
         1.5336 - 1.4377i
        -1.0638
        -0.0017 + 0.9225i
        -0.0017 - 0.9225i
```

Given an array of roots, the function *poly* returns the coefficients of the polynomial with these roots ordered by decreasing powers. Thus, in this example, we can recover the original polynomial as follows:

```
>> poly(solution)

ans =
    1.0000 -2.0000 2.0000 3.0000 1.0000 4.0000
```

Matlab allows us to find the zero of a function using the *fzero* command. To do this, we need to create a file (e.g. 'f.m') containing the function. In this example, we shall choose $f(x) = \exp(x) - 2\cos(x)$.

```
function f=f(x)
f=exp(x)-2*cos(x)
```

To find the solution of $f(x) = 0$ in the neighborhood of the point $x = 0.5$, we run the command 'fzero('f',0.5)'. This function displays the values of $f(x)$ computed at each iteration until the desired solution x^* is found.

```
>>d=fzero('f',0.5)

f =
    -0.1064
    -0.1430
    -0.0692
    -0.1579
```

```
    -0.0536
    -0.1788
    -0.0313
    -0.2080
     5.8950e-004
    -3.0774e-005
    -4.1340e-009
     2.2204e-016
    -8.8818e-016

d =
    0.5398
```

1.5.2. *Bisection method*

To implement the bisection method, we can create a script called 'bisection.m', with the following code:

```
a=0;
fa=-5;
b=3;
fb=16;
eps=1.0e-3;
while (b-a) >eps
  x=(a+b)/2;
  fx=x^3-2*x-5;
  if (sign(fx) == sign(fa))
     a=x;
     fa=fx;
  else
     b=x;
     fb=fx;
  end
end;
x
```

We can run the script 'bisection.m' as follows to compute the root of the function $f(x) = x^3 - 2x - 5$ contained in the interval $[0, 3]$:

```
>> bisection

x=
   2.0940
```

1.5.3. *Newton's method*

The following script, saved under the name 'newton.m', implements Newton's method:

```
format
short;
clc;
syms x;

display ('Newton''s method is an iterative method for solving
             nonlinear systems');
f = input('Enter the function f(x)=');

x0 = input('Enter the initial estimate of the solution: ');

n = input('Enter the maximum number of iterations: ');

e = input('Enter the margin of error: ');
i = 0;
while (i<n)
    df = diff(f,x);
    dfx0 = subs(df,x,x0);
    fx0 = subs(f,x,x0);
    x1 = x0 - (fx0/dfx0);
    dx = x1 - x0;
   if abs(dx)/abs(x0)<=e
       fprintf('The solution is = %i\n',x1);
       fprintf('The number of iterations required was = %i\n',i);
       break
   end
    i = i+1;
    x0=x1;
end
if abs(dx)/abs(x0) >e
    fprintf('Convergence was not achieved in n iterations ');
end
```

We can apply this script to compute the root of the equation $x^3 - 3x + 1 = 0$. Executing the script 'newton.m' returns the following output:

```
Newton's method is an iterative method for solving nonlinear systems
Enter the function f(x)=x^3-3*x+1
```

```
Enter the initial estimate of the solution: 0

Enter the maximum number of iterations: 100

Enter the margin of error: 0.001

The solution is = 3.472964e-001

The number of iterations required was = 2
```

Therefore, the solution computed by this script is $x = 0.3473$.