People's Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research Academic
University of Bejaïa – A/MIRA

Faculty of Technology
Department of Electrical Engineering

Course in

Numerical Methods

Intended for second-year LMD Bachelor's degree students in Electrical Engineering (Electromechanics (ELM) and Electrotechnics (ELT))

Associate Professor (MCA:

Mrs. Leila YOUNSI née ABBACI.

ACADEMIC YEAR 2025/2026

CONTENTS

LIST OF FIGURES

PRÉFACE

This course booklet on Numerical Methods has been prepared for second-year LMD Electrical Engineering students in Electromechanics (ELM) and Electrotechnics (ELT) within the framework of Semester 4, in accordance with the 2021/2022 curriculum. It serves as an educational support designed to complement in-person lectures and to facilitate the understanding of fundamental concepts in numerical methods applied to scientific and engineering problems. The module framework is as follows: Semester 4, Teaching Unit UEF 2.2.2, Subject: Numerical Methods, Contact Hours: 45 h (Lectures: 1 h 30, Tutorials: 1 h 30), Credits: 4, Coefficient: 2.

The main objective of this booklet is to present, in a clear and progressive manner, various numerical techniques, namely: Methods for Solving Nonlinear Equations, Numerical Integration, Polynomial Interpolation, Numerical Solution of Ordinary Differential Equations, Direct Methods for Solving Systems of Linear Equations, and Approximate (Iterative) Methods for Solving Systems of Linear Equations. Each chapter includes practical examples and exercises to help students consolidate their knowledge and develop autonomy in applying the methods studied.

This booklet does not replace lectures but serves as a useful reference tool for preparing tutorials and examinations. Students are encouraged to supplement their reading with specialized textbooks and additional educational resources.

CHAPTER 1

METHODS FOR SOLVING NONLINEAR EQUATIONS

In this chapter, we focus on the approximation of the zeros of a real-valued function of a real variable. In other words, we aim to numerically solve the following problem:

$$(\text{Pr}) \begin{cases} \text{Let } f \text{ be a real-valued function of a real variable,} \\ \text{Find } \alpha \in \mathbb{R} \text{ such that } f(\alpha) = 0. \end{cases}$$

We will study three numerical methods for solving nonlinear equations in one variable, also known as transcendental equations. Examples of such equations include $f(x) = \sin(x) + x = 0$ and $f(x) = \ln(x) - 2x + 3 = 0$. These equations generally do not have exact solutions that can be obtained analytically. Therefore, numerical methods are used to compute approximate solutions. The accuracy of these solutions can be improved as needed, especially with the help of computational tools.

These numerical methods generally compute only one root within a carefully chosen interval. Hence, if the equation has multiple roots, it is necessary to first locate them within appropriate intervals, and then compute each root separately.

All the methods for solving problem (Pr) that we will present are *iterative methods*, meaning they are based on the construction of sequences $(x_n)_{n\geq 0}$ defined by a recurrence relation of the form:

$$x_{n+1} = F(x_n)$$

Under certain conditions, these sequences are expected to converge to the desired solution α, that is:

$$\lim_{n\to+\infty} x_n = \alpha.$$

With the exception of a few rare cases of simple equations (such as a quadratic equation: $ax^2 + bx + c = 0$), nonlinear equations are particularly difficult to solve analytically. For this reason, approximate solution techniques—called *numerical methods*—are employed.

In this chapter, we present three numerical methods for solving nonlinear equations of the form $f(x) = 0$, namely:

- the Bisection method (or Dichotomy),
- the Fixed-Point method (or Successive Approximations),
- and Newton's method.

1.1 Introduction to Numerical Errors and Approximations

Traditionally, mathematics introduces us to theories and methods that enable the analytical solving of various problems, such as integration and equation solving. These approaches typically yield exact results for a given problem. In contrast, numerical analysis often produces approximate solutions, as the methods employed depend on several parameters that influence the accuracy of the outcome. A central concern in numerical analysis is therefore to control the effects of the errors introduced, which generally arise from three main sources:

1.1.1 Absolute and Relative Errors

Exact numbers $\begin{cases} in & \mathbb{N} & 1, 3, 6, 7; \\ in & \mathbb{Q} & \frac{1}{6}, \frac{3}{7}, \frac{11}{19}; \\ in & \mathbb{R} & \sqrt{8}, \pi, e; \end{cases}$

Let x be an exact number and $x*$ an approximate value of x, we write:

$x \backsimeq x*$ or $x \approx x*$

- If $x* > x$, $x*$ is called an overestimate.
- If $x* < x$, $x*$ is called an underestimate.

Examples: $\frac{2}{3} \approx 0.6666$, $\pi \approx 3.14$, $\sqrt{5} \approx 2.23$ are underestimates, but $e \approx 2.72$ is an overestimate.

1.1.1.1 Absolute Error

Definition 1.1. The **absolute error** of the approximate value $x*$ of a number x is the positive real quantity, denoted by $\triangle(x)$, defined as:

$$\triangle(x) = |x - x*|.$$

Comment: The smaller the absolute error, the more accurate the approximation $x*$ is.

Example 1.1. For the exact value $x = \frac{2}{3}$, the approximate value $x_1^* = 0.666667$ is 1000 times more accurate than the approximation $x_2^* = 0.667$. Indeed, we have:

$$\triangle_1(x) = |x - x_1^*| = |\frac{2}{3} - 0.666667| = \frac{1}{3}10^{-6}.$$
$$\triangle_2(x) = |x - x_3^*| = |\frac{2}{3} - 0.667| = \frac{1}{3}10^{-3}.$$

1.1.1.2 Relative error

Definition 1.2. The *relative error* of the approximate number x^* of the real positive quantity x, denoted $r(x)$, is defined by:

$$r(x) = \frac{|x - x^*|}{|x|} = \frac{\Delta(x)}{|x|}$$

Comment: The relative error is often expressed as a percentage (relative precision) by:

$$r\% = r(x) \times 100$$

Example 1.2. For exact values $x = \frac{2}{3}$ and $y = \frac{1}{15}$, we consider the approximate values $x^* = 0.67$ and $y^* = 0.07$, respectively.
The corresponding absolute errors are:

$$\Delta(x) = |x - x^*| = \left|\frac{2}{3} - 0.67\right| = \frac{1}{3} \cdot 10^{-2}, \quad \Delta(y) = |y - y^*| = \left|\frac{1}{15} - 0.07\right| = \frac{1}{3} \cdot 10^{-2}$$

The corresponding relative errors are:

$$r(x) = \frac{|x - x^*|}{|x|} = 0.5 \cdot 10^{-2} = 0.5\%, \quad r(y) = \frac{|y - y^*|}{|y|} = 5 \cdot 10^{-2} = 5\%$$

Thus, although the absolute errors are equal, x^* is a 10 times more precise approximation for x than y^* is for y.

1.1.1.3 Upper Bounds of Absolute and Relative Errors

If the exact value is known, one can determine the absolute and relative errors. But in most cases, it is not. The absolute and relative errors then become unknown, and to estimate them, we introduce the notion of *upper bound* of the absolute and relative error.

Definition 1.3. We call an *upper bound of the absolute error* of an approximate value x^* of a positive real number x, any positive real number Δx such that:

$$\Delta(x) = |x - x^*| \leq \Delta x$$

or equivalently:

$$x^* - \Delta x \leq x \leq x^* + \Delta x$$

and we write:

$$x = x^* \pm \Delta x \quad \text{which means:} \quad x \in [x^* - \Delta x, x^* + \Delta x]$$

Remark 1.1.

1. *The smaller Δx is, the more accurate the approximation x^* is. Therefore, in practice, one chooses the smallest possible Δx.*

2. *Since $x \simeq x^*$, in practice we take $r_x \simeq \frac{\Delta x}{|x^*|}$, which is an upper bound of the relative error of x^*, and we write:*

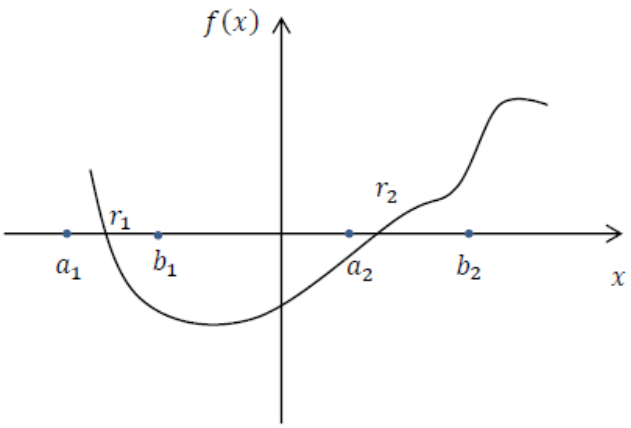
$$x = x^* \pm |x^*| r_x$$

3. *When the absolute (or relative) error is not known, the effective error $\Delta x(r_x)$ is abusively referred to as the absolute (or relative) error of x^*.*

1.1.2 Separation of the roots of $f(x) = 0$

Let $f(x) : \mathbb{R} \longrightarrow \mathbb{R}$ be a given function. We are looking for one or more solutions of the function $f(x) = 0$. In the rest of this chapter, we assume that $f(x) = 0$ is a continuous function and that x^* is the unique solution of $f(x) = 0$ in the interval $[a, b]$.

The separation of roots of $f(x) = 0$ involves determining the different intervals, each containing a single root. To do this, we use one of the following two methods: - The graphical method: We plot the curve of $f(x)$ and locate (separate) the interval $[a, b]$ where the solution lies - The algebraic method: Separation of roots of $f(x) = 0$ using the intermediate value theorem.



Separation of roots (r_1 and r_2) of a function $f(x) = 0$: $r_1 \in [a_1, b_1]$ and $r_1 \in [a_2, b_1]$.

For all numerical methods used to solve (Pr), one must first perform a very important step, which consists in finding intervals of the form $[a_i, b_i]$ in which the equation $f(x) = 0$ admits one and only one solution. This step is called the separation of zeros. To achieve this, we rely on the following theorem.

Theorem 1.1. *(Intermediate Value Theorem)*
Let f be a function defined and continuous on the interval $[a, b]$.

- *If $f(a) \cdot f(b) < 0$, then there exists at least one real number $\alpha \in [a, b]$ such that $f(\alpha) = 0$.*
- *Furthermore, if $f(x)$ is monotonic on the interval $[a, b]$(if $f'(x) \neq 0$ for all $x \in [a, b]$), then the solution is unique.*

Remark 1.2. *If $f(a) \cdot f(b) > 0$, then either f has no zeros in $[a, b]$, or it has an even number of zeros in that interval.*

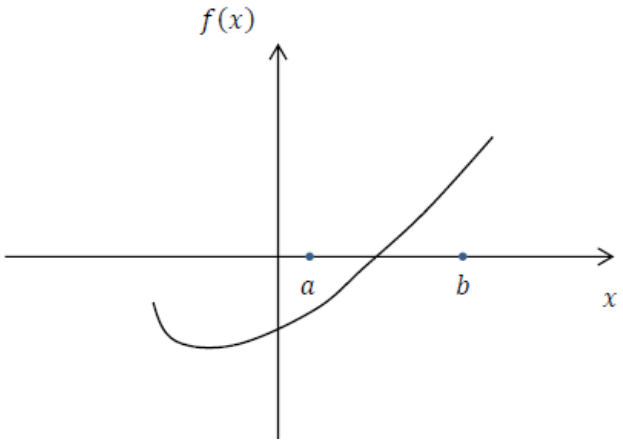


Figure 1.1: A function f(x) is monotonic on the interval $[a, b]$.

1.2 Bisection Method (or Dichotomy Method)

The bisection method is based on the Intermediate Value Theorem. It is used to determine a root of a function $f(x)$ by iteratively eliminating subintervals where the function does not change sign. The core idea of the bisection method also known as the half-interval method is to divide the interval in half at each iteration by selecting its midpoint. Then, using the Intermediate Value Theorem, we identify which of the two subintervals contains a sign change, and hence the root. Specifically, we retain the subinterval where the function values at the endpoints have opposite signs. The algorithm terminates when the length of the interval becomes smaller than a prescribed precision ϵ.
The method is illustrated in Figure 4.1 and proceeds as follows. We start with an interval $[a, b]$ such that $f(a)f(b) < 0$, and set $a_0 = a$, $b_0 = b$. Let $c_1 = (a_0 + b_0)/2$ be the mid-point of the interval $[a_0, b_0]$. Since $f(a_0) > 0$, $f(b_0) < 0$, and $f(c_1) < 0$, out of the two subintervals, $[a_0, c_1]$ and $[c_1, b_0]$, the first one is guaranteed to contain a root of f. Hence,

we set $a_1 = a_0$, $b_1 = c_1$, and continue the search with the interval $[a_1, b_1]$. For the mid-point c_2 of this interval we have $f(c_2) < 0$, therefore there must be a root of f between a_1 and c_2, and we continue with $[a_2, b_2] = [a_1, c_2]$. After one more step, we obtain $[a_3, b_3] = [c_3, b_2]$ as the new search interval. If $b_3 - a_3 < \epsilon$, then $c_4 = (a_3 + b_3)/2$ is output as an approximation of the root x^*.

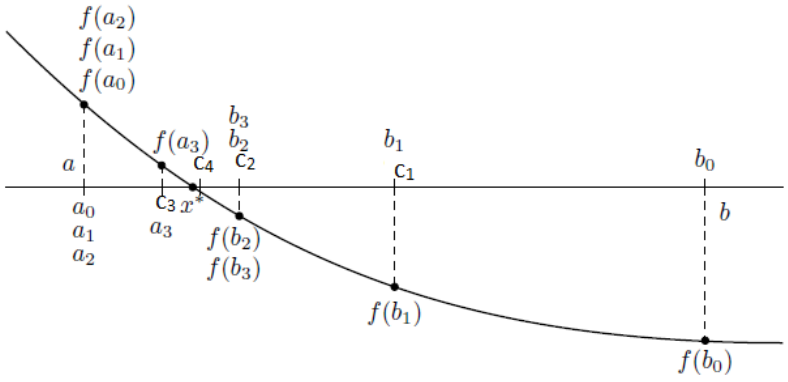


Figure 1.2: An illustration of the bisection method.

1.2.1 Convergence conditions

The bisection method is applicable and guaranteed to converge only when the following three conditions are satisfied.

- $f(x)$ is continuous on the interval $[a, b]$.
- The solution α lies in the open interval $[a, b]$, i.e., $\alpha \in [a, b]$.
- The function takes opposite signs at the endpoints: $f(a) \cdot f(b) < 0$.

The bisection algorithm produces a set of centers $c_1, ..., c_k, ...$ such that $\lim_{k\to\infty} c_k = x^*$, where $x^* \in [a_0, b_0]$ is a root of f. Since at iteration n of the bisection method the root x^* is in the interval $[a_n, b_n]$ and c_n is the mid-point of this interval, we have $|x^* - c_n| \leq \frac{1}{2}(b_n - a_n)$.

Also,

$$b_n - a_n = \frac{1}{2}(b_{n-1} - a_{n-1}) = \frac{1}{2^2}(b_{n-2} - a_{n-2}) = \cdots = \frac{1}{2^n}(b_0 - a_0).$$

Hence,

$$|x^* - c_n| \leq \frac{1}{2^{n+1}}(b_0 - a_0), \tag{1.1}$$

and the bisection method converges. Moreover, (1.1) can be used to determine the number of iterations required to achieve a given precision ϵ by solving the inequality:

$$\frac{1}{2^n}(b_0 - a_0) < \epsilon,$$

which is equivalent to

$$n > \log_2\left(\frac{b_0 - a_0}{\epsilon}\right) = \frac{\ln\left(\frac{b_0 - a_0}{\epsilon}\right)}{\ln 2}. \tag{4.12}$$

For example, if $[a_0, b_0] = [0, 1]$ and $\epsilon = 10^{-5}$, we have:

$$n > \frac{5\ln(10)}{\ln 2} \approx 16.6,$$

and the number of iterations guaranteeing the precision ϵ is 17.

1.2.2 Algorithm

Let f be a continuous function on the interval $[a, b]$, such that $f(a)f(b) < 0$. The objective is to find an approximate root of f within a given tolerance $\epsilon > 0$.

Step 1: Initialization. Set the tolerance $\epsilon > 0$, and initialize: $a_0 = a, \quad b_0 = b, \quad k = 1$.

Step 2: Compute the midpoint. $x_k = \frac{a_k + b_k}{2}$.

Step 3: Check stopping condition. If $|b_k - a_k| < \epsilon$, stop the algorithm. The approximate solution is $x^* = c_k$.

Step 4: Determine the new interval.

- If $f(a_k)f(x_k) < 0$, then set: $a_{k+1} = a_k$,$b_{k+1} = c_k$.
- Otherwise, set: $a_{k+1} = c_k$,$b_{k+1} = b_k$.

Step 5: Increment. Set $k = k + 1$, and return to Step 2.

1.3 Example: Solving with the Bisection Method

We aim to solve the following equation using the bisection method $f(x) = x^3 + x^2 - 3x - 3$ on the interval $[1,\ 2]$.

We compute $f(1) = 1^3 + 1^2 - 3 \times 1 - 3 = -4, \quad f(2) = 8 + 4 - 6 - 3 = 3$

Therefore,$f(1) \cdot f(2) = -4 \times 3 = -12 < 0$, which confirms that a root exists in the interval $[1,\ 2]$.

First iteration (k = 1):

$$c_1 = \frac{1+2}{2} = 1.5$$

$f(1.5) = (1.5)^3 + (1.5)^2 - 3 \times 1.5 - 3 = -1.875$ Since $f(1.5) < 0$ and $f(2) > 0$, the sign changes in the interval $[1.5,\ 2]$. We take this as the new working interval.

Second iteration (k = 2):

$$c_2 = \frac{1.5+2}{2} = 1.75$$

$f(1.75) = (1.75)^3 + (1.75)^2 - 3 \times 1.75 - 3 = 0.171875$

Now, since $f(1.75) > 0$ and $f(1.5) < 0$, the sign changes in $[1.5,\ 1.75]$.

Third iteration (k = 3):

$$c_3 = \frac{1.5+1.75}{2} = 1.625$$

$f(1.625) = (1.625)^3 + (1.625)^2 - 3 \times 1.625 - 3 \approx -0.863$

So, the new interval is $[1.625,\ 1.75]$. If we want an absolute error smaller than 0.5×10^{-2}, we need to perform at least: $n > \frac{\log(x_2 - x_1) - \log(\varepsilon)}{\log(2)} = \frac{\log(2) - \log(0.5 \times 10^{-2})}{\log(2)} - 1 \approx 7.643856$. So we must perform at least 8 iterations to guarantee this level of precision.

The following table summarizes the results:

n	x_1	c_k	x_2	$f(x_1)$	$f(c_k)$	$f(x_2)$	$\lvert x_{c(n)} - x_{c(n-1)} \rvert$
1	1.000	1.500000	2.000	-4.000000	-1.875000	3.000000	–
2	1.500	1.750000	2.000	-1.875000	0.171875	3.000000	0.250000
3	1.500	1.625000	1.750	-1.875000	-0.943359	0.171875	0.125000
4	1.625	1.687500	1.750	-0.943359	-0.409423	0.171875	0.062500
5	1.6875	1.718750	1.750	-0.409423	-0.124786	0.171875	0.031250
6	1.71875	1.734375	1.750	-0.124786	0.022029	0.171875	0.015625
7	1.71875	1.726563	1.734375	-0.124786	-0.051755	0.022029	0.007812
8	1.726563	1.730469	1.734375	-0.051755	-0.049572	0.022029	0.003906

An illustration of the bisection method.

1.4 Fixed Point Method

In this section, we discuss the fixed point method, which is also known as simple one-point iteration or method of successive substitutions. We consider an equation in the form

$$f(x) = 0. \tag{1.2}$$

where $f : [a, b] \longrightarrow \mathbb{R}$, and suppose that the interval $[a, b]$ is known to contain a root of this equation. We can always reduce (1.2) to an equation in the form

$$x = g(x). \tag{1.3}$$

Theorem 1.2. *Let $[a, b]$ be a non-empty interval of $\mathbb{R}$, and let g be a mapping from $[a, b]$ into itself.*
*If g is continuous, then there exists $\xi \in [a, b]$ such that $g(\xi) = \xi$; this is called a **fixed point** of g.*

Consequently, under certain conditions, approximating the zeros of f is equivalent to approximating the fixed points of g. A common method for determining fixed points consists in constructing a sequence $\{x_n\}_{n\geq 0}$ using the following iterative process:

$$\begin{cases} x_0 : \text{given initial approximation} \\ x_{n+1} = g(x_n) \end{cases}$$

Remark 1.3. *It should be noted that there are infinitely many ways to choose the function g. However, the choice of g must ensure that the sequence constructed by the fixed-point method converges.*

1.5 Convergence Criterion for the Fixed-Point Method

Theorem 1.3. *Let g be a function of class $\mathcal{C}^1$ on the interval $[a, b]$. Let $x_0 \in [a, b]$ and define the sequence by $x_{k+1} = g(x_k), \quad k \geq 0$. If the following two conditions are satisfied:*

First condition ***Contraction condition:*** *There exists a constant $M \in [0, 1[$ such that $|g'(x)| \leq M < 1, \quad \forall x \in [a, b]$, with $M = \max\limits_{x\in[a,b]} |g'(x)|$.*

Second condition ***Stability condition:*** *$g(x) \in [a, b]$ for all $x \in [a, b]$.*

Then, the function g has a unique fixed point $\tilde{x}$ in $[a, b]$, and the sequence $\{x_k\}_{k\geq 0}$ defined by $x_{k+1} = g(x_k)$ converges to $\tilde{x}$ for all $x_0 \in [a, b]$.

Moreover, the following formula provides an upper bound for the error at iteration n:

$$|x_k - \tilde{x}| \leq \frac{M^k}{1 - M}|x_1 - x_0|.$$

Proof. Let (x_n) be a sequence defined by:

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

and let α be the fixed point. If g is fonction continuous with constant $M \in [0, 1]$, then the sequence converges to α, for all $x_0 \in [a, b]$, and the following error estimate holds:

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

where M is the $M = \max_{x\in[a,b]} |g'(x)|$.

1. Existence:

We have:

$$|x_{n+1} - x_n| \leq M^n |x_1 - x_0|, \quad \forall n \in \mathbb{N}$$

Since $x_{i+1} - x_i = g(x_i) - g(x_{i-1})$ and $M = \max_{x\in[a,b]} |g'(x)|$, then:

$$|x_{n+1} - x_n| \leq M|x_n - x_{n-1}| \leq M^2|x_{n-1} - x_{n-2}| \leq \cdots \leq M^n|x_1 - x_0|$$

Therefore, for any $n, p \in \mathbb{N}$:

$$\begin{aligned}
|x_{n+p} - x_n| &\leq |x_{n+p} - x_{n+p-1}| + \cdots + |x_{n+1} - x_n| \\
&\leq (M^{n+1} + \cdots + M^{n+p})|x_1 - x_0| \\
&= M^n(M + M^2 + \cdots + M^p)|x_1 - x_0| \\
&= \frac{M^n(1 - M^p)}{1 - M}|x_1 - x_0| \\
&\leq \frac{M^n}{1 - M}|x_1 - x_0| \longrightarrow 0 \quad \text{as } n \to +\infty (because :0 < M < 1)
\end{aligned}$$

So the sequence (x_n) is a Cauchy sequence in $\mathbb{R}$, hence convergent (since $\mathbb{R}$ is complete). Let α be the limit of the sequence. By continuity of g, we get $g(\alpha) = \alpha$, hence α is a fixed point.

By fixing n and letting $p \to \infty$, we get:

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

2. Uniqueness:

Let α_1 and α_2 be two fixed points of g. Then:

$$|\alpha_1 - \alpha_2| = |g(\alpha_1) - g(\alpha_2)| \leq M|\alpha_1 - \alpha_2|$$

and since $M < 1$, it implies that $\alpha_1 = \alpha_2$.

□

1.5.1 Geometric interpretation

Finding α such that $g(\alpha) = \alpha$ amounts to finding the intersection of the graph of g (i.e., $y = g(x)$) with the line $y = x$, which is equivalent to finding the fixed point of g.

It is clearly observed that the slope of the curve $y = g(x)$ in the neighborhood of the root α is small, that is, $g'(x) \leq 1$, and the iteration converges to the unique solution α.

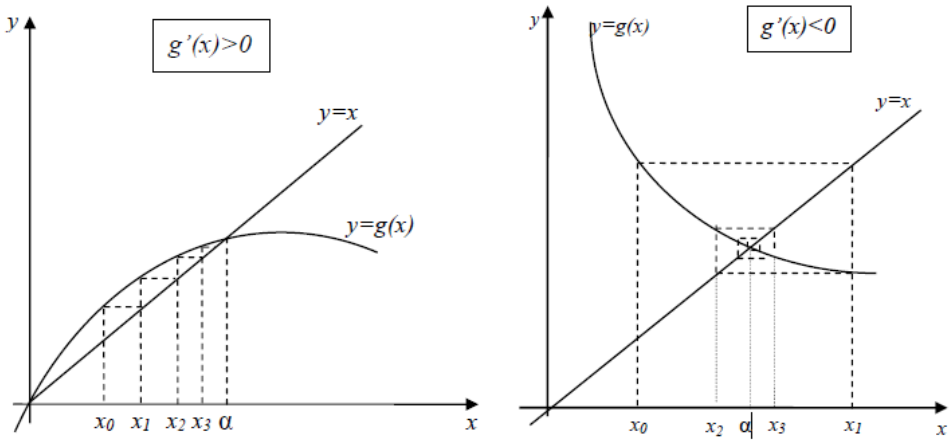


Figure 1.3: Geometric Interpretation of the Fixed Point Method.

1.5.2 Algorithm

- **Step 1.** With the defined approximation error, $x_0 \in [a, b]$, we set $k = 0$.
- **Step 2.** Compute $x_{k+1} = g(x_k)$.

  If $|x_{i+1} - x_i| < \varepsilon$, stop the iterations, the solution is $x^* = x_{i+1}$.

  Otherwise, set $k = k + 1$ and return to Step 2.

1.5.3 Termination criterion

The computations for the fixed-point method are stopped when the absolute difference between two successive iterations is less than a given tolerance ϵ:

$$|x_n - x_{n-1}| \leq \epsilon$$

The number of required iterations n is calculated using this relation:

$$\frac{M^n}{1-M}\,|x_1 - x_0| < \epsilon$$

Indeed,

$$\frac{M^n}{1-M}|x_1 - x_0| \leq \varepsilon \quad \Longleftrightarrow \quad M^n \leq \frac{1-M}{|x_1 - x_0|}\varepsilon$$

Hence, the number of iterations n required to achieve a given accuracy ε satisfies:

$$n \geq \frac{\ln\left(\dfrac{\varepsilon(1-M)}{|x_1 - x_0|}\right)}{\ln M}$$

Example 1.3. Consider the equation $f(x) = x^2 - 2 = 0$.

- Solving with the Fixed-Point Method.
- Find the approximate value of the root of this equation using the fixed-point method with an error of 10^{-4}.

Solution:

Choice 1

$$x^2 - 2 = 0 \Leftrightarrow x = \frac{2}{x} = g_1(x)$$

Verification of Condition 1: Is $g_1(x)$ a contraction on $[1, 1.5]$?

- $g_1(x)$ is continuously differentiable on $[1, 1.5]$ ..... satisfied
- $\max |g_1'(x)| = \max \left|\frac{-2}{x^2}\right| = 2 = M$, $M > 1, \forall x \in [1, 1.5]$ ..... not satisfied

Therefore, $g_1(x)$ is not a contraction on $[1, 1.5]$, and the first condition is not satisfied. Thus, with the choice of $g(x) = \frac{2}{x}$, the algorithm will never converge.

Choice 2

$$x^2 - 2 = 0 \Leftrightarrow (x-1)^2 + 2x - 3 = 0 \Rightarrow x = -\frac{1}{2}((x-1)^2 - 3) = g_2(x)$$

Verification of Condition 1: Is $g_2(x)$ a contraction on $[1, 1.5]$?

- $g_2(x)$ is continuously differentiable on $[1, 1.5]$ ..... satisfied
- $\max |g_2'(x)| = \max |x - 1| = \frac{1}{2} = M$, $M < 1, \forall x \in [1, 1.5]$ ..... satisfied

Then $g_2(x)$ is a contraction on $[1, 1.5]$, and the first condition is satisfied.

Verification of Condition 2: Is $g_2(x)$ stable within $[1, 1.5]$?

It can be observed that $\forall x \in [1, 1.5],\ g(x) \in [1, 1.5] \Rightarrow$ condition satisfied.

Both conditions are satisfied, so with the choice of $g(x) = -\frac{1}{2}((x-1)^2 - 3)$, the algorithm will converge after a few iterations.

$$x_{k+1} = g(x_k) = -\frac{1}{2}((x_k - 1)^2 - 3)$$

Setting $x_0 = 1.0000$:

$$
\begin{aligned}
k=0, &\quad x_1 = -\frac{1}{2}((x_0-1)^2-3) = 1.5000, &\quad |x_1-x_0| = 0.5000 > 0.0001\\
k=1, &\quad x_2 = -\frac{1}{2}((x_1-1)^2-3) = 1.3750, &\quad |x_2-x_1| = 0.1250 > 0.0001\\
k=2, &\quad x_3 = -\frac{1}{2}((x_2-1)^2-3) = 1.4297, &\quad |x_3-x_2| = 0.0547 > 0.0001\\
k=3, &\quad x_4 = -\frac{1}{2}((x_3-1)^2-3) = 1.4077, &\quad |x_4-x_3| = 0.0220 > 0.0001\\
k=4, &\quad x_5 = -\frac{1}{2}((x_4-1)^2-3) = 1.4169, &\quad |x_5-x_4| = 0.0092 > 0.0001\\
k=5, &\quad x_6 = -\frac{1}{2}((x_5-1)^2-3) = 1.4131, &\quad |x_6-x_5| = 0.0038 > 0.0001\\
k=6, &\quad x_7 = -\frac{1}{2}((x_6-1)^2-3) = 1.4147, &\quad |x_7-x_6| = 0.0016 > 0.0001\\
k=7, &\quad x_8 = -\frac{1}{2}((x_7-1)^2-3) = 1.4140, &\quad |x_8-x_7| = 0.0007 > 0.0001\\
k=8, &\quad x_9 = -\frac{1}{2}((x_8-1)^2-3) = 1.4143, &\quad |x_9-x_8| = 0.0003 > 0.0001\\
k=9, &\quad x_{10} = -\frac{1}{2}((x_9-1)^2-3) = 1.4142, &\quad |x_{10}-x_9| = 0.0001 > 0.0001\\
k=10, &\quad x_{11} = -\frac{1}{2}((x_{10}-1)^2-3) = 1.4142, &\quad |x_{11}-x_{10}| = 0.0000 < 0.0001
\end{aligned}
$$

Since $|x_{11} - x_{10}| = 0.0000 < \epsilon = 0.0001$, the solution is $x^* = x_{11} = 1.4142$.

1.6 Newton's method

This is the fastest method, but requires analytical computation of the derivative of $f(x)$.

1.6.1 Principle of the Method:

The principle of this method consists in approximating $f(x_{n-1})$ for $n = 1, 2, \ldots$ by the tangent line to the curve of f such that the intersection of this tangent with the x-axis is the point x_n. The condition is that f is differentiable and f' does not vanish in the considered interval $[a, b]$.

Given the approximation x_{n-1} ($n \geq 1$), the next approximation x_n of the root $\tilde{x}$ is obtained as follows:

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

Note that Newton's method is a particular case of a fixed-point method by setting

$$x_n = g(x_{n-1}) \quad \text{with} \quad g(x) = x - \frac{f(x)}{f'(x)}.$$

We can derive Newton's Method graphically, or by a Taylor series. We again want to construct a sequence $x_0, x_1, x_2, \ldots$ that converges to the root $x = r$. Consider the x_{n+1} member of this sequence, and perform a Taylor series expansion of $f(x_{n+1})$ about the point x_n. We have:

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

To determine x_{n+1}, we drop the higher-order terms in the Taylor series and assume $f(x_{n+1}) = 0$. Solving for x_{n+1}, we get:

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

Starting Newton's Method requires an initial guess x_0, hopefully close to the root $x = \alpha$.

1.6.2 Convergence Criterion of Newton's Method

We present the convergence criterion in the form of a theorem.

Theorem 1.4. *Let f be a C^2 class function on the interval $[a, b]$. Let $x_0 \in [a, b]$, and define the sequence by*

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}, \quad n \geq 0.$$

If the following three conditions are satisfied:

Condition 1) $f(a)f(b) < 0$

Condition 2) $f'(x) \neq 0$ *for all* $x \in [a, b]$ *(this implies that f is strictly monotonic)*

Condition 3) $f''(x) \neq 0$ *for all* $x \in [a, b]$ *(the concavity of f has a consistent sign, see Figure 1.3),*

then the sequence $(x_n)_{n\geq 0}$ generated by Newton's method converges to $\tilde{x}$ for any $x_0 \in [a, b]$ such that $f(x_0)f''(x_0) > 0$. Moreover, the convergence is quadratic (of order 2):

$$|x_n - \tilde{x}| \leq M|x_{n-1} - \tilde{x}|^2.$$

with $M = \dfrac{\max\limits_{x\in[a,b]} |f''(x)|}{2 \min\limits_{x\in[a,b]} |f'(x)|}$.

First condition and second condition ensure that there exists a unique $\tilde{x} \in [a, b]$ such that $f(\tilde{x}) = 0$.
Second condition guarantees that the sequence $(x_n)_{n\geq 0}$ is decreasing and bounded below by $\tilde{x}$ if $f''(x) > 0$ (i.e., f is convex), and increasing and bounded above by $\tilde{x}$ if $f''(x) < 0$ (i.e., f is concave).

Proof. Let us now demonstrate the quadratic convergence stated in Theorem (1.3). Using the second-order Taylor expansion of f at the point x_{n-1}, we obtain:

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

for some c between x_{n-1} and $\tilde{x}$.

Since $f(\alpha) = 0$, we get:

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

Dividing by $f'(x_{n-1})$, we obtain:

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

Therefore,

$$\alpha - \left(x_{n-1} - \frac{f(x_{n-1})}{f'(x_{n-1})}\right) = \frac{(\alpha - x_{n-1})^2 f''(c)}{2f'(x_{n-1})}.$$

Using Newton's recurrence relation $x_n = x_{n-1} - \dfrac{f(x_{n-1})}{f'(x_{n-1})}$, we finally obtain:

$$|x_n - \alpha| = \left|\frac{f''(c)}{2f'(x_{n-1})}\right| |x_{n-1} - \alpha|^2 \leq M|x_{n-1} - \alpha|^2.$$

with $M = \dfrac{\max\limits_{x\in[a,b]} |f''(x)|}{2 \min\limits_{x\in[a,b]} |f'(x)|}$. □

Remark 1.4. *1. If x_0 is chosen sufficiently close to $\tilde{x}$, Newton's method converges very rapidly to α because the convergence is quadratic.*

2. If $f'(\alpha) = 0$, Newton's method converges only linearly (i.e., with order 1) to α in a neighborhood of α.

1.6.3 Geometric Interpretation of Newton's Method

The idea is to replace a small arc of the curve $y = f(x)$ with its tangent drawn at a certain point on the curve.

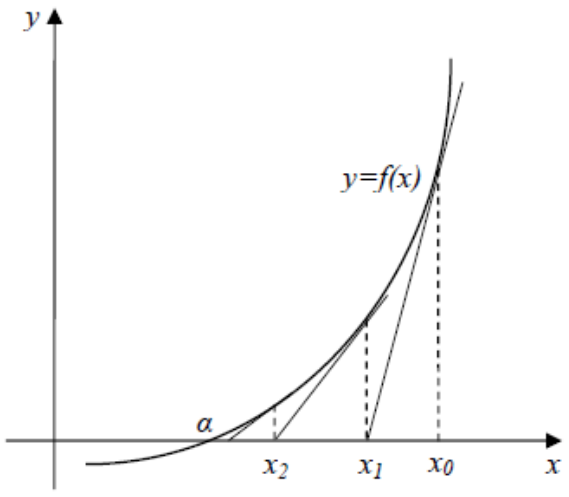


Figure 1.4: Geometric Interpretation of the Fixed Point Method.

From a well-chosen point $x_0 \in [a, b]$, x_1 is the x-coordinate of the point where the tangent to the graph of f at $(x_0, f(x_0))$ intersects the x-axis. Hence:

$$\frac{f(x_0)}{f'(x_0)} = x_0 - x_1 \Rightarrow x_1 = x_0 - \frac{f(x_0)}{f'(x_0)} \quad \text{(if } f'(x_0) \neq 0\text{)}.$$

Repeating the process at x_1, we obtain a new point x_2, and so on.

The sequence of points $(x_n)_{n\in\mathbb{N}}$ satisfies the recurrence relation:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}, \quad n = 0, 1, 2, \ldots$$

This is the most commonly used Newton-Raphson formula for finding roots.

1.6.4 Newton's Method Algorithm

1. Step 1. With the defined approximation error ε and initial guess $x_0 \in [a, b]$, set $k = 0$.
2. Step 2. Compute:
$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$$
3. Step 3. If $|x_{k+1} - x_k| < \varepsilon$, stop the iterations. The solution is $x^* = x_{k+1}$.
4. Otherwise, set $k = k + 1$ and return to Step 2.

Algorithm 1: Newton's Method Algorithm

1.6.5 Termination criterion

The Newton method iterations are stopped when the absolute difference between two successive iterations becomes smaller than a given precision ε:

$$|x_n - x_{n-1}| < \varepsilon.$$

1.7 Example: Solving with the Newton's Method

Approximate the solution α of the equation

$$x^2 e^x + 2x - 1 = 0, \quad x \in [0.3, 0.5]$$

using Newton's method with a precision $\varepsilon = 10^{-3}$. We can choose $x_0 = 0.5$.

- We have $f(0.3)f(0.5) = -0.278 \times 0.412 < 0$, and the derivative is $f'(x) = 2xe^x + x^2e^x + 2 > 0$ for all $x \in (0.3, 0.5)$, hence the solution α is unique in $[0.3, 0.5]$.

- The second derivative is $f''(x) = x^2e^x + 4xe^x + 2e^x > 0$ for all $x \in (0.3, 0.5)$, so f is convex.
- Therefore, the Newton sequence defined by

$$x_{n+1} = x_n - \frac{x_n^2 e^{x_n} + 2x_n - 1}{2x_n e^{x_n} + x_n^2 e^{x_n} + 2}$$

converges to the exact solution α.

Let us now perform the iterations:

Iteration 1: $x_0 = 0.5 \quad \Rightarrow \quad x_1 = x_0 - \frac{x_0^2 e^{x_0} + 2x_0 - 1}{2x_0 e^{x_0} + x_0^2 e^{x_0} + 2} = 0.3985.$

We have $|x_1 - x_0| = 0.101 > 10^{-3}$, so the stopping criterion is not satisfied.

Iteration 2: We compute the next iteration: $x_2 = x_1 - \frac{x_1^2 e^{x_1} + 2x_1 - 1}{2x_1 e^{x_1} + x_1^2 e^{x_1} + 2} = 0.3887.$

We have $|x_2 - x_1| = 0.098 > 10^{-3}$.

Iteration 3: We compute the next iteration: $x_3 = x_2 - \frac{x_2^2 e^{x_2} + 2x_2 - 1}{2x_2 e^{x_2} + x_2^2 e^{x_2} + 2} = 0.3886.$

We observe that $|x_3 - x_2| = 10^{-4} < 10^{-3}$. Therefore, the process is stopped.

The approximate solution is $x_3 = 0.3886$, thus $\alpha \approx x_3 = 0.3886$.

1.8 Exercises

Exercise 1.1. Let $f(x) = 2x^3 - x - 5$.

1. Plot the graph of the function f.
2. Show that the function f has a root in the interval $[1, 2]$.
3. Using the Bisection Method on the interval $[1, 2]$ with a precision $\varepsilon = 10^{-2}$, compute the zeros of this function.

Exercise 1.2. We want to approximate the solution of the following equation:

$$\ln(x) + x = 0.$$

1. Show that there is only one solution $x_0 \in \mathbb{R}^+$.
2. Apply the Newton-Raphson method on the interval $\left[\frac{1}{2}, \frac{3}{4}\right]$ for the first five iterations.

Exercise 1.3. We want to find the zeros of the following function:

$$f(x) = e^{x^2} - 4x^2.$$

1. Study graphically the equation $f(x) = 0$. Deduce that there is a root in the interval $[0, 1]$.
2. Study the convergence of the sequence generated by the Fixed Point Method and determine its order of convergence.
3. Study the convergence of the sequence generated by Newton's Method, specifying the order of convergence.
4. What conclusion can you draw?

1.9 Solutions to the Exercises

Solution to Exercise 1

Consider the function $f(x) = 2x^3 - x - 5$, which is continuous and differentiable on $\mathbb{R}$.

On the interval $[1, 2]$, we have

$$f'(x) = 6x^2 - 1 > 0,$$

which shows that f is strictly increasing on $[1, 2]$. Moreover,

$$f(1) = -4 < 0, \quad f(2) = 9 > 0.$$

By the Intermediate Value Theorem, there exists a unique $c \in (1, 2)$ such that

$$f(c) = 0.$$

Bisection Method with $\varepsilon = 10^{-2}$

We apply the method on the interval

$$1, 2$$

. The required number of iterations is

$$n \geq \frac{\ln\left(\frac{b-a}{\varepsilon}\right)}{\ln 2} - 1 \approx 6.$$

Iteration	Interval $[a_k, b_k]$	Midpoint $\frac{a_k+b_k}{2}$	$f\left(\frac{a_k+b_k}{2}\right)$	New Interval
0	$[1, 2]$	$\frac{3}{2}$	$\frac{1}{4} > 0$	$[1, \frac{3}{2}]$
1	$[1, \frac{3}{2}]$	$\frac{5}{4}$	$-\frac{75}{32} < 0$	$[\frac{5}{4}, \frac{3}{2}]$
2	$[\frac{5}{4}, \frac{3}{2}]$	$\frac{11}{8}$	$-\frac{301}{256} < 0$	$[\frac{11}{8}, \frac{3}{2}]$
3	$[\frac{11}{8}, \frac{3}{2}]$	$\frac{23}{16}$	$-\frac{1017}{2048} < 0$	$[\frac{23}{16}, \frac{3}{2}]$
4	$[\frac{23}{16}, \frac{3}{2}]$	$\frac{47}{32}$	$-\frac{2161}{16384} < 0$	$[\frac{47}{32}, \frac{3}{2}]$
5	$[\frac{47}{32}, \frac{3}{2}]$	$\frac{95}{64}$	≈ 0	Root $\in [\frac{47}{32}, \frac{3}{2}]$

Thus, after 5 iterations, the approximate root is

$$x \approx \frac{95}{64} \approx 1.484.$$

Solution Exercise 2

The function $f(x) = x + \ln(x)$ is defined, continuous, and strictly increasing on $\mathbb{R}^+$. Moreover,

$$\lim_{x \to 0^+} f(x) = -\infty, \quad \lim_{x \to +\infty} f(x) = +\infty.$$

By the Intermediate Value Theorem, there exists a unique $c \in \mathbb{R}^+$ such that $f(c) = 0$.

We now apply the Newton–Raphson method on the interval $I = \left[\frac{1}{2}, \frac{3}{4}\right]$. The Newton–Raphson iteration is given by

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

Starting with $x_0 = \frac{1}{2}$, we obtain:

$$x_1 \approx 0.56438239352.$$

[$x_2 \approx 0.567138987716$, $x_3 \approx 0.567143290399$, $x_4 \approx 0.567143290410$, $x_5 \approx 0.5671432904098 \approx 0.567143290410$.

We observe that with x_5, we already obtain 10 exact decimal places.

Solution Exercise 3

1. Let $f(x) = e^{x^2} - 4x^2$ be a function defined on $\mathbb{R}$. We observe that $f(-x) = f(x)$, i.e., the function f is even. Therefore, we can restrict our study to the interval $[0, +\infty[$.

   On the other hand, f is continuous and $f(0) = 1 > 0$ and $f(1) = e - 4 < 0$. Therefore, by the Intermediate Value Theorem, $\exists c \in]0, 1[: \ f(c) = 0$.

2. Let us study the convergence of the fixed-point sequence of the equation $f(x) = 0$:

   $$e^{x^2} - 4x^2 = 0 \quad \Longleftrightarrow \quad 2x = \sqrt{e^{x^2}} \quad \Longleftrightarrow \quad x = \frac{1}{2} e^{\frac{x^2}{2}}.$$

   We set

   $$g(x) = \frac{1}{2} e^{\frac{x^2}{2}},$$

   which is $C^1([0, 1])$.

   For $0 \leq x \leq 1$, we have

   $$0 \leq \frac{1}{2} e^{\frac{x^2}{2}} \leq \frac{1}{2} e^{\frac{1}{2}} < 1.$$

   Hence, $g([0, 1]) \subset [0, 1]$.

$$|g'(x)| = \left|\frac{x}{2}e^{\frac{x^2}{2}}\right| = |xg(x)| < |x| < 1,$$

thus g is contractive.

Therefore, by the Fixed Point Theorem, the sequence defined by

$$x_{n+1} = g(x_n), \quad x_0 \in [0, 1],$$

is convergent to the root $c \in]0, 1[$.

On the other hand,

$$g'(c) = cg(c) = c^2 \neq 0,$$

hence the fixed-point method converges only with order 1.

3. We define the Newton sequence of the function f as follows:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} = x_n - \frac{e^{x_n^2} - 4x_n^2}{2x_n e^{x_n^2} - 8x_n} = x_n - \frac{e^{x_n^2} - 4x_n^2}{2x_n\left(e^{x_n^2} - 4\right)}.$$

Since c is a simple root of f, the Newton method has order 2.

4. We conclude that the Newton method is more efficient than the fixed-point method.

CHAPTER 2

NUMERICAL INTEGRATION

The objective of this chapter is to compute definite integrals numerically. Let f be a function defined and continuous on the interval $[a, b]$. We consider the following definite integral:

$$I_f = \int_a^b f(x)\,dx. \tag{2.1}$$

In many situations, the computation of I_f by analytical methods is very complicated, or even impossible. For example, $\int_0^2 e^{-x^2}\,dx$ cannot be calculated using exact methods, since $\int e^{-x^2}\,dx$

does not have an explicit antiderivative (e^{-x^2} admits no primitive). Consequently, numerical methods are used to approximate the integral I_f.

Let us first subdivide the interval $[a, b]$ into small subintervals $[x_{i-1}, x_i]$, for $i = 1, 2, \ldots, n$ such that $a = x_0 < x_1 < \cdots < x_{n-1} < x_n = b$. We define the step size by $h_i = x_i - x_{i-1}$. It is clear that

$$I_f = \int_a^b f(x)\,dx = \int_{x_0}^{x_1} f(x)\,dx + \int_{x_1}^{x_2} f(x)\,dx + \cdots + \int_{x_{n-1}}^{x_n} f(x)\,dx \tag{2.2}$$

$$= \sum_{i=1}^{n} \int_{x_{i-1}}^{x_i} f(x)\,dx. \tag{2.3}$$

The principle of numerical integration is to approximate (or simply replace) f on the interval $[x_{i-1}, x_i]$ by a polynomial p of degree $d \geq 0$ (i.e., $f(x) \approx p(x)$ on $[x_{i-1}, x_i]$).

In this chapter, we present three of the most commonly used numerical integration methods, namely: the rectangle method with its three variants (left, right, and midpoint), the trapezoidal rule, and finally Simpson's rule.

2.1 Rectangle Methods

2.1.1 Riemann Sums

The so-called rectangle approximation methods are natural and rely on the notion of Riemann sums. Let us briefly recall this notion.

Let $f : [a, b] \to \mathbb{R}$ be a piecewise continuous function. A subdivision σ of $[a, b]$ is a finite set $\{a_0, \dots, a_k\}$ such that

$$a = a_0 < a_1 < a_2 < \cdots < a_k = b.$$

We denote $|\sigma| = \max_{i=0,\dots,k-1}(a_{i+1} - a_i)$ the length of the subdivision. If for every $i = 0, \dots, k-1$ we choose $x_i \in [a_i, a_{i+1}]$, we say that $\sigma^p = (\sigma, (x_i)_i)$ is a *tagged subdivision* of $[a, b]$.

We the n define

$$S_{\sigma^p}(f) := \sum_{i=0}^{k-1} f(x_i)(a_{i+1} - a_i).$$

This sum is called the *Riemann sum* associated with the tagged subdivision σ^p, see Figure 2.1. The foundation of

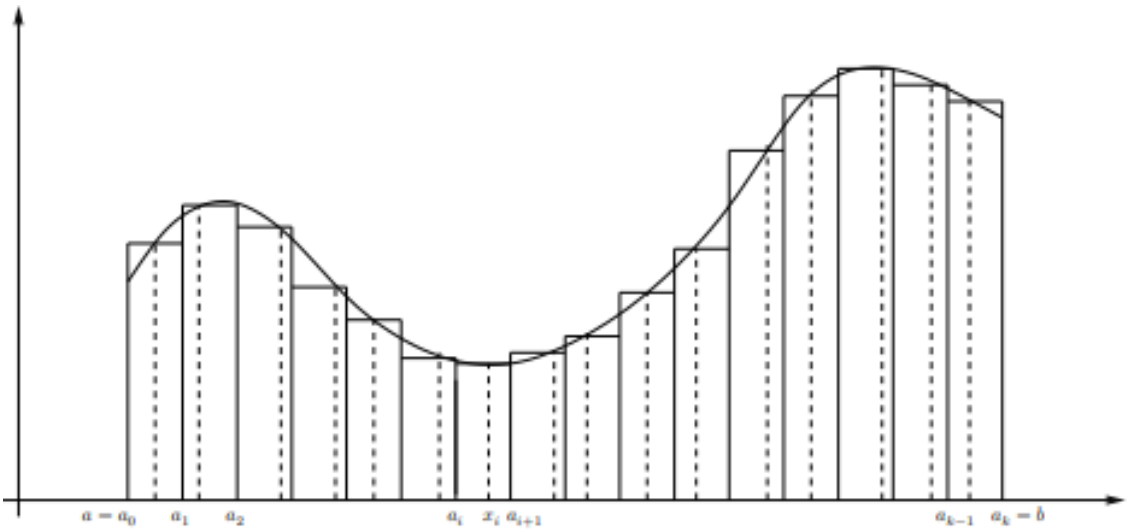


Figure 2.1: Riemann sum associated with the tagged subdivision σ^p.

the theory of Riemann integration states that

Theorem 2.1. *If f is piecewise continuous, then for any sequence $(\sigma_n^p)_n$ of tagged subdivisions of $[a, b]$ such that $|\sigma_n^p| \to 0$ as $n \to \infty$, we have*

$$\lim_{n\to\infty} S_{\sigma_n^p}(f) = \int_a^b f(x)\,dx.$$

2.1.2 Left and Right Rectangles

The left and right rectangle methods are based on Theorem 2.1, where we respectively choose the following tagged subdivisions:

- **Left rectangles:** $\sigma_n^p = \{a_0, \dots, a_n\}$ where $a_k = a + \dfrac{k(b-a)}{n}$ and $x_k = a_k$,
- **Right rectangles:** $\sigma_n^p = \{a_0, \dots, a_n\}$ where $a_k = a + \dfrac{k(b-a)}{n}$ and $x_k = a_{k+1}$.

In other words, we divide the interval $[a, b]$ into n segments of equal length, each equal to $\dfrac{b-a}{n}$, and we choose as points x_k the left endpoints (respectively, the right endpoints) of each subinterval.

We denote simply S_n^g and S_n^d the corresponding Riemann sums, i.e.,

$$S_n^g = \frac{b-a}{n} \sum_{k=0}^{n-1} f\left(a + \frac{k}{n}(b-a)\right), \qquad \text{and} \qquad S_n^d = \frac{b-a}{n} \sum_{k=1}^{n} f\left(a + \frac{k}{n}(b-a)\right)$$

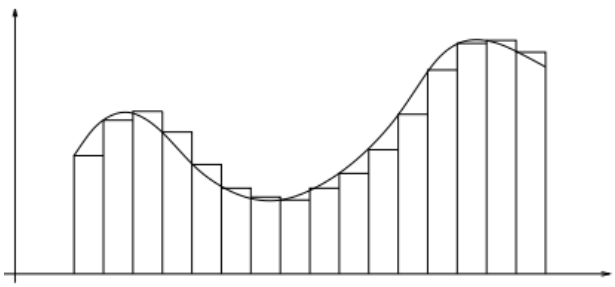


Figure 2.2: Left rectangles

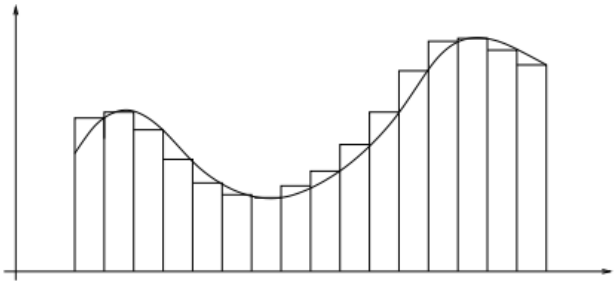


Figure 2.3: Right rectangles

2.1.3 Error Analysis for Left and Right Rectangles

We know that the two sequences $(S_n^g)_n$ and $(S_n^d)_n$ converge to $\int_a^b f(x)\,dx$. The question is therefore to estimate the error if we take one of these values as an approximation.

Proposition 2.1. *Let $f : [a, b] \to \mathbb{R}$ be a C^1 function and let $M_1 = \sup_{x\in[a,b]} |f'(x)|$. Then, for any n,*

$$\left| \int_a^b f(x)\,dx - S_n^g \right| \leq \frac{M_1(b-a)^2}{2n}, \qquad \left| \int_a^b f(x)\,dx - S_n^d \right| \leq \frac{M_1(b-a)^2}{2n}. \tag{2.1}$$

Proof. We prove the result for S_n^g, the proof being the same for S_n^d. We set $a_k = a + \dfrac{k}{n}(b-a)$. First, note that M_1 is indeed a finite number. In fact, since $f \in C^1$, we know that f' is continuous on the segment $[a, b]$, and therefore bounded there.

By Chasles' relation and the triangular inequality, we obtain

$$\begin{aligned}\left|\int_a^b f(x)\,dx - S_n^g\right| &= \left|\sum_{k=0}^{n-1}\int_{a_k}^{a_{k+1}} f(x)\,dx - \frac{b-a}{n}\sum_{k=0}^{n-1} f(a_k)\right| \\ &= \left|\sum_{k=0}^{n-1}\int_{a_k}^{a_{k+1}} \big(f(x) - f(a_k)\big)\,dx\right| \\ &\leq \left|\sum_{k=0}^{n-1}\int_{a_k}^{a_{k+1}} |f(x) - f(a_k)|\,dx\right|.\end{aligned}$$

By the Mean Value Inequality, we can write, for every k and for every $x \in [a_k, a_{k+1}]$,

$$|f(x) - f(a_k)| \leq M_1(x - a_k).$$

Hence,

$$\begin{aligned}\left|\int_a^b f(x)\,dx - S_n^g\right| &\leq M_1\sum_{k=0}^{n-1}\int_{a_k}^{a_{k+1}} (x - a_k)\,dx \\ &\leq \frac{M_1}{2}\sum_{k=0}^{n-1}(a_{k+1} - a_k)^2.\end{aligned}$$

Since $a_{k+1} - a_k = \dfrac{b-a}{n}$, we obtain

$$\left|\int_a^b f(x)\,dx - S_n^g\right| \leq \frac{M_1(b-a)^2}{2n}.$$

□

Remark 2.1. *One can show that the error bound is optimal in the sense that there exist functions f for which equality holds in (2.1). For this to happen, all the inequalities in the proof must be equalities. This is the case if f is affine.*

For example, if $f(x) = x$ on $[0, 1]$, we have

$$S_n^g = \frac{n-1}{2n} \qquad \textit{and} \qquad S_n^d = \frac{n+1}{2n},$$

while $\displaystyle\int_0^1 x\,dx = \frac{1}{2}$. We then find that

$$\int_0^1 x\,dx - S_n^g = \int_0^1 x\,dx - S_n^d = \frac{1}{2n} = \frac{M_1(b-a)^2}{2n}.$$

Example 2.1. Let us consider the function

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

Compute the approximate value of the integral of $f(x)$ over $[a, b] = [0, 1]$ using the composite rectangle method with $n = 6$.

Solution.

We have

$$I = \int_a^b f(x)\,dx \approx h\sum_{i=0}^{n-1} f_i,$$

where

$$x_i = a + ih, \quad h = \frac{b-a}{n} = \frac{1}{6}, \quad f_i = f(x_i), \quad i = 0, \ldots, 5.$$

Thus,

$$x_0 = a = 0, \quad x_1 = \tfrac{1}{6}, \quad x_2 = \tfrac{2}{6}, \quad x_3 = \tfrac{3}{6}, \quad x_4 = \tfrac{4}{6}, \quad x_5 = \tfrac{5}{6}.$$

$$f_0 = 0^2 = 0, \quad f_1 = \left(\tfrac{1}{6}\right)^2 = \tfrac{1}{36}, \quad f_2 = \left(\tfrac{2}{6}\right)^2 = \tfrac{4}{36},$$
$$f_3 = \left(\tfrac{3}{6}\right)^2 = \tfrac{9}{36}, \quad f_4 = \left(\tfrac{4}{6}\right)^2 = \tfrac{16}{36}, \quad f_5 = \left(\tfrac{5}{6}\right)^2 = \tfrac{25}{36}.$$

Therefore,

$$I \approx h\sum_{i=0}^{5} f_i = \frac{1}{6}\left(0 + \tfrac{1}{36} + \tfrac{4}{36} + \tfrac{9}{36} + \tfrac{16}{36} + \tfrac{25}{36}\right).$$

$$I \approx \frac{1}{6}\cdot\frac{55}{36} = \frac{55}{216} \approx 0.2546.$$

The exact value of the integral is

$$I = \int_0^1 x^2\,dx = \left[\tfrac{x^3}{3}\right]_0^1 = \tfrac{1}{3} \approx 0.3333.$$

The relative error is

$$\varepsilon_r = \frac{|0.3333 - 0.2546|}{0.3333} \approx 0.236 \approx 23.6\%.$$

2.1.4 Midpoint Method

In the left and right rectangle methods, the convergence appears to be slow. However, the actual error may in principle be much smaller than the one given by (2.1).

Indeed, to obtain the latter we estimated the error committed on each subinterval $[a_k, a_{k+1}]$ and then added these errors. However, these errors may "compensate" each other, some being positive and others negative.

In the left rectangle method, the error is negative on intervals where f is increasing (in the sense that $\frac{b-a}{n} f(a_k) \leq \int_{a_k}^{a_{k+1}} f(x)\,dx$) and it is positive when f is decreasing.

Conversely, in the right rectangle method, the error is positive on intervals where f is increasing and negative when f is decreasing.

In particular, if the function f is increasing on $[a, b]$ or decreasing on $[a, b]$, there is no compensation of errors between the intervals, which explains why the error bound (2.1) can be attained.

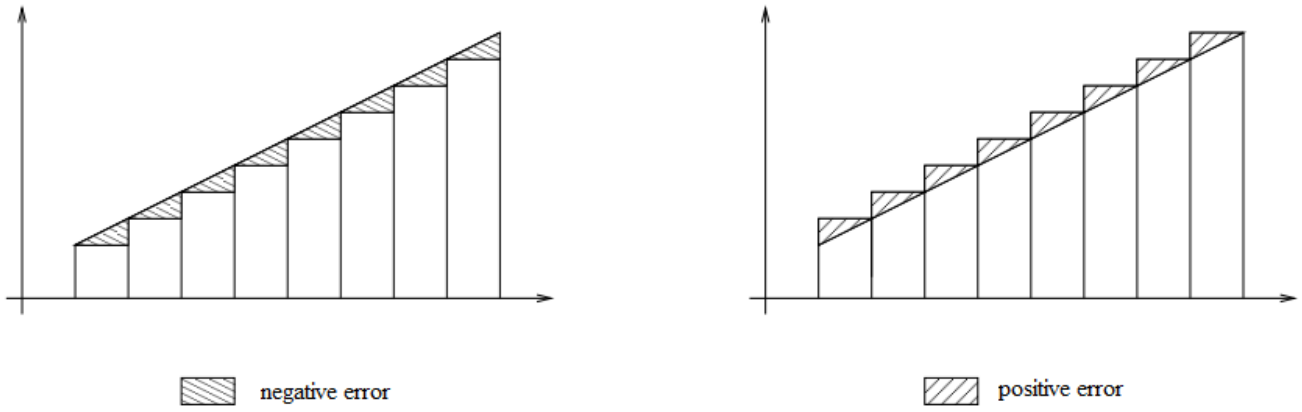


Figure 2.4: Errors in the left and right rectangle methods.

The midpoint method provides a significant improvement in the rate of convergence compared to the left or right rectangle methods. The idea here is to subdivide the interval $[a, b]$ again into subintervals of equal length, i.e., we still set $a_k = a + \frac{k}{n}(b-a)$, but we choose the points x_k at the midpoint of the subintervals, i.e., $x_k = \frac{a_k + a_{k+1}}{2}$.

Whether on an interval $[a_k, a_{k+1}]$ the function f is increasing or decreasing, there will be a form of "partial compensation" within the interval itself.

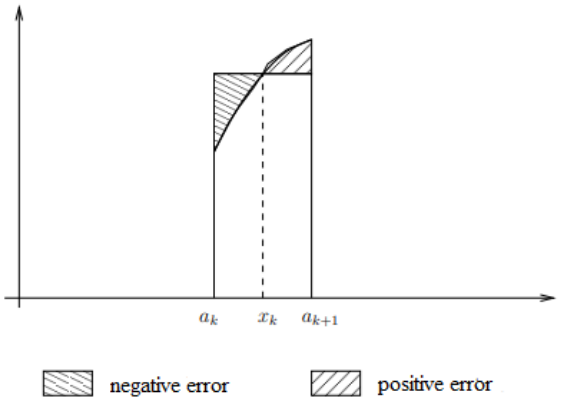


Figure 2.5: Errors in the midpoint method.

We denote $m_k = \frac{(a_k + a_{k+1})}{2}$ the midpoint of the interval $[a_k, a_{k+1}]$, and

$$S_n^m = \frac{(b-a)}{n} \sum_{k=0}^{n-1} f(m_k).$$

2.1.5 Error Analysis for Midpoint Method

The associated Riemann sum. Again, Theorem (2.1) ensures that the sequence $(S_n^m)_n$ converges to $\int_a^b f(x)dx$, and we wish to estimate the error $\left| \int_a^b f(x)dx - S_n^m \right|$.

Proposition 2.2. *Let $f : [a, b] \longrightarrow R$ be of class C^2, and let $M_2 = sup_{x\in[a,b]}|f''(x)|$. Then, for all n,*

$$\left|\int_a^b f(x)dx - S_n^m\right| \leq \frac{M_2(b-a)^3}{(24n^2)}. \tag{2.2}$$

Proof. The same computation as in the proof of (2.1) gives

$$\int_a^b f(x)\,dx - S_n^m = \sum_{k=0}^{n-1}\int_{a_k}^{a_{k+1}} \big(f(x) - f(m_k)\big)\,dx.$$

For each k, we apply the Taylor–Lagrange formula between x and m_k: there exists c_k between x and m_k such that

$$f(x) - f(m_k) = f'(m_k)(x - m_k) + \frac{f''(c_k)}{2}(x - m_k)^2.$$

Hence,

$$\int_a^b f(x)\,dx - S_n^m = \sum_{k=0}^{n-1}\left(f'(m_k)\int_{a_k}^{a_{k+1}}(x - m_k)\,dx + \int_{a_k}^{a_{k+1}}\frac{f''(c_k)}{2}(x - m_k)^2\,dx\right).$$

Since m_k is the midpoint of $[a_k, a_{k+1}]$, we have

$$\int_{a_k}^{a_{k+1}}(x - m_k)\,dx = \tfrac{1}{2}\Big((a_{k+1} - m_k)^2 - (a_k - m_k)^2\Big) = 0,$$

which yields

$$\begin{aligned}\left|\int_a^b f(x)\,dx - S_n^m\right| &= \left|\sum_{k=0}^{n-1}\int_{a_k}^{a_{k+1}}\frac{f''(c_k)}{2}(x - m_k)^2\,dx\right| \\ &\leq \sum_{k=0}^{n-1}\frac{M_2}{2}\int_{a_k}^{a_{k+1}}(x - m_k)^2\,dx \\ &\leq \frac{M_2(b-a)^3}{24n^2}.\end{aligned}$$

□

Remark 2.2. *One might think that the improvement in the error compared to the left or right rectangle methods comes from using the second-order Taylor expansion. This is not the case. If we apply the same strategy to the left rectangles, for instance, the first-order term gives*

$$\int_{a_k}^{a_{k+1}}(x - a_k)\,dx = \frac{1}{2}(a_{k+1} - a_k)^2 = \frac{(b-a)^2}{2n^2}.$$

Thus, we obtain

$$\int_a^b f(x)\,dx - S_n^g = \sum_{k=0}^{n-1} f'(a_k)\,\frac{(b-a)^2}{2n^2} + \sum_{k=0}^{n-1}\int_{a_k}^{a_{k+1}}\frac{f''(c_k)}{2}(x - a_k)^2\,dx.$$

The second term on the right-hand side can be bounded by $\frac{(b-a)^3}{6n^2}$, and is therefore also of order $\frac{1}{n^2}$. However, the first term is a priori bounded by

$$\sum_{k=0}^{n-1} f'(a_k)\,\frac{(b-a)^2}{2n^2} \leq \frac{M_1(b-a)^2}{2n}.$$

This recovers exactly the error obtained in (2.1).

Example 2.2. We evaluate numerically, for $n = 3$, the integral

$$I(f) = \int_0^{\frac{\pi}{2}} \sin x \, dx,$$

whose exact value is $I(f) = 1$.

We have $a = 0$, $b = \frac{\pi}{2}$, $n = 3$, $h = \dfrac{b-a}{n}$,

$$x_i = a + ih, \quad 0 \le i \le n, \quad f(x) = \sin x.$$

Thus,

$$h = \frac{\frac{\pi}{2} - 0}{3} = \frac{\pi}{6}, \quad x_i = \frac{i\pi}{6}, \quad 0 \le i \le 3.$$

We obtain the following tables:

i	x_i	$\dfrac{x_{i-1}+x_i}{2}$	$f(x_i) = \sin(x_i)$	$f\left(\dfrac{x_{i-1}+x_i}{2}\right)$
0	0		0	
		$\frac{\pi}{12}$		$\sin\frac{\pi}{12} = \frac{\sqrt{6}-\sqrt{2}}{4}$
1	$\frac{\pi}{6}$		$\frac{1}{2}$	
		$\frac{\pi}{4}$		$\sin\frac{\pi}{4} = \frac{\sqrt{2}}{2}$
2	$\frac{\pi}{3}$		$\frac{\sqrt{3}}{2}$	
		$\frac{5\pi}{12}$		$\sin\frac{5\pi}{12} = \frac{\sqrt{6}+\sqrt{2}}{4}$
3	$\frac{\pi}{2}$		1	

For the integral, we compute:

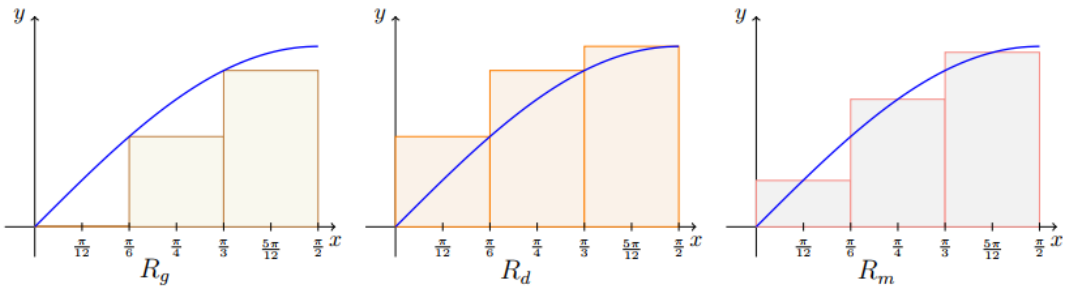
$$R_g = h\sum_{i=1}^{3} f(x_{i-1}) = \frac{\pi}{6}\left(0 + \tfrac{1}{2} + \tfrac{\sqrt{3}}{2}\right) = \frac{1+\sqrt{3}}{12}\pi \approx 0.7152492,$$

$$R_d = h\sum_{i=1}^{3} f(x_i) = \frac{\pi}{6}\left(\tfrac{1}{2} + \tfrac{\sqrt{3}}{2} + 1\right) = \frac{3+\sqrt{3}}{12}\pi \approx 1.238848,$$

$$R_m = h\sum_{i=1}^{3} f\left(\tfrac{x_{i-1}+x_i}{2}\right) = \frac{\pi}{6}\left(\sin\tfrac{\pi}{12} + \sin\tfrac{\pi}{4} + \sin\tfrac{5\pi}{12}\right) \approx 1.011515.$$

2.2 Trapezoidal Method

The previous methods are based on the notion of Riemann sums, and therefore on the idea of approximating, on each subinterval $[a_k, a_{k+1}]$, the function f by a constant function (equal to the value of f at the chosen point). It is natural to try instead to approximate f not by a constant, but by an affine function (or even a higher-order polynomial, see Chapter 3).



Once again, we consider only the case where the intervals $[a_k, a_{k+1}]$ all have the same length, i.e. $a_k = \frac{k}{n}(b-a)$. We denote by f_k the affine function that interpolates f at the points $(a_k, f(a_k))$ and $(a_{k+1}, f(a_{k+1}))$. In other words,

$$f_k(x) = \frac{f(a_{k+1}) - f(a_k)}{a_{k+1} - a_k}(x - a_k) + f(a_k).$$

We will take f_k as an approximation of f on the interval $[a_k, a_{k+1}]$. The approximated value of the integral of f is then given by

$$T_n := \sum_{k=0}^{n-1} \int_{a_k}^{a_{k+1}} f_k(x)\,dx. \tag{2.3}$$

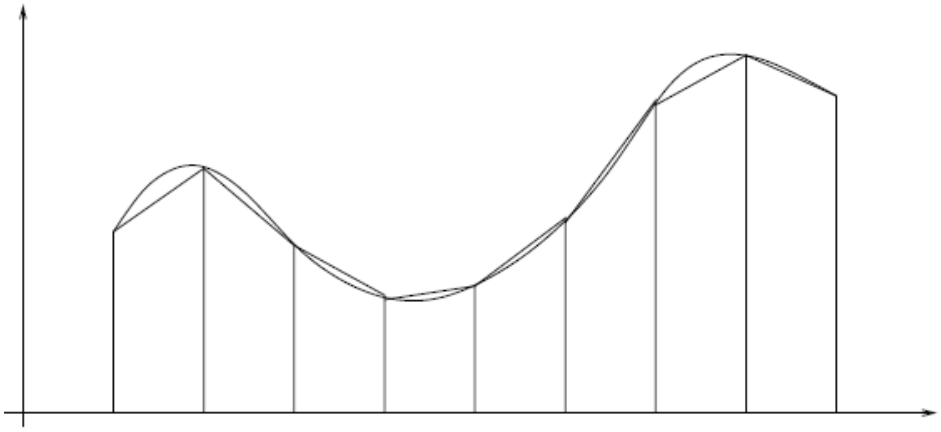


Figure 2.6: Trapezoidal Method.

We can easily compute (this is the area of a trapezoid, see Figure 2.6)

$$\int_{a_k}^{a_{k+1}} f_k(x)\,dx = \frac{1}{2}\big(f(a_{k+1}) + f(a_k)\big)(a_{k+1} - a_k) = \frac{b-a}{n}.$$

Thus we obtain

$$T_n = \frac{b-a}{2n}\sum_{k=0}^{n-1}\big(f(a_{k+1}) + f(a_k)\big) = \frac{b-a}{2n}\left(f(a) + 2\sum_{k=1}^{n-1} f(a_k) + f(b)\right).$$

2.2.1 Error Analysis for Trapezoidal Method

We then have the following upper bound for the error.

Proposition 2.3. *Let $f : [a, b] \to \mathbb{R}$ be of class C^2 and let $M_2 = \sup_{x\in[a,b]} |f''(x)|$. Then, for all n,*

$$\left| \int_a^b f(x)\,dx - T_n \right| \leq \frac{M_2(b-a)^3}{12n^2}. \tag{2.4}$$

Proof. The beginning of the proof is the same: using Chasles' relation and the triangle inequality, we write

$$\int_a^b f(x)\,dx - T_n \;\leq\; \sum_{k=0}^{n-1} \int_{a_k}^{a_{k+1}} \big(f(x) - f_k(x)\big)\,dx.$$

The idea is then to estimate each term

$$\int_{a_k}^{a_{k+1}} g_k(x)\,dx, \quad \text{where } g_k(x) = f(x) - f_k(x),$$

by first performing an integration by parts. We first note that, by construction, $g_k(a_k) = g_k(a_{k+1}) = 0$, so there is no boundary term in the integration by parts. Thus we can write

$$\int_{a_k}^{a_{k+1}} g_k(x)\,dx = -\int_{a_k}^{a_{k+1}} g_k'(x)(x-c)\,dx.$$

By parts, still without generating boundary terms. The primitives of $v(x) = x - c$ are of the form $\dfrac{1}{2}(x-c)^2 + d$, and therefore

$$\begin{aligned}\int_{a_k}^{a_{k+1}} g_k(x)\,dx &= -\int_{a_k}^{a_{k+1}} g_k'(x)(x-c)\,dx \int_{a_k}^{a_{k+1}} g_k'(x)\left(\tfrac{1}{2}(x-c)^2 + d\right)dx \\ &= -\Big[g_k''(x)\left(\tfrac{1}{2}(x-c)^2 + d\right)\Big]_{a_k}^{a_{k+1}}.\end{aligned}$$

In order for the new boundary term to vanish, it suffices that the polynomial $\frac{1}{2}(x-c)^2 + d$ vanishes at a_k and a_{k+1}. In other words, we choose c and d so that

$$\tfrac{1}{2}(x-c)^2 + d = \tfrac{1}{2}(x-a_k)(x-a_{k+1}).$$

We then have

$$\begin{aligned}\left| \int_{a_k}^{a_{k+1}} g_k(x)\,dx \right| &\leq \frac{1}{2}\int_{a_k}^{a_{k+1}} \left|g_k''(x)(x-a_k)(x-a_{k+1})\right| dx \\ &\leq \frac{M_2}{2}\int_{a_k}^{a_{k+1}} (x-a_k)(a_{k+1}-x)\,dx \\ &= \frac{M_2(a_{k+1}-a_k)^3}{12} \\ &= \frac{M_2(b-a)^3}{12n^3},\end{aligned}$$

where in the second line we used the fact that f_k is affine and therefore $|g_k''(x)| = |f''(x)| \leq M_2$.

Finally, we obtain

$$\left| \int_a^b f(x)\,dx - T_n \right| \leq \sum_{k=0}^{n-1} \left| \int_{a_k}^{a_{k+1}} g_k(x)\,dx \right| \leq \frac{M_2(b-a)^3}{12n^2}.$$

□

Example 2.3. Consider

$$I = \int_0^1 e^{-x^2}\,dx.$$

1. Approximate I using the trapezoidal rule for $n = 5$ with a uniform step size.

2. Determine the maximum error committed by the trapezoidal rule.

Solution. We provide in the following table the values of x_i, $f(x_i)$ and $\dfrac{f(x_{i-1}) + f(x_i)}{2}$ for $n = 5$ with the step size $h = \frac{1-0}{5} = 0.2$.

x_i	$f(x_i)$	$\dfrac{f(x_{i-1}) + f(x_i)}{2}$
0	1.0000	0.9803
0.2	0.9607	0.9064
0.4	0.8521	0.7749
0.6	0.6976	0.6124
0.8	0.5272	0.4475
1	0.3678	

By applying the trapezoidal rule formula given by (2.7), we obtain the following approximation:

$$I_T = h \sum_{i=1}^{5} \frac{f(x_{i-1}) + f(x_i)}{2} = 0.2\,(0.9803 + 0.9064 + \cdots + 0.4475) = 0.7443.$$

To determine the maximum error, we need to compute the second derivative. For $f(x) = e^{-x^2}$, we have

$$f'(x) = -2xe^{-x^2}, \qquad f''(x) = e^{-x^2}(4x^2 - 2).$$

Thus,

$$V = \max_{x\in[a,b]} |f''(x)| = \max_{x\in[0,1]} \left|e^{-x^2}(4x^2 - 2)\right| = |f''(0)| = 2.$$

Therefore, the maximum error of the trapezoidal method is given by

$$\frac{V}{12}(b-a)h^2 = \frac{2}{12}(1-0)(0.2)^2 = \frac{0.04}{6} \approx 0.0066.$$

2.3 Simpson's Method

We conclude this first chapter on numerical integration with Simpson's method. In this case, the idea is again to approximate the function f on each interval $[a_k, a_{k+1}]$, with $a_k = \frac{k}{n}(b-a)$, by a "simple" function. We have already used a constant function, or a polynomial of degree at most 0, in Section 2.1, and an affine function, or a polynomial of degree at most 1, in Section 2.2. It is natural here to choose a polynomial of degree at most 2.

Given an interval $[c, d]$, we consider the polynomial $P \in \mathbb{R}_2[X]$ that interpolates f at the points of abscissa c, d, and $m = \dfrac{c+d}{2}$. That is, $P(X) = \alpha + \beta X + \gamma X^2$ such that $P(c) = f(c)$, $P(m) = f(m)$, and $P(d) = f(d)$.

Lemma 2.1. *There exists a unique $P \in \mathbb{R}_2[X]$ which interpolates f at the points with abscissas c, d, and m. It is given by*

$$P(X) = f(c)\frac{(X-m)(X-d)}{(c-m)(c-d)} + f(m)\frac{(X-c)(X-d)}{(m-c)(m-d)} + f(d)\frac{(X-c)(X-m)}{(d-m)(d-c)}. \tag{2.5}$$

This formula is a particular case of the Lagrange interpolation polynomial. We will return to this in Section 3.1. A straightforward computation shows that if P is given by (2.5), then

$$\int_c^d P(x)\,dx = \frac{d-c}{6}\big(f(c) + 4f(m) + f(d)\big). \tag{2.6}$$

As in the previous sections, on each interval $[a_k, a_{k+1}]$ we approximate f by the corresponding interpolation polynomial and the integral of f by that of the polynomial. Thus, Simpson's method consists in approximating $\int_a^b f(x)\,dx$ by (where for each k we have $a_{k+1} - a_k = \frac{b-a}{n}$)

$$S_n := \frac{b-a}{6n}\sum_{k=0}^{n-1}\Big(f(a_k) + 4f(m_k) + f(b_k)\Big).$$

2.3.1 Error Analysis for Simpson's Method

The error estimate is based on the following lemma

Lemma 2.2. *Let $f : [c, d] \to \mathbb{R}$ be a function of class C^4 and let $M_4 = \sup_{x\in[c,d]} |f^{(4)}(x)|$. Then*

$$\left|\int_c^d f(x)\,dx - \frac{d-c}{6}\big(f(c) + 4f\big(\tfrac{c+d}{2}\big) + f(d)\big)\right| \le \frac{M_4(d-c)^5}{2880}.$$

By applying the lemma on each interval $[a_k, a_{k+1}]$, we immediately obtain:

Proposition 2.4. *Let $f : [a, b] \to \mathbb{R}$ be of class C^4 and let $M_4 = \sup_{x\in[a,b]} |f^{(4)}(x)|$. Then, for all n,*

$$\left|\int_a^b f(x)\,dx - S_n\right| \le \frac{M_4(b-a)^5}{2880n^4}.$$

The Simpson's method has an error of order h^4. It is the most commonly used method in practice because its error converges very quickly to zero as $h \to 0$, compared to other methods (the trapezoidal rule of order h^2 and the rectangle rule of order h or h^2).

Example 2.4. • Compute $I = \int_0^2 x^2 e^x \, dx$ using Simpson's method for $n = 8$ with a regular step.

• Determine the maximum error committed by Simpson's method.

Solution We present in the following table the values of x_i, $f(x_i)$, $\dfrac{f(x_{i-1}) + f(x_i)}{2}$, and $f(\bar{x}_i) = f\Big(\frac{x_{i-1}+x_i}{2}\Big)$ for $n = 8$ with the step $h = \frac{2-0}{8} = 0.25$.

x_i	$f(x_i)$	$\dfrac{f(x_{i-1}) + f(x_i)}{2}$	$\bar{x}_i = \dfrac{x_{i-1} + x_i}{2}$	$f(\bar{x}_i)$
0	0	0.0802	0.125	0.0177
0.25	0.0802	0.4924	0.375	0.2046
0.50	0.4121	1.6029	0.625	0.7297
0.75	1.1908	3.9090	0.875	1.8366
1.00	2.7182	8.1719	1.125	3.8983
1.25	5.4536	15.5374	1.375	7.4775
1.50	10.0838	27.7072	1.625	13.4102
1.75	17.6234	47.1796	1.875	22.9247
2.00	29.5562			

Applying Simpson's formula, we obtain the following approximation:

$$\begin{aligned} \hat{I}_S &= \frac{h}{6} \sum_{i=1}^{8} \left(f(x_{i-1}) + f(x_i) + 4f(\bar{x}_i)\right) \\ &= \frac{0.25}{6}\Big((0.0802 + \cdots + 47.1796) + 4(0.0177 + \cdots + 22.9247)\Big) \\ &= 12.7783. \end{aligned}$$

To determine the maximum error, we need to compute the fourth derivative. Since $f(x) = x^2 e^x$, we have:

$$f'(x) = x^2 e^x + 2xe^x, \quad f''(x) = x^2 e^x + 4xe^x + 2e^x,$$

$$f^{(3)}(x) = x^2 e^x + 6xe^x + 6e^x, \quad f^{(4)}(x) = x^2 e^x + 8xe^x + 12e^x.$$

We get

$$W = \max_{x \in [a,b]} |f^{(4)}(x)| = \max_{x \in [0,2]} \left(x^2 e^x + 8xe^x + 12e^x\right) = f^{(4)}(2) = 236.4498,$$

because $f^{(4)}(x)$ is strictly positive and strictly increasing on $[0, 2]$.

Therefore, the maximum error of Simpson's method is given by

$$\frac{W}{2880}(b-a)h^4 = \frac{236.4498}{2880}(2-0)(0.25)^4 \approx 0.0006.$$

2.4 Exercises

Exercise 2.1. Let the function $f(x)$ be defined by the following table:

x	0	$\frac{\pi}{8}$	$\frac{\pi}{4}$	$\frac{3\pi}{8}$	$\frac{\pi}{2}$
$f(x)$	0	0.382683	0.707107	0.923880	1

1. Determine $\int_0^{\pi/2} f(x)\,dx$ using the trapezoidal rule and then Simpson's rule.

2. Knowing that $f(x) = \sin x$, compare the obtained results with the exact value.

Exercise 2.2. Let the integral

$$I = \int_0^{\pi} \sin(x)\,dx.$$

1. Compute the exact value of I.

2. Using the composite trapezoidal rule and the composite Simpson's rule with step size $h = \frac{\pi}{4}$:

   (a) Compute the approximation of I.

   (b) Provide an upper bound for the error.

   (c) Evaluate the error.

3. Determine the step size h and the number of subdivisions of the interval $[0, \pi]$ such that the error obtained by the composite trapezoidal rule (respectively, the composite Simpson's rule) is less than 5×10^{-4}.

2.5 Solutions to the Exercises

Solution Exercise 1

1/ Calculation of the integral using the trapezoidal rule

$$I_T(f) = \frac{b-a}{2n}\left[f(a) + f(b) + 2\sum_{i=1}^{n-1} f(x_i)\right], \quad h = \frac{b-a}{n} = \frac{\pi/2}{4} = \frac{\pi}{8}.$$

Thus,

$$I_T(f) = \frac{\pi}{16}\left[f(0) + f\left(\tfrac{\pi}{2}\right) + 2\left(f\left(\tfrac{\pi}{8}\right) + f\left(\tfrac{\pi}{4}\right) + f\left(\tfrac{3\pi}{8}\right)\right)\right].$$

$$I_T(f) = \frac{\pi}{16}\left[0 + 1 + 2(0.382683 + 0.707107 + 0.923880)\right] \approx 0.987116.$$

- Calculation of the integral using Simpson's rule

$$I_S(f) = \frac{h}{3}\left[f(a) + f(b) + 4\sum_{\substack{i=1\\ i\,\text{odd}}}^{n-1} f(x_i) + 2\sum_{\substack{i=2\\ i\,\text{even}}}^{n-2} f(x_i)\right], \quad h = \frac{\pi}{8}.$$

$$I_S(f) = \frac{\pi}{24}\left[f(0) + f\left(\tfrac{\pi}{2}\right) + 4\left(f\left(\tfrac{\pi}{8}\right) + f\left(\tfrac{3\pi}{8}\right)\right) + 2f\left(\tfrac{\pi}{4}\right)\right].$$

$$I_S(f) = \frac{\pi}{24}\left[0 + 1 + 4(0.382683 + 0.923880) + 2(0.707107)\right] \approx 1.000135.$$

2/ Comparison

The exact value is

$$I_{\text{exact}} = \int_0^{\pi/2} \sin(x)\,dx = \left[-\cos(x)\right]_0^{\pi/2} = 1.$$

Errors:

$$|I_{\text{exact}} - I_T| = 1 - 0.987116 = 0.012884,$$

$$|I_{\text{exact}} - I_S| = 1 - 1.000135 = 0.000135.$$

We conclude that the approximation of I by Simpson's rule is more accurate than the trapezoidal rule.

Solution Exercise 2

1.
$$I = \int_0^{\pi} \sin(x)\,dx = [-\cos(x)]_0^{\pi} = 2.$$

2. **I) We start with the Trapezoidal Rule:**

   (a)
   $$I_{\text{trap}} = \frac{h}{2}\big(f(x_0) + f(x_4) + 2(f(x_1) + f(x_2) + f(x_3))\big),$$
   with $x_0 = 0$, $x_1 = \frac{\pi}{4}$, $x_2 = \frac{\pi}{2}$, $x_3 = \frac{3\pi}{4}$, and $x_4 = \pi$. Thus:
   $$I_{\text{trap}} = \frac{\pi}{8}\big(f(0) + f(\pi) + 2(f(\tfrac{\pi}{4}) + f(\tfrac{\pi}{2}) + f(\tfrac{3\pi}{4}))\big) = \frac{\pi}{4}(1 + \sqrt{2}) \approx 1.896.$$

   (b) We have:
   $$E = -\frac{h^2}{12}(x_n - x_0)f''(\xi), \quad 0 \le \xi \le \pi.$$
   Hence:
   $$|E| = \frac{\pi^3}{192}|\sin(\xi)| \le \frac{\pi^3}{192} \approx 0.16149, \quad 0 \le \xi \le \pi.$$

(c) The relative error is:

$$E_{\text{rel}} = I_{\text{exact}} - I_{\text{trap}} \approx 0.1038.$$

II) Simpson's Method

(a)

$$I_{\text{Simp}} = \frac{\pi}{12}\Big(f(0) + f(\pi) + 2f\big(\tfrac{\pi}{2}\big) + 4\big(f(\tfrac{\pi}{4}) + f(\tfrac{3\pi}{4})\big)\Big) = \frac{\pi}{6}(1 + 4\sqrt{2}) \approx 3.4855207.$$

(b) We have:

$$E = -\frac{h^5}{90} f^{(4)}(\xi), \quad 0 \le \xi \le \pi.$$

Thus:

$$|E| = \frac{h^5}{90}|\sin(\xi)| \le \frac{h^5}{90} \approx 0.00033205, \quad 0 \le \xi \le \pi.$$

(c) The relative error is:

$$E_{\text{rel}} = I_{\text{exact}} - I_{\text{Simp}} \approx 1.4855207.$$

3. Let us compute the step size h and the number of subdivisions n of the interval $[0, \pi]$ for $\varepsilon = 5 \times 10^{-4}$.

Trapezoidal Method

We have:

$$E = -\frac{h^2}{12}(x_n - x_0) f''(\xi), \quad 0 \le \xi \le \pi.$$

Thus:

$$|E| = \frac{\pi}{12} h^2 |\sin(\xi)| \le \frac{\pi}{12} h^2 \approx 0.0005, \quad 0 \le \xi \le \pi.$$

Hence:

$$h^2 \le \frac{12}{\pi} \cdot 0.0005, \quad h \le 0.044.$$

Therefore:

$$n > \frac{\pi}{h} \approx 71.8.$$

Thus, the number of subdivisions of the interval $[0, \pi]$ is $n \ge 72$.

Simpson's Method

We have:

$$E = -\frac{\pi}{90} h^4 f^{(4)}(\xi), \quad 0 \le \xi \le \pi.$$

Thus:

$$|E| = \frac{\pi}{90} h^4 |\sin(\xi)| \le \frac{\pi}{90} h^4 \approx 0.0005, \quad 0 \le \xi \le \pi.$$

Hence:

$$h^4 \le \frac{90}{\pi} \cdot 0.0005, \quad h \le 0.4111408.$$

Therefore:

$$n > \frac{\pi}{h} \approx 7.64116.$$

Thus, the number of subdivisions of the interval $[0, \pi]$ with Simpson's method is $n \geq 8$.

CHAPTER 3

POLYNOMIAL INTERPOLATION

For example, consider an experiment in which the distance traveled by an object is recorded as a function of time. The results are given in the following table:

t (sec)	0	1	2	3	4
X (m)	0	5	15	0	3

We want, for instance, to calculate the position of the object at time $t = 2.5$ seconds, or the velocity of the object at a given time. For this purpose, we need an analytical expression of x as a function of t, namely $X(t)$. This function must at least match the points given in the table. Then we can compute $X(2.5)$, $\int_0^4 X(t)\,dt$, or the velocity $v(t) = \dfrac{dX(t)}{dt}$.

In this chapter, we will consider approximating $X(t)$ by a polynomial expression, that is:

$$X(t) = a_0 + a_1 t + a_2 t^2 + \cdots + a_n t^n,$$

where the coefficients a_i ($i = 0, \ldots, n$) are to be determined.

The polynomials we will study differ only in the way the coefficients a_i are computed. For any given table of values, the interpolation polynomial is unique.

3.1 Interpolation or Approximation

Definition 3.1 (**Interpolation**)**.** Unlike interpolation, approximating a function does not require the sought curve (of the approximated function) to pass through the points (x_i, y_i), but it requires that a certain approximation criterion be satisfied (e.g., least squares criterion).

Definition 3.2 (**Approximation**). Unlike interpolation, approximating a function does not require the sought curve (of the approximated function) to pass through the points (x_i, y_i), but it requires that a certain approximation criterion be satisfied (e.g., least squares criterion).

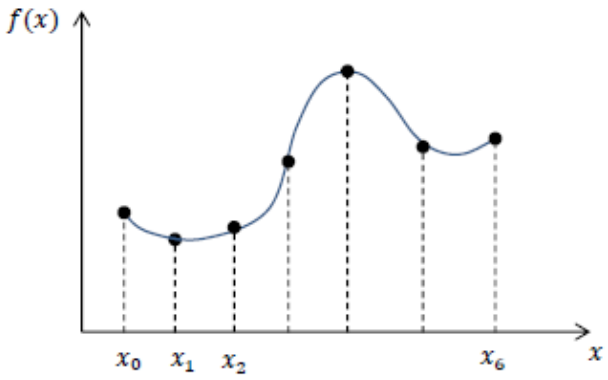


Figure 3.1: Polynomial interpolation.

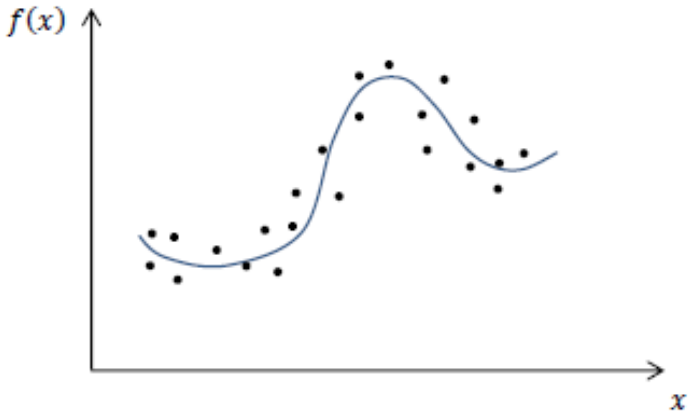


Figure 3.2: Function approximation.

3.2 Lagrange Interpolation Polynomial

Let $(n+1)$ distinct points $x_0, x_1, \ldots, x_n$ be given, and let f be a function whose values at these points are $f(x_0), f(x_1), \ldots, f(x_n)$. Then there exists a unique polynomial of degree less than or equal to n that coincides with the interpolation points, i.e.,

$$f(x_k) = P_n(x_k), \qquad k = 0, 1, 2, \ldots, n.$$

This polynomial is given by:

$$P_n(x) = \sum_{i=0}^{n} f(x_i)\, L_i(x) = f(x_0)L_0(x) + f(x_1)L_1(x) + \cdots + f(x_n)L_n(x),$$

where

$$L_k(x) = \prod_{\substack{i=0 \\ i\neq k}}^{n} \frac{x - x_i}{x_k - x_i} = \frac{x - x_0}{x_k - x_0} \cdot \frac{x - x_1}{x_k - x_1} \cdots \frac{x - x_{k-1}}{x_k - x_{k-1}} \cdot \frac{x - x_{k+1}}{x_k - x_{k+1}} \cdots \frac{x - x_n}{x_k - x_n}, \qquad (k = 0, \ldots, n).$$

The functions $L_k(x)$ are called the Lagrange basis polynomials. They are orthogonal in the sense that:

$$L_k(x_j) = 0 \quad \text{for } j \neq k, \qquad L_k(x_k) = 1.$$

3.2.1 Example:

Let us take again the table given at the beginning of the chapter and try to compute the Lagrange interpolation polynomial for this table. Note that for $(n+1)$ points, the degree of the interpolation polynomial is less than or equal to n. In our case, we have 5 points, which gives a polynomial of degree at most 4.

$$X(t) \approx P_4(t) = \sum_{i=0}^{4} f(t_i)\, L_i(t) = f(t_0)L_0(t) + f(t_1)L_1(t) + f(t_2)L_2(t) + f(t_3)L_3(t) + f(t_4)L_4(t).$$

Since the coefficients $f(t_i)$ are simply the values of $X(t_i)$ at the given points, we substitute them to obtain:

$$X(t) \approx P_4(t) = 0 \cdot L_0(t) + 5 \cdot L_1(t) + 15 \cdot L_2(t) + 0 \cdot L_3(t) + 3 \cdot L_4(t).$$

Next, we compute the Lagrange basis polynomials.

$$L_0(t) = \prod_{\substack{i=0 \\ i\neq 0}}^{4} \frac{t - t_i}{t_0 - t_i}.$$

Note that it is unnecessary to compute $L_0(t)$ and $L_3(t)$ because they will be multiplied by zero in the interpolation formula.

$$L_1(t) = \prod_{\substack{i=0 \\ i\neq 1}}^{4} \frac{t - t_i}{t_1 - t_i} = \frac{t-0}{1-0} \cdot \frac{t-2}{1-2} \cdot \frac{t-3}{1-3} \cdot \frac{t-4}{1-4} = -\frac{1}{6}\left(t^4 - 9t^3 + 26t^2 - 24t\right).$$

$$L_2(t) = \prod_{\substack{i=0 \\ i\neq 2}}^{4} \frac{t - t_i}{t_2 - t_i} = \frac{t-0}{2-0} \cdot \frac{t-1}{2-1} \cdot \frac{t-3}{2-3} \cdot \frac{t-4}{2-4} = \frac{1}{4}\left(t^4 - 8t^3 + 19t^2 - 12t\right).$$

$$L_4(t) = \prod_{\substack{i=0 \\ i\neq 4}}^{4} \frac{t - t_i}{t_4 - t_i} = \frac{t-0}{4-0} \cdot \frac{t-1}{4-1} \cdot \frac{t-2}{4-2} \cdot \frac{t-3}{4-3} = \frac{1}{24}\left(t^4 - 6t^3 + 11t^2 - 6t\right).$$

Finally, substituting the computed Lagrange polynomials, we obtain:

$$X(t) \approx P_4(t) = -25.75\, t + 50.95833\, t^2 - 23.25\, t^3 + 3.04167\, t^4.$$

3.3 Newton Interpolation Polynomial

We have seen that the Lagrange polynomial uses $(n+1)$ polynomial coefficients, which are themselves polynomials of degree less than or equal to n. Computing these polynomial coefficients is also a delicate task, which is why it is useful to employ a more flexible formulation: the Newton interpolation polynomial.

The construction of the Newton polynomial begins with a polynomial of degree 1, $P_1(x)$, which passes through the first two points. This polynomial is then used to compute another of degree 2, $P_2(x)$, which passes through the first three points, and so on, until the final polynomial of degree less than or equal to n, $P_n(x)$.

We have the following recurrence relation between two successive polynomials, $P_{i-1}(x)$ and $P_i(x)$ for $i = 2, 3, \ldots, n+1$:

$$\begin{cases} P_1(x) = a_0 + a_1(x - x_0), \\ P_2(x) = P_1(x) + a_2(x - x_0)(x - x_1), \\ P_3(x) = P_2(x) + a_3(x - x_0)(x - x_1)(x - x_2), \\ \quad \vdots \\ P_n(x) = P_{n-1}(x) + a_n(x - x_0)(x - x_1)\cdots(x - x_{n-1}). \end{cases}$$

We observe that the coefficients a_k $(k = 0, \ldots, n)$ are the essential elements in the computation of Newton polynomials. These coefficients are the *divided differences* of order k of the function f.

3.4 Computation of the Divided Differences of a Function f

The divided differences of a function f based on the points $x_0, x_1, \ldots, x_n$ are given by:

$$a_k = f[x_0, x_1, \ldots, x_k] = \sum_{i=0}^{k} \frac{y_i}{\prod_{\substack{j=0 \\ j \neq i}}^{k} (x_i - x_j)}.$$

In practice, for a limited number of points, the divided differences are computed using a table of the following form:

x_k	$f(x_k) = f[x_k]$	DD1	DD2	DD3	DD4
x_0	$f[x_0]$				
x_1	$f[x_1]$	$f[x_0, x_1]$			
x_2	$f[x_2]$	$f[x_1, x_2]$	$f[x_0, x_1, x_2]$		
x_3	$f[x_3]$	$f[x_2, x_3]$	$f[x_1, x_2, x_3]$	$f[x_0, x_1, x_2, x_3]$	
x_4	$f[x_4]$	$f[x_3, x_4]$	$f[x_2, x_3, x_4]$	$f[x_1, x_2, x_3, x_4]$	

With:

$$f[x_0, x_1] = \frac{f[x_1] - f[x_0]}{x_1 - x_0}, \qquad f[x_1, x_2] = \frac{f[x_2] - f[x_1]}{x_2 - x_1}, \quad \dots$$

$$f[x_0, x_1, x_2] = \frac{f[x_1, x_2] - f[x_0, x_1]}{x_2 - x_0}, \qquad f[x_1, x_2, x_3] = \frac{f[x_2, x_3] - f[x_1, x_2]}{x_3 - x_1}, \quad \dots$$

$$f[x_0, x_1, x_2, x_3] = \frac{f[x_1, x_2, x_3] - f[x_0, x_1, x_2]}{x_3 - x_0}, \qquad f[x_1, x_2, x_3, x_4] = \frac{f[x_2, x_3, x_4] - f[x_1, x_2, x_3]}{x_4 - x_1},$$

$$f[x_0, x_1, x_2, x_3, x_4] = \frac{f[x_1, x_2, x_3, x_4] - f[x_0, x_1, x_2, x_3]}{x_4 - x_0}.$$

3.5 Interpolation Error

This is the error made when replacing the function f by the corresponding interpolation polynomial. It is denoted by $\varepsilon(x)$ because it varies from one point to another within the interpolation interval. This error must be zero at the interpolation points, i.e., $\varepsilon(x_i) = 0$ for $i = 0, \dots, n$.

If the function f is continuous and $(n + 1)$ times differentiable on the interpolation interval $[a = x_0, b = x_n]$, then for every $x \in [a, b]$ there exists $z \in [a, b]$ such that:

$$\varepsilon(x) = |f(x) - P_n(x)| = \frac{\prod_{i=0}^{n}(x - x_i)}{(n+1)!} f^{(n+1)}(z)$$

If $|f^{(n+1)}(z)| \leq M \quad \forall z \in [a, b]$, we can write:

$$\varepsilon(x) = |f(x) - P_n(x)| \leq \frac{\prod_{i=0}^{n}(x - x_i)}{(n+1)!} M$$

In this case, M is an upper bound for the function $f^{(n+1)}(x)$ on the interval $[a, b]$.

3.5.1 Example:

Let us take again the table given at the beginning of the chapter and try to compute the Newton interpolation polynomial for this table. Note that for $n + 1$ points, the degree of the polynomial is less than or equal to n. In our case, we have 5 points, which gives a polynomial of degree less than or equal to 4. Let us write the Newton polynomials:

$$\begin{cases} P_1(t) = a_0 + a_1(t - t_0) \\ P_2(t) = P_1(t) + a_2(t - t_0)(t - t_1) \\ P_3(t) = P_2(t) + a_3(t - t_0)(t - t_1)(t - t_2) \\ P_4(t) = P_3(t) + a_4(t - t_0)(t - t_1)(t - t_2)(t - t_3) \end{cases}$$

The coefficients a_i are the divided differences of order i.

Let us compute the divided difference table:

t_k	$f(t_k) = f[t_k]$	DD1	DD2	DD3	DD4
0	$0 = a_0$				
1	5	$5 = a_1$			
2	15	10	$2.5 = a_2$		
3	0	-15	-12.5	$-5 = a_3$	
4	3	3	9	7.1667	$3.04167 = a_4$

By substituting the a_i and t_i values into the Newton polynomials, we obtain:

$$\begin{aligned} P_1(t) &= a_0 + a_1(t - t_0) = 0 + 5(t - 0) = 5t \\ P_2(t) &= P_1(t) + a_2(t - t_0)(t - t_1) = 5t + 2.5(t - 0)(t - 1) = 2.5t^2 + 2.5t \\ P_3(t) &= P_2(t) + a_3(t - t_0)(t - t_1)(t - t_2) = 2.5t^2 + 2.5t - 5(t - 0)(t - 1)(t - 2) \\ &= -5t^3 + 17.5t^2 - 7.5t \\ P_4(t) &= P_3(t) + a_4(t - t_0)(t - t_1)(t - t_2)(t - t_3) \\ &= -5t^3 + 17.5t^2 - 6t + 3.0417(t - 0)(t - 1)(t - 2)(t - 3) \\ &= 3.04167t^4 - 23.25t^3 + 50.95833t^2 - 25.75t \end{aligned}$$

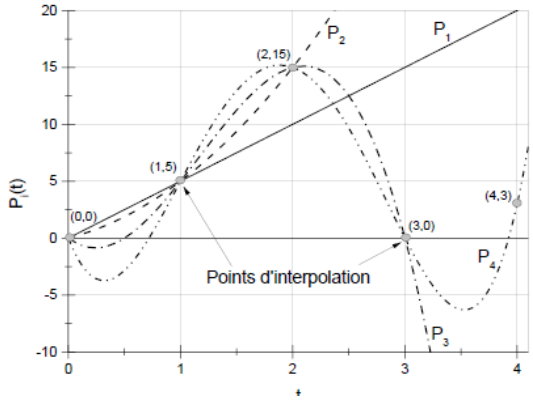


Figure 3.3: Plots of the Newton interpolation polynomials.

CHAPTER 4

NUMERICAL SOLUTION OF ORDINARY DIFFERENTIAL EQUATIONS

Differential equations are among the most important mathematical tools used in modeling problems in the physical sciences. Finding the solution of an ordinary differential equation (ODE) is a common problem, often difficult or impossible to solve analytically. It is therefore necessary to use numerical methods to solve such equations.

The accuracy of the various solution methods presented in this course is proportional to the order of these methods. We begin with the simple Euler method (which has a geometric interpretation) that will gradually lead us to more complex methods such as the fourth-order Runge–Kutta methods, which allow us to obtain highly accurate results.

In this chapter, we focus on first-order differential equations of the form:

$$y'(t) = f(t, y(t)).$$

The Cauchy problem (initial value problem) consists in finding a function $y(t)$ defined on the interval $I = [a, b]$ such that:

$$\begin{cases} \frac{dy(t)}{dt} = f(t, y(t)), \quad t \in I,\ y(t) \in \mathbb{R}, \\ y(t_0) = y_0, \quad t_0 \in I,\ y_0 \in \mathbb{R}. \end{cases}$$

The condition $y(t_0) = y_0$ is called an initial condition or Cauchy condition.

If we assume that the function f is continuous with respect to both variables t, y, and that f satisfies a Lipschitz condition with respect to y, that is:

$$\exists K > 0 \text{ such that } \forall t \in [a, b],\ \forall y_1, y_2 \in \mathbb{R}, \quad |f(t, y_1) - f(t, y_2)| \leq K|y_1 - y_2|,$$

then, in practice, to verify this condition, one computes:

$$K = \max_{t\in[a,b],\, y\in\mathbb{R}} \left| \frac{\partial f(t,y)}{\partial y} \right|.$$

Under these assumptions, the solution of the Cauchy problem exists and is unique on $[a, b]$ (this is the Cauchy theorem).

4.1 Numerical Methods

4.1.1 General Principle

Consider the following initial value Cauchy problem:

$$\begin{cases} y'(t) = f(t, y(t)), \\ y(t_0) = y_0. \end{cases}$$

We assume that this problem admits a unique solution on the interval $[a, b]$.

To obtain a numerical approximation of this solution $y(t)$ on the interval $[a, b]$, we divide the given interval into n equally spaced points $t_0, t_1, t_2, \ldots, t_n$, with $t_0 = a$ and $t_n = b$, which determines the integration step

$$h = \frac{b-a}{n},$$

and the abscissa of point t_i is given by

$$t_i = a + ih, \quad (i = 0, 1, \ldots, n).$$

The exact solution at point t_i is denoted by $y(t_i)$, while the approximate solution is denoted by

$$y_i \approx y(t_i).$$

A numerical method is a system of difference equations involving a certain number of successive approximations $y_n, y_{n+1}, \ldots, y_{n+k}$, where k denotes the number of steps of the method.

- If $k = 1$, we speak of a **single-step method**: these allow the computation of y_{n+1} using only the value of y_n. - If $k > 1$, we speak of a **multi-step method**: these allow the computation of y_{n+1} using the values $y_n, y_{n-1}, \ldots, y_{n-k}$ (for a fixed k).

The solution to be estimated can be approximated using a Taylor series expansion. The Taylor expansion of $y(t_{n+1})$ up to order m around the point t_n is written as:

$$y(t_{n+1}) = y(t_n) + hy'(t_n) + \frac{h^2}{2!}y''(t_n) + \cdots + \frac{h^m}{m!}y^{(m)}(t_n) + O(h^{m+1}).$$

4.1.2 Euler's Method

Euler's method is the oldest and simplest of the numerical methods for approximating first-order ordinary differential equations. If the functions $y(t)$ and $y'(t)$ are continuous, we can write the Taylor series expansion of $y(t)$ in the neighborhood of t_n:

$$y(t_{n+1}) = y(t_n) + h\, y'(t_n) + O(h^2).$$

The term $O(h^2)$ represents the error term of order 2. Neglecting the second-order terms, we can rewrite this equation as:

$$y(t_{n+1}) \approx y(t_n) + h\, y'(t_n).$$

Let y_i be an approximation of $y(t_i)$, i.e., $y_i \approx y(t_i)$. The first-order Euler method is written as:

$$\begin{cases} y_0 = y(t_0), \\ y_{i+1} = y_i + h\, f(t_i, y_i), \quad i = 0, 1, 2, \ldots \end{cases}$$

4.1.2.1 Geometric Interpretation

The Euler method consists of locally replacing, at each point t_i, the solution curve passing through y_i by its tangent.

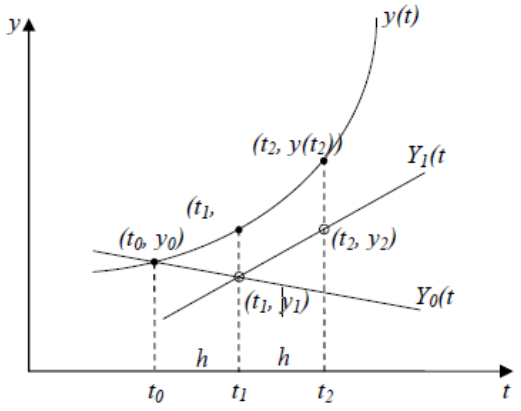


Figure 4.1: Euler's Method.

The tangent to the curve $y = y(t)$ at $t = t_0$ has the equation:

$$Y(t) = y(t_0) + y'(t_0)(t - t_0).$$

Since $y'(t_0) = f(t_0, y(t_0))$, we can write:

$$Y(t) = y(t_0) + f(t_0, y(t_0))(t - t_0).$$

Or equivalently:

$$Y(t) = y_0 + f(t_0, y_0)(t - t_0).$$

At the point $t = t_1$:

$$Y(t_1) = y_0 + f(t_0, y_0)(t_1 - t_0).$$

Since $t_1 - t_0 = h$, this becomes:

$$Y(t_1) = y_0 + hf(t_0, y_0).$$

Approximating $Y(t_1) \approx y_1$, we obtain:

$$y_1 = y_0 + hf(t_0, y_0).$$

Similarly, considering the tangent line at $t = t_1$:

$$Y(t) = y(t_1) + y'(t_1)(t - t_1).$$

At $t = t_2$:

$$Y(t_2) = y_1 + f(t_1, y_1)(t_2 - t_1).$$

Thus:

$$Y(t_2) = y_1 + hf(t_1, y_1).$$

By proceeding in the same way, we obtain the recurrence relation:

$$y_{i+1} = y_i + hf(t_i, y_i).$$

Example 4.1. Consider the initial value problem:

$$y'(t) = -y(t) + t + 1, \quad y(0) = 1.$$

1. Approximate the solution of this equation at $t = 0.5$ using Euler's method, by subdividing the interval into 5 equal parts (perform the computations with six decimals).
2. Compare the obtained results with the exact solution:

$$y(t) = e^{-t} + t.$$

Solution:

We have:

$$y'(t) = -y(t) + t + 1, \quad y(0) = 1, \quad n = 5, \quad \text{and the integration interval is } [0, 0.5].$$

Let us verify the Lipschitz condition on $t \in [0, 0.5]$ and $y \in \mathbb{R}$:

Notice that the function

$$f(t, y) = -y(t) + t + 1$$

is continuous on $[0, 0.5] \times \mathbb{R}$. Moreover, it is Lipschitz with respect to y on $[0, 0.5]$ with Lipschitz constant $K = 1$. Indeed, for all $t \in [0, 0.5]$ and $y_1, y_2 \in \mathbb{R}$, we have:

$$|f(t, y_1) - f(t, y_2)| = | - y_1 + t + 1 - (-y_2 + t + 1)| = | - y_1 + y_2| = |y_1 - y_2| \leq 1 \cdot |y_1 - y_2|.$$

Also,

$$\frac{\partial f}{\partial y}(t, y) = -1, \quad \Rightarrow \quad \max \left| \frac{\partial f}{\partial y} \right| = 1 = K.$$

Condition verified. Hence, the Cauchy problem admits a unique solution.

Computation of the approximate solution using Euler's method: We have:

$$f(t, y) = -y(t) + t + 1, \quad t_0 = 0, \quad y_0 = y(0) = 1, \quad h = \frac{b - a}{n} = \frac{0.5 - 0}{5} = 0.1.$$

Euler's method is given by:

$$\begin{cases} t_i = t_0 + ih, \quad i = 0, 1, \ldots, 5, \\ y_{i+1} = y_i + hf(t_i, y_i) = y_i + h\left(-y_i + t_i + 1\right). \end{cases}$$

Using this method, we successively obtain approximations of $y(0.1), y(0.2), y(0.3), \ldots$, denoted $y_1, y_2, y_3, \ldots$.

$$i = 0: \quad y_1 = y_0 + h(-y_0 + t_0 + 1) = 1 + 0.1(-1 + 0 + 1) = 1.000000,$$

$$i = 1: \quad y_2 = y_1 + h(-y_1 + t_1 + 1) = 1.000000 + 0.1(-1.000000 + 0.1 + 1) = 1.010000.$$

The following table summarizes the results of the first five iterations.

Exact solution: The analytical solution of this differential equation is:

$$y(t) = e^{-t} + t.$$

This allows us to compare the numerical (approximate) and analytical (exact) solutions and to observe the growth of the error.

Thus, the approximation of $y(t)$ at $t = 0.5$ using Euler's method is:

$$y_5 = 1.090490$$

i	t_i	y_i (Approximate Solution)	$y(t_i)$ (Exact Solution)	$\lvert y_i - y(t_i)\rvert$
0	0.0	1.000000	1.000000	0.000000
1	0.1	1.000000	1.004837	0.004837
2	0.2	1.010000	1.018731	0.008731
3	0.3	1.029000	1.040818	0.011818
4	0.4	1.056100	1.070302	0.014220
5	0.5	1.090490	1.106531	0.016041

Table 4.1: Comparison between approximate and exact solutions using Euler's method.

4.1.3 Runge-Kutta Method

Runge-Kutta methods are generalizations of Euler's method to higher orders (higher accuracy). These methods provide greater precision (they generate numerical solutions closer to the analytical solutions) than Euler's method. These methods are of the form:

$$y_{i+1} = y_i + b_1k_1 + b_2k_2 + b_3k_3 + \cdots$$

where

$$k_1 = hf(x_i, y_i),$$

$$k_2 = hf(x_i + c_2h,\, y_i + c_2k_1)\,,$$

$$k_3 = hf(x_i + c_3h,\, y_i + a_{31}k_1 + a_{32}k_2)\,.$$

$$\vdots$$

4.1.3.1 Second-Order Runge-Kutta Method: RK2

The second-order Runge-Kutta methods are of the form:

$$k_1 = hf\left(t_i + c_1h,\, y_i + c_1k_1\right),$$

$$k_2 = hf\left(t_i + c_2h,\, y_i + c_2k_2\right),$$

$$y_{i+1} = y_i + b_1k_1 + b_2k_2,$$

where b_1, b_2, c_1, c_2, k_1, and k_2 are determined.

The computation of the values b_1, b_2, c_1, c_2, k_1, and k_2 requires the calculation of $y^{(2)}, y^{(3)}$, and $y^{(4)}$ from $y' = f(x, y)$ and the Taylor expansion. We obtain:

$$k_1 = hf(t_i, y_i),$$

$$k_2 = hf(t_i + h,\, y_i + k_1),$$

$$y_{i+1} = y_i + \frac{1}{2}(k_1 + k_2).$$

The numerical scheme of the second-order Runge-Kutta method is given as follows:

$$\begin{cases} y_0 = y(t_0) \quad \text{given}, \\ K_1 = f(t_{i-1}, y_{i-1}), \quad K_2 = f(t_{i-1} + h,\, y_{i-1} + hK_1), \\ y_i = y_{i-1} + \dfrac{h}{2}(K_1 + K_2), \quad i = 1, \ldots, n. \end{cases}$$

Example 4.2. Solve the previous initial value problem using both the RK2 and RK4 methods to compute the first five values of the solution, taking the integration step $h = 0.1$.

Compare the numerical results with the exact solution.

Solution: We have: $f(t, y) = -y(t) + t + 1,\ y(0) = 1$ and $h = 0.1$.

The RK2 method is given by:

$$\begin{cases} k_1 = hf(t_i, y_i), \\ k_2 = hf(t_i + h, y_i + k_1), \\ y_{i+1} = y_i + \dfrac{k_1 + k_2}{2}. \end{cases}$$

Using this method, we obtain the successive approximations $y_1, y_2, \ldots, y_5$.

For $i = 0$:

$$k_1 = 0.1f(0, 1) = 0.1(-1 + 0 + 1) = 0,$$

$$k_2 = 0.1f(0.1, 1 + 0) = 0.1(-1 + 0.1 + 1) = 0.01,$$

$$y_1 = 1 + \frac{0 + 0.01}{2} = 1.005.$$

For $i = 1$:

$$k_1 = 0.1f(0.1, 1.005) = 0.1(-1.005 + 0.1 + 1) = 0.0095,$$

$$k_2 = 0.1f(0.2, 1.005 + 0.0095) = 0.1(-1.0145 + 0.2 + 1) = 0.02855,$$

$$y_2 = 1.005 + \frac{0.0095 + 0.02855}{2} = 1.019025.$$

The following table summarizes the results of the first five iterations:

i	t_i	y_i (approximate solution)	$y(t_i)$ (exact solution)	Error
0	0.0	1.000000	1.000000	0.000000
1	0.1	1.005000	1.004837	0.000163
2	0.2	1.019025	1.018731	0.000294
3	0.3	1.041218	1.040818	0.000400
4	0.4	1.070802	1.070302	0.000482
5	0.5	1.107075	1.106531	0.000544

Thus, the approximation of $y(t)$ at $t = 0.5$ using the RK2 method is $y_5 = 1.107075$.

We observe that the error is smaller with the RK2 method than with the Euler method.

4.1.3.2 Runge-Kutta of order 4: RK4

This is the most accurate and most widely used method, of order 4. It computes the value of the function at four intermediate points according to:

$$\begin{cases} k_1 = hf(t_i, y_i), \\ k_2 = hf\left(t_i + \dfrac{h}{2}, y_i + \dfrac{k_1}{2}\right), \\ k_3 = hf\left(t_i + \dfrac{h}{2}, y_i + \dfrac{k_2}{2}\right), \\ k_4 = hf(t_i + h, y_i + k_3), \\ y_{i+1} = y_i + \dfrac{k_1 + 2k_2 + 2k_3 + k_4}{6}. \end{cases}$$

The numerical scheme of the Runge-Kutta of order 4 method is given as follows:

$$\begin{cases} y_0 = y(t_0) \quad \text{given}, \\ K_1 = f(t_i, y_i), \\ K_2 = f\left(t_i + \frac{h}{2},\ y_i + \frac{h}{2}K_1\right), \\ K_3 = f\left(t_i + \frac{h}{2},\ y_i + \frac{h}{2}K_2\right), \\ K_4 = f\left(t_i + h,\ y_i + hK_3\right), \\ y_{i+1} = y_i + \dfrac{h}{6}\left(K_1 + 2K_2 + 2K_3 + K_4\right). \end{cases}$$

Example 4.3. Solve the previous initial value problem using both the RK4 and RK4 methods to compute the first five values of the solution, taking the integration step $h = 0.1$.

Compare the numerical results with the exact solution.

Solution:

The RK4 method is given by:

$$\begin{aligned} k_1 &= hf(t_i, y_i), \\ k_2 &= hf\left(t_i + \tfrac{h}{2},\, y_i + \tfrac{k_1}{2}\right), \\ k_3 &= hf\left(t_i + \tfrac{h}{2},\, y_i + \tfrac{k_2}{2}\right), \\ k_4 &= hf(t_i + h,\, y_i + k_3), \\ y_{i+1} &= y_i + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4). \end{aligned}$$

- For $i = 0$:

$$\begin{cases} k_1 = hf(t_0, y_0) = 0, \\ k_2 = hf\left(t_0 + \frac{h}{2}, y_0 + \frac{k_1}{2}\right) = 0.005, \\ k_3 = hf\left(t_0 + \frac{h}{2}, y_0 + \frac{k_2}{2}\right) = 0.00475, \\ k_4 = hf(t_0 + h, y_0 + k_3) = 0.00952. \end{cases}$$

Thus:

$$y_1 = y_0 + \tfrac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) = 1.004837.$$

- For $i = 1$:

$$\begin{cases} k_1 = hf(t_1, y_1) = 0.09516, \\ k_2 = hf\left(t_1 + \frac{h}{2}, y_1 + \frac{k_1}{2}\right) = 0.0140404, \\ k_3 = hf\left(t_1 + \frac{h}{2}, y_1 + \frac{k_2}{2}\right) = 0.013814, \\ k_4 = hf(t_1 + h, y_1 + k_3) = 0.018730. \end{cases}$$

Thus:

$$y_2 = y_1 + \tfrac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) = 1.01873.$$

The following table gathers the results of the first five iterations:

i	t_i	y_i (approx. solution)	$y(t_i)$ (exact solution)	Error
0	0.0	1.000000	1.000000	0.000000
1	0.1	1.004837	1.004837	0.819×10^{-7}
2	0.2	1.018730	1.018731	0.148×10^{-6}
3	0.3	1.040818	1.040818	0.210×10^{-6}
4	0.4	1.070320	1.070302	0.242×10^{-6}
5	0.5	1.106530	1.106531	0.274×10^{-6}

Thus, the approximation of $y(t)$ at $t = 0.5$ using the RK4 method is $y_5 = 1.106530$.

We observe that the error is of the order of 10^{-6}, which compares favorably with the errors obtained using lower-order methods (Euler, second-order Runge-Kutta).

4.2 Exercises

Exercise 4.1. Let us consider the initial value problem (IVP):

$$\begin{cases} y'(t) = y(t) + t, \\ y(0) = 1. \end{cases}$$

1. Compute the approximate solution of this equation at $t = 1$ using Euler's method, by subdividing the interval $[0, 1]$ into 10 equal parts.

2. Knowing that the exact solution is

$$y(t) = 2e^t - t - 1,$$

compare the obtained result with the exact solution.

Exercise 4.2. Let us consider the following Cauchy problem:

$$\begin{cases} y'(t) = y(t) - t^2, \\ y(0) = 1. \end{cases}$$

1. Compute the approximate solution of this equation at $t = 0.2$ using the second-order Runge-Kutta method (RK2) with step size $h = 0.2$.

2. Knowing that the exact solution is

$$y(t) = 2e^t + t + 1,$$

compare the obtained result with the exact solution.

Exercise 4.3. Let us consider the following Cauchy problem:

$$\begin{cases} y'(t) = e^{-t} - y(t), \quad t \in [0,1], \\ \\ y(0) = 1. \end{cases}$$

1. Show that $f(t,y)$ is Lipschitz continuous with respect to y, uniformly in t, and provide a Lipschitz constant.
2. Prove that this problem admits a unique solution.
3. Determine the exact solution of this problem as well as the value of $y(0.2)$.
4. Apply Euler's method to this problem and compute the approximation y_2 of $y(0.2)$ using the discretization step $h = 0.1$.
5. Compare the obtained result with the exact solution.

4.3 Solutions to the Exercises

Solution to Exercise 1

1. Compute the approximate solution of the differential equation at $t = 1$ using Euler's method:

We have:

$$f(t,y) = y + t.$$

The interval is $[0,1] \implies h = \frac{1-0}{10} = 0.1$.

Euler's method is given by:

$$\begin{cases} t_i = t_0 + ih, \quad y_0 = 1, \\ \\ y_{i+1} = y_i + hf(t_i, y_i) = y_i + h(y_i + t_i), \quad i = 0, 1, \ldots, 10. \end{cases}$$

Using this method, we successively obtain approximations of $y(0.1), y(0.2), y(0.3), \ldots$, denoted by $y_1, y_2, y_3, \ldots$.

For $i = 0$:

$$y_1 = y_0 + h(y_0 + t_0) = 1 + 0.1(1 + 0) = 1.1.$$

For $i = 1$:

$$y_2 = y_1 + h(y_1 + t_1) = 1.1 + 0.1(1.1 + 0.1) = 1.22.$$

The following table summarizes the results of the first ten iterations:

i	t_i	y_i
0	0.0	1.0000
1	0.1	1.1000
2	0.2	1.2200
3	0.3	1.3620
4	0.4	1.5282
5	0.5	1.7210
6	0.6	1.9431
7	0.7	2.1974
8	0.8	2.4871
9	0.9	2.8158
10	1.0	3.1874

Therefore, the approximation of $y(t)$ at $t = 1$ by Euler's method is $y_{10} = y(1) \approx 3.1874$. 2/ Comparison of the obtained result with the exact solution: The exact solution is: $y(t) = 2e^{-t} - 1$

$$y(1) = 2e^{-1} - 1 = 0.7358 \quad \text{(exact)}$$

Therefore, the error is:

$$E = y_{\text{exact}}(1) - y_{\text{approx}}(1) = 0.7358 - 0.4867 = 0.2491$$

Solution to Exercise 2

1/ Computation of the approximate solution of the differential equation at $t = 1$ using the Runge-Kutta method of order 2:

We have:

$$f(t, y) = -\tfrac{1}{2}y, \quad y(0) = 1$$

The interval is $[0, 0.2]$, with step size $h = 0.2$.

The RK2 method is given by:

$$k_1 = hf(t_i, y_i), \quad k_2 = hf\left(t_i + h, y_i + k_1\right), \quad y_{i+1} = y_i + \tfrac{1}{2}(k_1 + k_2).$$

Using this method, we obtain the successive approximations $y_1, y_2, \ldots, y_5$.

For $i = 0$:

$$k_1 = hf(0, 1) = 0.2(-0.5 \times 1) = -0.1$$

$$k_2 = hf(0.2, 1 + (-0.1)) = 0.2(-0.5 \times 0.9) = -0.09$$

$$y_1 = y_0 + \frac{1}{2}(k_1 + k_2) = 1 + \frac{1}{2}(-0.1 - 0.09) = 1.1866$$

Thus, the approximation of $y(t)$ at $t = 0.2$ by the RK2 method is:

$$y_1 = y(0.2) \approx 1.1866.$$

2/ Comparison of the obtained result with the exact solution:

The exact solution is:

$$y(t) = 2e^{-t}$$

At $t = 0.2$, the exact solution is:

$$y(0.2) = 2e^{-0.2} = 1.1832 \quad \text{(exact)}.$$

Therefore, the error is:

$$E = y_{\text{approx}}(0.2) - y_{\text{exact}}(0.2) = 1.1866 - 1.1832 = 0.0034.$$

CHAPTER 5

DIRECT METHODS FOR SOLVING SYSTEMS OF LINEAR EQUATIONS

In this chapter, we present various direct methods for solving systems of linear equations. Direct methods are techniques that allow us to calculate the solution of the system

$$Ax = b$$

in a finite number of operations (the solution is exact).

Let's consider solving a linear system of m equations with n unknowns x_i, $i = 1, \ldots, n$:

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n &= b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n &= b_2 \\ &\vdots \\ a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n &= b_m \end{aligned} \tag{5.1}$$

This system of linear equations can be written in the following matrix form:

$$Ax = b \tag{5.2}$$

where A is a matrix of dimension $m \times n$, x is a vector of dimension $n \times 1$, and b is a vector of dimension $m \times 1$.

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix}, \quad x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}, \quad b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix}$$

Therefore, we have the matrix equation:

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix} \tag{5.3}$$

The problem is to find the values of the unknowns x_i, $i = 1, \dots, n$, that satisfy the m equations simultaneously. Three cases can arise depending on the relationship between the number of equations m and the number of unknowns n:

- $m > n$ (the number of equations is strictly greater than the number of unknowns): **no solution.**
- $m < n$ (the number of equations is strictly less than the number of unknowns): **an infinite number of solutions.**
- $m = n$ (the number of equations is equal to the number of unknowns): **a unique solution.**

5.1 Préliminaires

5.1.1 Cramer's System

A system of linear equations with an equal number of equations and unknowns ($m = n$) and with a non-zero determinant of the coefficient matrix ($|\mathbf{A}| \neq 0$) is referred to as a **Cramer's system**. Most physical linear systems fall into the category of Cramer's systems.

Remark 1: In this course, we are only concerned with Cramer's systems.

5.1.2 Homogeneous System

A system of linear equations in the form:

$$\mathbf{Ax} = 0, \quad \text{i.e. } \mathbf{b} = 0$$

is called a **homogeneous system** of linear equations. All homogeneous systems admit at least one solution where $x_i = 0$ ($i = 1, \dots, n$). This solution is known as the **null or trivial solution**.

5.2 Matrix Operations and Properties

- **Matrix:**

  A matrix of dimensions $n \times m$ (denoted $\mathbf{A}_{n\times m}$) is a table of numbers with n rows and m columns. The numbers that make up the matrix $(a_{ij},$ where $i = 1, \ldots, n$ and $j = 1, \ldots, m)$ are called **elements** or **coefficients** of the matrix.

$$\mathbf{A}_{n\times m} = [a_{ij}] = \begin{bmatrix} a_{11} & \ldots & a_{1m} \\ \vdots & \ddots & \vdots \\ a_{n1} & \ldots & a_{nm} \end{bmatrix}$$

- **Square Matrix:**

  A **square matrix** is a matrix that has the same number of rows and columns ($n = m$).

- **Matrix Diagonal:**

  (To be completed: definition of the diagonal of a matrix.)

- **Main Diagonal:**

  The **main diagonal** (or simply the diagonal) of a matrix $\mathbf{A}$ is the set of elements a_{ii} of the matrix.

- **Diagonal Matrix:**

  A **diagonal matrix** is a square matrix in which all elements outside the main diagonal are zero. The elements on the main diagonal may or may not be zero.

- **Identity Matrix (Unit Matrix):**

  The **identity matrix** is a diagonal matrix in which all elements on the main diagonal are equal to 1, denoted as $a_{ii} = 1$.

- **Addition and Subtraction of Matrices:**

  Only matrices of the same dimensions can be added or subtracted. Consider two matrices $\mathbf{A}$ and $\mathbf{B}$ of the same dimension:

$$\text{Addition: } \mathbf{C} = \mathbf{A} + \mathbf{B}, \quad c_{ij} = a_{ij} + b_{ij}$$

$$\text{Subtraction: } \mathbf{D} = \mathbf{A} - \mathbf{B}, \quad d_{ij} = a_{ij} - b_{ij}$$

- **Transpose of a Matrix:**

  The **transpose** of a matrix $\mathbf{A}$ of dimensions $n \times m$ is denoted by $\mathbf{A}^T$ and has dimensions $m \times n$. It is obtained by interchanging the rows and columns of the original matrix:

$$\mathbf{A}_{n\times m} = \begin{bmatrix} a_{11} & \ldots & a_{1m} \\ \vdots & \ddots & \vdots \\ a_{n1} & \ldots & a_{nm} \end{bmatrix} \quad \Rightarrow \quad \mathbf{A}^T_{m\times n} = \begin{bmatrix} a_{11} & \ldots & a_{n1} \\ \vdots & \ddots & \vdots \\ a_{1m} & \ldots & a_{nm} \end{bmatrix}$$

- **Scalar Multiplication of a Matrix:**

  The product of a matrix by a scalar is obtained by multiplying each element of the matrix by that scalar.

- **Matrix Multiplication:**

  Matrix multiplication of $\mathbf{A}_{n\times m}$ by $\mathbf{B}_{m\times p}$ is only possible if the number of columns of $\mathbf{A}$ equals the number of rows of $\mathbf{B}$. The resulting matrix $\mathbf{C}_{n\times p}$ has the same number of rows as $\mathbf{A}$ and the same number of columns as $\mathbf{B}$:

$$\mathbf{C}_{n\times p} = \mathbf{A}_{n\times m} \times \mathbf{B}_{m\times p}$$

5.2.1 Matrix Examples and Properties

5.2.1.1 Example 1: Matrix Multiplication

Consider two matrices $\mathbf{A}$ and $\mathbf{B}$:

$$\mathbf{A}_{2\times 3} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix}, \quad \mathbf{B}_{3\times 2} = \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \\ b_{31} & b_{32} \end{bmatrix}, \quad \mathbf{C}_{2\times 2} = \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix}$$

The elements of $\mathbf{C} = \mathbf{AB}$ are given by:

$$c_{11} = a_{11}b_{11} + a_{12}b_{21} + a_{13}b_{31}, \quad c_{12} = a_{11}b_{12} + a_{12}b_{22} + a_{13}b_{32},$$

$$c_{21} = a_{21}b_{11} + a_{22}b_{21} + a_{23}b_{31}, \quad c_{22} = a_{21}b_{12} + a_{22}b_{22} + a_{23}b_{32}.$$

The transpose of a product satisfies:

$$(\mathbf{AB})^T = \mathbf{B}^T\mathbf{A}^T$$

Matrix Trace The **trace** of a square matrix is the sum of the elements on its main diagonal:

$$\mathrm{tr}(\mathbf{A}) = \sum_i a_{ii}$$

Minors The **minor** M_{ij} of a matrix $\mathbf{A}$ is the determinant of the matrix obtained by removing the i-th row and j-th column from $\mathbf{A}$.

5.2.1.2 Example 2: Minors and Cofactors

Consider the matrix

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

Some minors of $\mathbf{A}$ are:

$$M_{12} = \begin{vmatrix} a_{22} & a_{23} \\ a_{32} & a_{33} \end{vmatrix}, \quad M_{22} = \begin{vmatrix} a_{21} & a_{23} \\ a_{31} & a_{33} \end{vmatrix}, \quad M_{32} = \begin{vmatrix} a_{11} & a_{13} \\ a_{21} & a_{23} \end{vmatrix}$$

Cofactor The **cofactor** D_{ij} of $\mathbf{A}$ is defined by:

$$D_{ij} = (-1)^{i+j} M_{ij}$$

Remark: The minor and cofactor have the same absolute value; the difference is only in the sign.

Determinant of an n-th Order Matrix Let $\mathbf{A}$ be a square matrix and D_{ij} its cofactors. The determinant of $\mathbf{A}$ can be computed using the cofactor expansion:

$$\det(\mathbf{A}) = \sum_{j=1}^{n} a_{1j} D_{1j} = a_{11} D_{11} + a_{12} D_{12} + \cdots + a_{1n} D_{1n}$$

1. Choose a row or column of $\mathbf{A}$. To simplify calculations, select the row or column with the highest number of zeros.
2. Multiply each element a_{ij} of the chosen row or column by its corresponding cofactor D_{ij}.
3. Sum up all these products:

$$\det(\mathbf{A}) = \sum_{j \in \text{row/column}} a_{ij} D_{ij}.$$

5.2.1.3 Example 3: Determinant Calculation

Calculate the determinant of the matrix

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}.$$

By choosing the first row:

$$\begin{aligned} \det(A) &= a_{11} D_{11} + a_{12} D_{12} + a_{13} D_{13} \\ &= a_{11}(-1)^{1+1} \begin{vmatrix} a_{22} & a_{23} \\ a_{32} & a_{33} \end{vmatrix} + a_{12}(-1)^{1+2} \begin{vmatrix} a_{21} & a_{23} \\ a_{31} & a_{33} \end{vmatrix} + a_{13}(-1)^{1+3} \begin{vmatrix} a_{21} & a_{22} \\ a_{31} & a_{32} \end{vmatrix} \\ &= a_{11} \begin{vmatrix} a_{22} & a_{23} \\ a_{32} & a_{33} \end{vmatrix} - a_{12} \begin{vmatrix} a_{21} & a_{23} \\ a_{31} & a_{33} \end{vmatrix} + a_{13} \begin{vmatrix} a_{21} & a_{22} \\ a_{31} & a_{32} \end{vmatrix}. \end{aligned}$$

By choosing the third column:

$$\begin{aligned} \det(A) &= a_{13} D_{13} + a_{23} D_{23} + a_{33} D_{33} \\ &= a_{13}(-1)^{1+3} \begin{vmatrix} a_{21} & a_{22} \\ a_{31} & a_{32} \end{vmatrix} + a_{23}(-1)^{2+3} \begin{vmatrix} a_{11} & a_{12} \\ a_{31} & a_{32} \end{vmatrix} + a_{33}(-1)^{3+3} \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} \\ &= a_{13} \begin{vmatrix} a_{21} & a_{22} \\ a_{31} & a_{32} \end{vmatrix} - a_{23} \begin{vmatrix} a_{11} & a_{12} \\ a_{31} & a_{32} \end{vmatrix} + a_{33} \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix}. \end{aligned}$$

5.2.1.4 Example 4: Determinant and Adjoint Matrix

Calculate $\det(A)$ for the matrix

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}.$$

By choosing the first row:

$$\begin{aligned} \det(A) &= a_{11}M_{11} + a_{12}M_{12} + a_{13}M_{13} \\ &= a_{11}\begin{vmatrix} a_{22} & a_{23} \\ a_{32} & a_{33} \end{vmatrix} - a_{12}\begin{vmatrix} a_{21} & a_{23} \\ a_{31} & a_{33} \end{vmatrix} + a_{13}\begin{vmatrix} a_{21} & a_{22} \\ a_{31} & a_{32} \end{vmatrix}. \end{aligned}$$

- **Adjoint Matrix (Cofactor Matrix):**

  The adjoint matrix, also called the cofactor matrix, is obtained by taking the cofactors of each element of A and arranging them in a matrix:

$$\text{adj}(A) = \begin{bmatrix} D_{11} & D_{12} & D_{13} \\ D_{21} & D_{22} & D_{23} \\ D_{31} & D_{32} & D_{33} \end{bmatrix},$$

  where $D_{ij} = (-1)^{i+j}M_{ij}$ is the cofactor corresponding to element a_{ij}.

  Let A be a square matrix of order n:

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix}.$$

  The adjoint matrix of A, denoted $\text{adj}(A)$, is defined as:

$$\text{adj}(A) = \begin{bmatrix} D_{11} & D_{12} & \dots & D_{1n} \\ D_{21} & D_{22} & \dots & D_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ D_{n1} & D_{n2} & \dots & D_{nn} \end{bmatrix},$$

  where $D_{ij} = (-1)^{i+j}M_{ij}$ is the cofactor of the element a_{ij}, and M_{ij} is the minor of a_{ij} (the determinant of the matrix obtained by removing row i and column j from A).

- **Inverse of a Matrix:**

  A square matrix A of order n is invertible (or nonsingular) if there exists a matrix B of the same order such that

$$AB = BA = I_n,$$

where I_n is the identity matrix of order n. The inverse of A is denoted by A^{-1} and is given by the formula:

$$A^{-1} = \frac{\mathrm{adj}(A)^{\mathrm{T}}}{\det(A)},$$

where $\det(A) \neq 0$ and $\mathrm{adj}(A)$ is the adjoint matrix of A.

5.3 Direct Methods for Solving Systems of Linear Equations

In the case where $m = n$, the system of linear equations can be written as:

$$Ax = b \quad \Leftrightarrow \quad \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}, \tag{5.4}$$

where a_{ij} and b_j are known, and n is the order of the system (the number of unknowns, which is equal to the number of equations). We are seeking the values of the unknowns x_i, $i = 1, \dots, n$.

5.4 Gauss Method

This method involves transforming the system $Ax = b$ into an **upper triangular system** (i.e., eliminating all elements of A below the main diagonal) starting from the augmented matrix:

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} & b_1 \\ a_{21} & a_{22} & \dots & a_{2n} & b_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} & b_n \end{bmatrix}.$$

Notations

- L_i : denotes the i-th row of the matrix.
- $L_k \leftrightarrow L_j$: interchange the k-th and j-th rows.
- $L_i \leftarrow L_i - \delta L_k$: replace the i-th row with the i-th row minus a multiple δ of the k-th row.

5.4.1 Algorithm of Gauss Method

Initialization: $k = 1$

1. **Step k:**

- If the pivot $a_{kk} = 0$, interchange rows $L_k \leftrightarrow L_j$ (with $j > k$ such that $a_{jk} \neq 0$) and go to step 3.
- Otherwise, continue.

2. For $i = k + 1, \ldots, n$, perform the elimination:

$$L_i \leftarrow L_i - \frac{a_{ik}}{a_{kk}} L_k$$

3. Increment $k \leftarrow k + 1$.
   - If $k \leq n - 1$, return to step 1.
   - Otherwise, stop: we obtain the upper triangular system

$$\begin{bmatrix} a_{11}^* & a_{12}^* & \ldots & a_{1n}^* & b_1^* \\ 0 & a_{22}^* & \ldots & a_{2n}^* & b_2^* \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \ldots & a_{nn}^* & b_n^* \end{bmatrix}.$$

Backward Substitution

Once the upper triangular system is obtained, the solution $\mathbf{x}^T = [x_1, x_2, \ldots, x_n]$ is computed by applying backward substitution:

$$x_n = \frac{b_n^*}{a_{nn}^*}, \quad x_i = \frac{1}{a_{ii}^*} \left(b_i^* - \sum_{m=i+1}^{n} a_{im}^* x_m \right), \quad i = n-1, n-2, \ldots, 1.$$

5.4.2 Example:

Solve the system of linear equations using the Gauss method:

$$\begin{cases} x_2 + x_3 = 2 \\ 2x_1 - x_2 - x_3 = 0 \\ x_1 + x_2 - x_3 = 1 \end{cases}$$

Solution

The system can be written in matrix form as follows:

$$\begin{bmatrix} 0 & 1 & 1 \\ 2 & -1 & -1 \\ 1 & 1 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 2 \\ 0 \\ 1 \end{bmatrix}.$$

Starting from the augmented matrix:

$$\left[\begin{array}{ccc|c} 0 & 1 & 1 & 2 \\ 2 & -1 & -1 & 0 \\ 1 & 1 & -1 & 1 \end{array}\right].$$

Step 1: $k = 1$

The pivot $a_{11} = 0$, so we swap rows $L_1 \leftrightarrow L_2$:

$$\left[\begin{array}{ccc|c} 2 & -1 & -1 & 0 \\ 0 & 1 & 1 & 2 \\ 1 & 1 & -1 & 1 \end{array}\right].$$

The new pivot is 2, and we eliminate all elements below this pivot:

- Row 2: the element below the pivot is already zero, so $L_2 \leftarrow L_2$.
- Row 3: the element below the pivot is 1, so we make it zero:

$$L_3 \leftarrow L_3 - \frac{1}{2}L_1$$

$$\left[\begin{array}{ccc|c} 2 & -1 & -1 & 0 \\ 0 & 1 & 1 & 2 \\ 0 & \frac{3}{2} & -\frac{1}{2} & 1 \end{array}\right].$$

Step 2: $k = 2$

The pivot $a_{22} = 1 \neq 0$. We eliminate the element below the pivot:

$$L_3 \leftarrow L_3 - \frac{3/2}{1}L_2 = L_3 - \frac{3}{2}L_2$$

The resulting augmented matrix is:

$$\left[\begin{array}{ccc|c} 2 & -1 & -1 & 0 \\ 0 & 1 & 1 & 2 \\ 0 & 0 & -2 & -2 \end{array}\right] = [A^* \mid b^*].$$

The matrix A^* is now in upper triangular form, so we stop the elimination process. The system can be written as:

$$A^*x = b^*, \quad \begin{bmatrix} 2 & -1 & -1 \\ 0 & 1 & 1 \\ 0 & 0 & -2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 2 \\ -2 \end{bmatrix}.$$

Backward Substitution

$$-2x_3 = -2 \quad \Rightarrow \quad x_3 = 1$$

$$x_2 + x_3 = 2 \quad \Rightarrow \quad x_2 = 1$$

$$2x_1 - x_2 - x_3 = 0 \quad \Rightarrow \quad x_1 = 1$$

$$\boxed{x_1 = 1,\ x_2 = 1,\ x_3 = 1}$$

5.4.3 Gauss Method with Partial Pivoting Strategy

At step k, the pivot element is chosen as the largest (in absolute value) element in the k-th column from rows k to n:

$$\text{pivot}_k = a_{rk} = \max_{i=k,\ldots,n} |a_{ik}|$$

where r is the row index corresponding to this maximum value.

We then swap rows r and k:

$$L_r \leftrightarrow L_k$$

and continue the elimination process using the standard Gauss algorithm.

5.4.4 Example: Gauss Method with Partial Pivoting

Solve the system of linear equations using the Gauss method with partial pivoting strategy:

$$\begin{cases} x_2 + x_3 = 2 \\ 2x_1 - x_2 - x_3 = 0 \\ x_1 + x_2 - x_3 = 1 \end{cases}$$

Solution

The system in matrix form:

$$\begin{bmatrix} 0 & 1 & 1 \\ 2 & -1 & -1 \\ 1 & 1 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 2 \\ 0 \\ 1 \end{bmatrix}$$

Starting from the augmented matrix:

$$\left[\begin{array}{ccc|c} 0 & 1 & 1 & 2 \\ 2 & -1 & -1 & 0 \\ 1 & 1 & -1 & 1 \end{array}\right] \quad L_1, L_2, L_3$$

Step 1: $k = 1$ Partial pivoting: pivot = $\max_{i=1,3} |a_{ik}| = \max(|0|, |2|, |1|) = 2$ Swap rows: $L_1 \leftrightarrow L_2$

$$\left[\begin{array}{ccc|c} 2 & -1 & -1 & 0 \\ 0 & 1 & 1 & 2 \\ 1 & 1 & -1 & 1 \end{array}\right]$$

Eliminate elements below pivot: - $L_2 \leftarrow L_2 - \frac{0}{2} L_1$ (unchanged) - $L_3 \leftarrow L_3 - \frac{1}{2} L_1$

$$\left[\begin{array}{ccc|c} 2 & -1 & -1 & 0 \\ 0 & 1 & 1 & 2 \\ 0 & 3/2 & -1/2 & 1 \end{array}\right]$$

Step 2: $k = 2$ Partial pivoting: pivot = $\max(|1|, |3/2|) = 3/2$ Swap rows: $L_2 \leftrightarrow L_3$

$$\left[\begin{array}{ccc|c} 2 & -1 & -1 & 0 \\ 0 & 3/2 & -1/2 & 1 \\ 0 & 1 & 1 & 2 \end{array}\right]$$

Eliminate element below pivot: $L_3 \leftarrow L_3 - \frac{1}{3/2} L_2 = L_3 - \frac{2}{3} L_2$

$$\left[\begin{array}{ccc|c} 2 & -1 & -1 & 0 \\ 0 & 3/2 & -1/2 & 1 \\ 0 & 0 & 4/3 & 4/3 \end{array}\right]$$

Step 3: Backward Substitution From the last row: $\frac{4}{3} x_3 = \frac{4}{3} \implies x_3 = 1$

Second row: $\frac{3}{2} x_2 - \frac{1}{2} x_3 = 1 \implies \frac{3}{2} x_2 - \frac{1}{2} = 1 \implies x_2 = 1$

First row: $2x_1 - x_2 - x_3 = 0 \implies 2x_1 - 1 - 1 = 0 \implies x_1 = 1$

$$\boxed{x_1 = 1,\ x_2 = 1,\ x_3 = 1}$$

$$\begin{bmatrix} 2 & -1 & -1 \\ 0 & 3/2 & -1/2 \\ 0 & 0 & 4/3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 4/3 \end{bmatrix}$$

From the last row:

$$\frac{4}{3} x_3 = \frac{4}{3} \implies x_3 = 1$$

Second row:

$$\frac{3}{2}x_2 - \frac{1}{2}x_3 = 1 \implies \frac{3}{2}x_2 - \frac{1}{2} = 1 \implies x_2 = 1$$

First row:

$$2x_1 - x_2 - x_3 = 0 \implies 2x_1 - 1 - 1 = 0 \implies x_1 = 1$$

$$\boxed{x_1 = 1,\ x_2 = 1,\ x_3 = 1}$$

5.4.5 Gauss Method with Composite Partial Pivoting

At step k, the pivot element a_{ik} is chosen as:

$$\text{pivot}_k = \max_{i=k,\ldots,n} |a_{ik}|,$$

and within the row i, the element a_{ij} with the largest absolute value is selected:

$$d_i = \max_{j=1,\ldots,n} |a_{ij}|.$$

We then swap the rows $i \leftrightarrow k$ and continue the elimination using the standard Gauss algorithm.

5.4.6 Gauss method with total pivoting strategy

At step k, the pivot element a_{ik} is chosen as:

$$\text{pivot}_k = \max_{\substack{i=k,\ldots,n \\ j=k,\ldots,n}} |a_{ij}|$$

Once the pivot $a_{rk,m}$ is identified:

- Swap rows $r \leftrightarrow k$.
- If $m \neq k$, swap columns $m \leftrightarrow k$ and also swap the corresponding unknowns $x_m \leftrightarrow x_k$ in the vector $\mathbf{x}$.

After pivoting, continue the elimination process using the standard Gauss algorithm to reduce the matrix to upper triangular form.

This method ensures maximum numerical stability by selecting the largest available element in the remaining submatrix as the pivot.

5.4.7 Determinant of a Matrix Using the Gauss Method

The determinant of a square matrix A can be computed using the Gauss elimination method:

$$\det(A) = (-1)^{\alpha} \prod_{i=1}^{n} a_{ii}^{*}$$

where:

- a_{ii}^{*} are the pivot elements (diagonal elements of the upper triangular matrix after elimination),
- α is the total number of row swaps performed during the elimination process.

5.5 Gauss-Jordan Method

This method involves transforming the matrix A into an identity matrix. The solution is obtained directly from the last column of the augmented matrix.

Algorithm

1. **Initialization:** $k = 1$.
2. If the pivot $a_{kk} = 0$, swap row k with a row $j > k$ such that $a_{jk} \neq 0$.
3. Divide row k by the pivot a_{kk}.
4. For all rows $i \neq k$, replace $L_i \leftarrow L_i - a_{ik}L_k$ to make all elements in column k zero except the pivot.
5. If A has become the identity matrix, then the solution is

$$x_i = b_i^{*},$$

   where b_i^{*} are the elements of the last column of the final augmented matrix.
6. Otherwise, set $k \leftarrow k + 1$ and return to step 2.

5.5.1 Example:

Solve the system of linear equations using the Gauss-Jordan method:

$$\begin{cases} x_2 + x_3 = 2 \\ 2x_1 - x_2 - x_3 = 0 \\ x_1 + x_2 - x_3 = 1 \end{cases}$$

Solution:

We write the augmented matrix:

$$[A|b] = \begin{bmatrix} 0 & 1 & 1 & 2 \\ 2 & -1 & -1 & 0 \\ 1 & 1 & -1 & 1 \end{bmatrix}.$$

Step 1: Pivot in column 1. Swap $L_1 \leftrightarrow L_2$:

$$\begin{bmatrix} 2 & -1 & -1 & 0 \\ 0 & 1 & 1 & 2 \\ 1 & 1 & -1 & 1 \end{bmatrix}.$$

Divide L_1 by the pivot 2:

$$L_1 \leftarrow \frac{1}{2}L_1 \Rightarrow \begin{bmatrix} 1 & -1/2 & -1/2 & 0 \\ 0 & 1 & 1 & 2 \\ 1 & 1 & -1 & 1 \end{bmatrix}.$$

Eliminate column 1 for L_3: $L_3 \leftarrow L_3 - L_1$:

$$\begin{bmatrix} 1 & -1/2 & -1/2 & 0 \\ 0 & 1 & 1 & 2 \\ 0 & 3/2 & -1/2 & 1 \end{bmatrix}.$$

Step 2: Pivot in column 2. Eliminate column 2 for L_3: $L_3 \leftarrow L_3 - \frac{3}{2}L_2$:

$$\begin{bmatrix} 1 & -1/2 & -1/2 & 0 \\ 0 & 1 & 1 & 2 \\ 0 & 0 & -2 & -2 \end{bmatrix}.$$

Divide L_3 by -2: $L_3 \leftarrow L_3/(-2)$:

$$\begin{bmatrix} 1 & -1/2 & -1/2 & 0 \\ 0 & 1 & 1 & 2 \\ 0 & 0 & 1 & 1 \end{bmatrix}.$$

Step 3: Eliminate above pivots:

$$L_2 \leftarrow L_2 - L_3, \quad L_1 \leftarrow L_1 + \frac{1}{2}L_3$$

$$\begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}.$$

Solution:

$$x_1 = 1, \quad x_2 = 1, \quad x_3 = 1.$$

5.5.2 Inverse of a Matrix Using the Gauss-Jordan Method

Let A be a square invertible (non-singular) matrix. Its inverse is denoted as A^{-1}.

To compute A^{-1} using the Gauss-Jordan method, we form the augmented matrix with the identity matrix:

$$[A|I_n],$$

where I_n is the identity matrix of order n. Then, we apply Gauss-Jordan elimination to transform A into I_n. The resulting augmented part will be the inverse:

$$[I_n|A^{-1}].$$

5.5.3 Determinant of a Matrix Using the Gauss-Jordan Method

The determinant of the matrix A, $\det(A)$, can be computed during the Gauss-Jordan process. It is given by:

$$\det(A) = \prod (\text{the elements used to divide the rows to achieve pivots of 1}).$$

That is, multiply all the pivot divisors used in the elimination steps to get the determinant.

5.5.4 Example: Inverse and Determinant Using Gauss-Jordan Method

Calculate the inverse and the determinant of the matrix

$$A = \begin{bmatrix} 0.5 & 0 & 2 \\ 0 & 0.5 & 1 \\ 1 & 1 & 2 \end{bmatrix}.$$

Solution:

We start by forming the augmented matrix $[A|I]$:

$$\left[\begin{array}{ccc|ccc} 0.5 & 0 & 2 & 1 & 0 & 0 \\ 0 & 0.5 & 1 & 0 & 1 & 0 \\ 1 & 1 & 2 & 0 & 0 & 1 \end{array}\right]$$

Step 1: Divide L_1 by the pivot 0.5:

$$L_1 \leftarrow \frac{1}{0.5} L_1$$

$$\left[\begin{array}{ccc|ccc} 1 & 0 & 4 & 2 & 0 & 0 \\ 0 & 0.5 & 1 & 0 & 1 & 0 \\ 1 & 1 & 2 & 0 & 0 & 1 \end{array}\right]$$

Step 1: Eliminate below and normalize the first pivot:

$$L_2 \leftarrow L_2, \quad L_3 \leftarrow L_3 - L_1$$

$$\left[\begin{array}{ccc|ccc} 1 & 0 & 4 & 2 & 0 & 0 \\ 0 & 0.5 & 1 & 0 & 1 & 0 \\ 0 & 1 & -2 & -2 & 1 & 1 \end{array}\right]$$

Step 2: Divide L_2 by 0.5:

$$L_2 \leftarrow \frac{1}{0.5} L_2$$

$$\left[\begin{array}{ccc|ccc} 1 & 0 & 4 & 2 & 0 & 0 \\ 0 & 1 & 2 & 0 & 2 & 0 \\ 0 & 1 & -2 & -2 & 1 & 1 \end{array}\right]$$

Step 3: Eliminate elements in column 2:

$$L_1 \leftarrow L_1 - 4L_3, \quad L_3 \leftarrow L_3 - L_2$$

$$\left[\begin{array}{ccc|ccc} 1 & 0 & 4 & 0 & -2 & -1 \\ 0 & 1 & 2 & 0 & 2 & 0 \\ 0 & 0 & 1 & 0.5 & 0.5 & -0.25 \end{array}\right]$$

Step 4: Backward elimination to get identity on the left:

$$L_1 \leftarrow L_1 - 4L_3, \quad L_2 \leftarrow L_2 - 2L_3$$

$$\left[\begin{array}{ccc|ccc} 1 & 0 & 0 & 0 & -2 & -1 \\ 0 & 1 & 0 & 1 & 1 & 0.5 \\ 0 & 0 & 1 & 0.5 & 0.5 & -0.25 \end{array}\right]$$

Result: The inverse of A is

$$A^{-1} = \begin{bmatrix} 0 & -2 & -1 \\ 1 & 1 & 0.5 \\ 0.5 & 0.5 & -0.25 \end{bmatrix}$$

Determinant: The determinant is the product of the pivots used:

$$\det(A) = (0.5) \cdot (0.5) \cdot (-4) = -1$$

5.6 Crout method (LU Decomposition)

Applicability Condition: The Crout method can be applied (i.e., the system admits an LU decomposition) only if all leading principal minors of A are non-zero:

$$\det(A_k) \neq 0, \quad k = 1, \ldots, n$$

For example, for $n = 3$:

$$\Delta_1 = a_{11} \neq 0, \quad \Delta_2 = \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} \neq 0, \quad \Delta_3 = \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} \neq 0$$

LU Decomposition: We decompose A as

$$A = LU,$$

where L is lower triangular with non-zero diagonal and U is upper triangular with unit diagonal:

$$L = \begin{bmatrix} L_{11} & 0 & 0 \\ L_{21} & L_{22} & 0 \\ L_{31} & L_{32} & L_{33} \end{bmatrix}, \quad U = \begin{bmatrix} 1 & U_{12} & U_{13} \\ 0 & 1 & U_{23} \\ 0 & 0 & 1 \end{bmatrix}.$$

Solving the System: For $Ax = b$, we have

$$Ax = b \implies LUx = b.$$

We introduce $y = Ux$, leading to two simpler systems:

$$\begin{cases} Ly = b & \text{(lower triangular system)} \\ Ux = y & \text{(upper triangular system)} \end{cases}$$

These two systems can be solved easily using forward and backward substitution, respectively.

Identification of Elements: In Crout's method, the elements of L and U are computed sequentially to satisfy $A = LU$, ensuring all pivots are non-zero.

Then the product LU gives:

$$LU = \begin{bmatrix} L_{11} & 0 & 0 \\ L_{21} & L_{22} & 0 \\ L_{31} & L_{32} & L_{33} \end{bmatrix} \begin{bmatrix} 1 & U_{12} & U_{13} \\ 0 & 1 & U_{23} \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}.$$

Procedure to Identify Elements

1. Determine the **first column of** L and the **first row of** U.
2. Determine the **second column of** L and the **second row of** U.
3. Determine the **third column of** L and the **third row of** U.

Solving the System

Once all elements L_{ij} and U_{ij} are determined:

$$\begin{cases} Ly = b & \text{(solve for } y \text{ using forward substitution)} \\ Ux = y & \text{(solve for } x \text{ using backward substitution)} \end{cases}$$

Here, the vector y is obtained from $Ly = b$, and then the vector x is obtained from $Ux = y$.

Example :

Solve the system of linear equations using the Crout method (LU decomposition):

$$\begin{cases} x_2 + x_3 = 2 \\ 2x_1 - x_2 - x_3 = 0 \\ x_1 + x_2 - x_3 = 1 \end{cases}$$

The coefficient matrix is:

$$A = \begin{bmatrix} 0 & 1 & 1 \\ 2 & -1 & -1 \\ 1 & 1 & -1 \end{bmatrix}.$$

Check applicability of the Crout method:

The Crout method requires that all leading principal minors of A are non-zero. The first leading principal minor is:

$$\Delta_1 = |a_{11}| = |0| = 0.$$

Since $\Delta_1 = 0$, the Crout method cannot be applied to this system.

Example:

Solve the system of linear equations using the Crout method (LU decomposition):

$$\begin{cases} x_1 + 3x_2 + 3x_3 = -2 \\ 2x_1 + 2x_2 + 5x_3 = 7 \\ 3x_1 + 2x_2 + 6x_3 = 12 \end{cases}$$

Solution:

The coefficient matrix and the right-hand side vector are:

$$A = \begin{bmatrix} 1 & 3 & 3 \\ 2 & 2 & 5 \\ 3 & 2 & 6 \end{bmatrix}, \quad b = \begin{bmatrix} -2 \\ 7 \\ 12 \end{bmatrix}.$$

Check applicability:

$$\det(A_1) = |1| = 1 \neq 0, \quad \det(A_2) = \begin{vmatrix} 1 & 3 \\ 2 & 2 \end{vmatrix} = -4 \neq 0, \quad \det(A_3) = \begin{vmatrix} 1 & 3 & 3 \\ 2 & 2 & 5 \\ 3 & 2 & 6 \end{vmatrix} = 5 \neq 0.$$

Since all leading principal minors are non-zero, the Crout method can be applied.

LU decomposition:

$$A = LU$$

Proceeding by term-to-term identification:

$$L = \begin{bmatrix} l_{11} & 0 & 0 \\ l_{21} & l_{22} & 0 \\ l_{31} & l_{32} & l_{33} \end{bmatrix}, \quad U = \begin{bmatrix} 1 & u_{12} & u_{13} \\ 0 & 1 & u_{23} \\ 0 & 0 & 1 \end{bmatrix}.$$

Step-by-step identification:

- Column 1 of L: $l_{11} = 1, l_{21} = 2, l_{31} = 3$.
- Row 1 of U: $l_{11}u_{12} = 3 \Rightarrow u_{12} = 3$, $l_{11}u_{13} = 3 \Rightarrow u_{13} = 3$.
- Column 2 of L: $l_{21}u_{12} + l_{22} = 2 \Rightarrow l_{22} = -4$, $l_{31}u_{12} + l_{32} = 2 \Rightarrow l_{32} = -7$.
- Row 2 of U: $l_{21}u_{13} + l_{22}u_{23} = 2 \Rightarrow u_{23} = \frac{1}{4}$.
- Column 3 of L: $l_{31}u_{13} + l_{32}u_{23} + l_{33} = 6 \Rightarrow l_{33} = -\frac{5}{4}$.

Final matrices:

$$L = \begin{bmatrix} 1 & 0 & 0 \\ 2 & -4 & 0 \\ 3 & -7 & -\frac{5}{4} \end{bmatrix}, \quad U = \begin{bmatrix} 1 & 3 & 3 \\ 0 & 1 & \frac{1}{4} \\ 0 & 0 & 1 \end{bmatrix}.$$

$$L = \begin{bmatrix} 1 & 0 & 0 \\ 2 & -4 & 0 \\ 3 & -7 & -5/4 \end{bmatrix}, \quad b = \begin{bmatrix} -2 \\ 7 \\ 12 \end{bmatrix}$$

Solve $Ly = b$:

$$\begin{bmatrix} 1 & 0 & 0 \\ 2 & -4 & 0 \\ 3 & -7 & -5/4 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} -2 \\ 7 \\ 12 \end{bmatrix} \quad \Rightarrow \quad y_1 = -2, \quad y_2 = -\frac{11}{4}, \quad y_3 = 1$$

Step 2: Solve $Ux = y$

$$U = \begin{bmatrix} 1 & 3 & 3 \\ 0 & 1 & 1/4 \\ 0 & 0 & 1 \end{bmatrix}, \quad y = \begin{bmatrix} -2 \\ -11/4 \\ 1 \end{bmatrix}$$

Solve $Ux = y$:

$$\begin{bmatrix} 1 & 3 & 3 \\ 0 & 1 & 1/4 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} -2 \\ -11/4 \\ 1 \end{bmatrix} \quad \Rightarrow \quad x_3 = 1, \quad x_2 = -3, \quad x_1 = 4$$

$$\mathbf{x} = \begin{bmatrix} 4 \\ -3 \\ 1 \end{bmatrix}$$

5.6.1 Inverse of a matrix using LU Decomposition (Crout)

Let $A = LU$ and assume $L^{-1} = \hat{L}$. Then we solve:

$$A^{-1} = U^{-1}L^{-1} = U^{-1}\hat{L}$$

Using forward substitution, the elements of L^{-1} are obtained step by step:

$$\hat{L}_{11} = \frac{1}{L_{11}}, \quad \hat{L}_{21} = -\frac{L_{21}\hat{L}_{11}}{L_{22}}, \quad \hat{L}_{22} = \frac{1}{L_{22}}, \quad \dots$$

Similarly, we compute U^{-1} and finally:

$$A^{-1} = U^{-1}L^{-1}$$

This completes the computation of the inverse using the Crout (LU) method.

5.6.2 Determinant of a Matrix using Crout Method

If a matrix A admits an LU decomposition, $A = LU$, then the determinant is given by:

$$\det(A) = \det(L)\det(U)$$

For the Crout method, L is a lower triangular matrix with unit diagonal in U (or U has 1 on its diagonal depending on convention). Therefore:

$$\det(U) = 1 \quad \Rightarrow \quad \det(A) = \det(L)$$

Since L is lower triangular, its determinant is the product of its diagonal elements:

$$\det(A) = \prod_{i=1}^{n} L_{ii}$$

5.6.3 Example: Inverse and Determinant using Crout Method

Solve for the inverse and determinant of

$$A = \begin{bmatrix} 1 & 3 & 3 \\ 2 & 2 & 5 \\ 3 & 2 & 6 \end{bmatrix}.$$

Solution:

Perform LU decomposition (Crout method):

$$L = \begin{bmatrix} 1 & 0 & 0 \\ 2 & -4 & 0 \\ 3 & -7 & -5/4 \end{bmatrix}, \quad U = \begin{bmatrix} 1 & 3 & 3 \\ 0 & 1 & 1/4 \\ 0 & 0 & 1 \end{bmatrix}.$$

Inverse of L

Let $L^{-1} = \hat{L}$, then solve $L\hat{L} = I$:

$$\hat{L} = \begin{bmatrix} 1 & 0 & 0 \\ 1/2 & -1/4 & 0 \\ -2/5 & 7/5 & -4/5 \end{bmatrix}.$$

Inverse of U

Let $U^{-1} = \hat{U}$, then solve $U\hat{U} = I$:

$$\hat{U} = \begin{bmatrix} 1 & -3 & -9/4 \\ 0 & 1 & -1/4 \\ 0 & 0 & 1 \end{bmatrix}.$$

Inverse of A

$$A^{-1} = U^{-1}L^{-1} = \begin{bmatrix} 2/5 & -12/5 & 9/5 \\ 3/5 & -3/5 & 1/5 \\ -2/5 & 7/5 & -4/5 \end{bmatrix}.$$

Determinant of A

$$\det(A) = \det(L)\det(U) = \det(L) = \prod_{i=1}^{n} L_{ii} = (1)(-4)(-5/4) = 5.$$

5.7 Cholesky method

Applicability Conditions

The Cholesky method can be applied if the matrix A satisfies:

1. A is symmetric: $A^T = A$.

2. A is positive definite (all leading principal minors are positive), i.e.,

$$\Delta_1 = a_{11} > 0, \quad \Delta_2 = \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} > 0, \quad \Delta_3 = \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} > 0, \ldots$$

Decomposition

We set

$$A = LL^T,$$

where L is a lower triangular matrix:

$$L = \begin{bmatrix} l_{11} & 0 & 0 \\ l_{21} & l_{22} & 0 \\ l_{31} & l_{32} & l_{33} \end{bmatrix}.$$

Solving the System $Ax = b$

Let

$$b = Ax \implies b = LL^T x.$$

Set

$$Ly = b \quad \text{and} \quad L^T x = y,$$

which gives two triangular systems:

1. Solve $Ly = b$ for y (forward substitution).
2. Solve $L^T x = y$ for x (backward substitution).

Identification of Elements of L

The elements of L are determined similarly to the Crout method, replacing U by L^T. Once L is known:

- Solve $Ly = b$ for y.
- Solve $L^T x = y$ for x.

5.7.1 Example: Solve a System Using the Cholesky Method

Solve the system:

$$\begin{cases} x_1 + x_2 = 1 \\ x_1 + 2x_2 + x_3 = 0 \\ x_2 + 5x_3 = 2 \end{cases}$$

Solution

The system can be written as $Ax = b$ with

$$A = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 2 & 1 \\ 0 & 1 & 5 \end{bmatrix}, \quad b = \begin{bmatrix} 1 \\ 0 \\ 2 \end{bmatrix}.$$

Applicability Conditions

Check positive definiteness (leading principal minors):

$$\Delta_1 = |1| = 1 > 0, \quad \Delta_2 = \begin{vmatrix} 1 & 1 \\ 1 & 2 \end{vmatrix} = 1 > 0, \quad \Delta_3 = \begin{vmatrix} 1 & 1 & 0 \\ 1 & 2 & 1 \\ 0 & 1 & 5 \end{vmatrix} = 5 > 0.$$

Hence, the Cholesky method can be applied.

Cholesky Decomposition

We seek $A = LL^T$, where

$$L = \begin{bmatrix} l_{11} & 0 & 0 \\ l_{21} & l_{22} & 0 \\ l_{31} & l_{32} & l_{33} \end{bmatrix}.$$

By term-to-term identification:

$$l_{11} = 1, \quad l_{21} = 1, \quad l_{31} = 0,$$

$$l_{22} = 1, \quad l_{32} = 1, \quad l_{33} = 2.$$

So

$$L = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 2 \end{bmatrix}, \quad L^T = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 2 \end{bmatrix}.$$

Solve $Ax = b$

1. Forward substitution: $Ly = b$

$$\begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 2 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 2 \end{bmatrix} \implies y_1 = 1, \quad y_2 = -1, \quad y_3 = \frac{3}{2}.$$

2. Backward substitution: $L^T x = y$

$$\begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1 \\ -1 \\ 3/2 \end{bmatrix} \implies x_3 = \frac{3}{4}, \quad x_2 = -\frac{7}{4}, \quad x_1 = \frac{11}{4}.$$

5.7.1.1 Inverse of A Using Cholesky

Let $A^{-1} = L^{-1}(L^T)^{-1}$. The elements of L^{-1} are obtained using forward substitution. Similarly, $(L^T)^{-1}$ is obtained using backward substitution.

5.7.1.2 Determinant of A

$$\det(A) = \det(L) \cdot \det(L^T) = (\prod_{i=1}^{n} l_{ii})^2 = (1 \cdot 1 \cdot 2)^2 = 4.$$

CHAPTER 6

APPROXIMATE METHOD FOR SOLVING SYSTEMS OF LINEAR EQUATION

In this chapter, we introduce various indirect methods (also referred to as *approximate* or *iterative methods*) for solving systems of linear equations. Iterative methods involve calculating a sequence of vectors $x^{(k)}$ that converge after a certain number of iterations to the solution x^* of the system $Ax = b$ (approximate solution). In comparison to so-called direct methods, these approaches are better suited for implementation on a calculator.

To solve systems of equations in the form $Ax = b$, we can proceed in a manner similar to solving equations of the type $f(x) = 0$, where we constructed a numerical sequence $x_{n+1} = g(x_n)$ that converged to the solution α of $f(x) = 0$. Indeed, the system $Ax = b$ can be equivalently transformed into the form $f(x) = 0$, where

$$x = \Xi x + \mathcal{P},$$

with Ξ being a matrix and $\mathcal{P}$ a vector. This way, we construct the recursive sequence known as the iterative process:

$$x_{n+1} = \Xi x_n + \mathcal{P}.$$

Typical iterative schemes are obtained by splitting the matrix A into $A = M - N$, where M is chosen invertible and easy to solve with, which yields the iteration

$$x_{n+1} = M^{-1}Nx_n + M^{-1}b.$$

Depending on the choice of M and N, classical methods such as JACOBI and GAUSS–SEIDEL are recovered. Convergence of the iterative process typically depends on the spectral radius $\rho(\Xi)$; a sufficient condition for convergence is $\rho(\Xi) < 1$.

6.1 Recap on the calculation of matrix norms

Let M be a matrix defined as

$$M = \begin{bmatrix} m_{11} & m_{12} & \cdots & m_{1n} \\ m_{21} & m_{22} & \cdots & m_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ m_{m1} & m_{m2} & \cdots & m_{mn} \end{bmatrix}.$$

The matrix norms are defined as follows:

$$\|M\|_1 = \max_j \left(\sum_{i=1}^{n} |m_{ij}| \right), \qquad \|M\|_\infty = \max_i \left(\sum_{j=1}^{n} |m_{ij}| \right).$$

6.2 Préliminaires

6.2.1 Characteristic polynomial of a matrix

The **characteristic polynomial** of a matrix M of dimension $n \times n$ is defined as:

$$P(\lambda) = \det(M - \lambda I),$$

where I is the identity matrix of dimension $n \times n$.

This polynomial is of degree n and has real or complex conjugate roots.

6.2.2 Eigenvalues λ_i of a matrix

The **characteristic polynomial** of a matrix M of dimension $n \times n$ is defined as:

$$P(\lambda) = \det(M - \lambda I),$$

where I is the identity matrix of dimension $n \times n$.

This polynomial is of degree n and has real or complex conjugate roots.

6.2.3 Spectral radius of a matrix

The **spectral radius** of a matrix M is defined as:

$$\rho(M) = \max_{i=1,\ldots,n} |\lambda_i|, \tag{6.3}$$

where λ_i are the eigenvalues of M, and $|\lambda_i|$ denotes the complex modulus of λ_i.

6.2.4 Strictly diagonally dominant matrix

A matrix $M = [m_{ij}]$ is said to be **strictly diagonally dominant (SDD)** if:

$$|m_{ii}| > \sum_{\substack{j=1 \\ j \neq i}}^{n} |m_{ij}|, \quad \forall i = 1, 2, \ldots, n,$$

where m_{ij} are the elements of the matrix M.

6.3 Indirect methods for solving systems of linear equations

6.3.1 Jacobi Iterative Method

Consider the system of linear equations:

$$Ax = b,$$

where

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix}, \quad x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}, \quad b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}.$$

The **Jacobi iterative process**, which allows us to compute the solution iteratively, is obtained as follows:

- Ensure that $a_{ii} \neq 0$ for all i; otherwise, interchange two rows.
- Decompose the matrix A of the system $Ax = b$ into the form:

$$A = D + L + U,$$

  where:

  - D is the diagonal part of A,
  - L is the strictly lower triangular part,
  - U is the strictly upper triangular part.

Let $A = D + L + U$, where the matrices D, L, and U are defined as:

$$U = \begin{bmatrix} 0 & a_{12} & a_{13} & \cdots & a_{1n} \\ 0 & 0 & a_{23} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & a_{n-1,n} \\ 0 & 0 & \cdots & 0 & 0 \end{bmatrix}, \quad D = \begin{bmatrix} a_{11} & 0 & 0 & \cdots & 0 \\ 0 & a_{22} & 0 & \cdots & 0 \\ 0 & 0 & a_{33} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & a_{nn} \end{bmatrix}, \quad L = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 \\ a_{21} & 0 & 0 & \cdots & 0 \\ a_{31} & a_{32} & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & a_{n3} & \cdots & 0 \end{bmatrix}.$$

The system $Ax = b$ can thus be written as:

$$(D + L + U)x = b.$$

Rearranging terms:

$$Dx = b - (L + U)x \quad \Rightarrow \quad x = D^{-1}[b - (L + U)x].$$

We set:

$$J = -D^{-1}(L + U), \quad C = D^{-1}b.$$

Hence, we obtain the matrix form of the Jacobi algorithm:

$$x^{(k+1)} = Jx^{(k)} + C, \tag{6.5}$$

where J is called the **iteration matrix** of the Jacobi method (or Jacobi matrix).

The elements of J are defined by:

$$J_{ii} = 0, \qquad J_{ij} = -\frac{a_{ij}}{a_{ii}}, \quad 1 \leq i \leq n,\ i \neq j. \tag{6.6}$$

The components of the solution vector can be computed directly using:

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k)} \right), \quad i = 1, 2, \ldots, n.$$

6.3.1.1 Convergence conditions

The sufficient conditions for the **Jacobi method** to converge are:

- $\|J\|_\infty < 1$
- $\|J\|_1 < 1$
- The matrix A of the system is **strictly diagonally dominant (SDD)**.

In cases where the sufficient conditions are not met, the Jacobi process still converges for all initial vectors $x^{(0)}$ if:

$$\rho(J) < 1,$$

where $\rho(J)$ denotes the **spectral radius** of J.

6.3.1.2 Number of necessary iterations

One can estimate the number of iterations required to achieve a given accuracy ε as:

$$k \geq \frac{\log\left(\dfrac{\varepsilon(1 - \|J\|)}{\|C\|}\right)}{\log(\|J\|)} \tag{6.8}$$

Remark 1: The theoretical estimation of k required to achieve the given accuracy often proves to be exaggerated in practice.

6.3.1.3 Stopping Criterion for the Jacobi Iterations

We stop the iterations when:

$$\max_{1\leq i\leq n} \left|x_i^{(k+1)} - x_i^{(k)}\right| \leq \varepsilon, \tag{6.9}$$

where ε is the desired accuracy.

The solution vector is then given by:

$$x^{(k)} = \begin{bmatrix} x_1^{(k)} \\ x_2^{(k)} \\ \vdots \\ x_n^{(k)} \end{bmatrix}.$$

6.3.1.4 Example: Solving a System using the Jacobi Method

Consider the system:

$$Ax = b,$$

where

$$A = \begin{bmatrix} 3 & 1 & -1 \\ 1 & 5 & 2 \\ 2 & -1 & -6 \end{bmatrix}, \quad x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}, \quad b = \begin{bmatrix} 2 \\ 17 \\ -18 \end{bmatrix}.$$

We solve this system using the Jacobi method with an accuracy of $\varepsilon = 10^{-2}$ and initial guess $x^{(0)} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$.

Solution: decompose $A = D + L + U$

$$D = \begin{bmatrix} 3 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & -6 \end{bmatrix}, \quad L = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 2 & -1 & 0 \end{bmatrix}, \quad U = \begin{bmatrix} 0 & 1 & -1 \\ 0 & 0 & 2 \\ 0 & 0 & 0 \end{bmatrix}.$$

Compute Jacobi Iteration Matrix and Vector:

$$J = -D^{-1}(L+U) = -\begin{bmatrix} 1/3 & 0 & 0 \\ 0 & 1/5 & 0 \\ 0 & 0 & -1/6 \end{bmatrix} \left(\begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 2 & -1 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 1 & -1 \\ 0 & 0 & 2 \\ 0 & 0 & 0 \end{bmatrix} \right) = \begin{bmatrix} 0 & -1/3 & 1/3 \\ -1/5 & 0 & -2/5 \\ 1/3 & -1/6 & 0 \end{bmatrix}.$$

$$C = D^{-1}b = \begin{bmatrix} 1/3 & 0 & 0 \\ 0 & 1/5 & 0 \\ 0 & 0 & -1/6 \end{bmatrix} \begin{bmatrix} 2 \\ 17 \\ -18 \end{bmatrix} = \begin{bmatrix} 2/3 \\ 17/5 \\ 3 \end{bmatrix}.$$

Iterative Formula:

The Jacobi iteration formula is:

$$x^{(k+1)} = Jx^{(k)} + C.$$

Convergence Conditions:

For the Jacobi iteration matrix J:

$$\|J\|_\infty = \max(2/3, 3/5, 1/6) = 0.67 < 1, \quad \|J\|_1 = \max(2/15, 1/2, 11/15) = 0.73 < 1.$$

The matrix A is strictly diagonally dominant (SDD):

$$|3| > |1| + |-1|, \quad |5| > |1| + |2|, \quad |6| > |2| + |-1|.$$

Since these three sufficient conditions are satisfied, the Jacobi process converges for any initial vector $x_0 \in \mathbb{R}^n$.

Spectral radius check:

$$\det(J - \lambda I) = \begin{vmatrix} -\lambda & -1/3 & 1/3 \\ -1/5 & -\lambda & -2/5 \\ 1/3 & -1/6 & -\lambda \end{vmatrix} = 0 \quad \Rightarrow \quad \lambda_1 = -0.58, \quad \lambda_{2,3} = -0.29 \pm 0.10i$$

$$\rho(J) = \max(0.58, 0.30) = 0.58 < 1 \quad \Rightarrow \text{Jacobi converges.}$$

Jacobi Iterations

The iteration formula is:

$$x^{(k+1)} = Jx^{(k)} + C, \quad C = \begin{bmatrix} 2/3 \\ 17/5 \\ 3 \end{bmatrix}.$$

Iteration 1:

$$x^{(1)} = Jx^{(0)} + C = \begin{bmatrix} 0 & -1/3 & 1/3 \\ -1/5 & 0 & -2/5 \\ 1/3 & -1/6 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 2/3 \\ 17/5 \\ 3 \end{bmatrix} = \begin{bmatrix} 0.67 \\ 3.40 \\ 3.00 \end{bmatrix}.$$

$$\max_i |x_i^{(1)} - x_i^{(0)}| = \max(|0.67 - 0|, |3.40 - 0|, |3.00 - 0|) = 3 > \varepsilon = 0.01$$

Iteration 2:

$$x^{(2)} = Jx^{(1)} + C = \begin{bmatrix} 0 & -1/3 & 1/3 \\ -1/5 & 0 & -2/5 \\ 1/3 & -1/6 & 0 \end{bmatrix} \begin{bmatrix} 0.67 \\ 3.40 \\ 3.00 \end{bmatrix} + \begin{bmatrix} 2/3 \\ 17/5 \\ 3 \end{bmatrix} = \begin{bmatrix} 0.53 \\ 2.07 \\ 2.65 \end{bmatrix}.$$

Iteration 3

$$x^{(3)} = Jx^{(2)} + C = \begin{bmatrix} 0 & -1/3 & 1/3 \\ -1/5 & 0 & -2/5 \\ 1/3 & -1/6 & 0 \end{bmatrix} \begin{bmatrix} 0.53 \\ 2.07 \\ 2.65 \end{bmatrix} + \begin{bmatrix} 2/3 \\ 17/5 \\ 3 \end{bmatrix} = \begin{bmatrix} 0.86 \\ 2.23 \\ 2.83 \end{bmatrix}$$

$$\max\left(|0.86 - 0.53|, |2.23 - 2.07|, |2.83 - 2.65|\right) = 0.33 > 0.01 = \varepsilon$$

Iteration 4

$$x^{(4)} = Jx^{(3)} + C = \begin{bmatrix} 0.87 \\ 2.09 \\ 2.91 \end{bmatrix}, \quad \max_i |x_i^{(4)} - x_i^{(3)}| = 0.14 > 0.01$$

Iteration 5

$$x^{(5)} = Jx^{(4)} + C = \begin{bmatrix} 0.94 \\ 2.06 \\ 2.94 \end{bmatrix}, \quad \max_i |x_i^{(5)} - x_i^{(4)}| = 0.07 > 0.01$$

Iteration 6

$$x^{(6)} = Jx^{(5)} + C = \begin{bmatrix} 0.96 \\ 2.03 \\ 2.97 \end{bmatrix}, \quad \max_i |x_i^{(6)} - x_i^{(5)}| = 0.03 > 0.01$$

Iteration 7

$$x^{(7)} = Jx^{(6)} + C = \begin{bmatrix} 0 & -1/3 & 1/3 \\ -1/5 & 0 & -2/5 \\ 1/3 & -1/6 & 0 \end{bmatrix} \begin{bmatrix} 0.96 \\ 2.03 \\ 2.97 \end{bmatrix} + \begin{bmatrix} 2/3 \\ 17/5 \\ 3 \end{bmatrix} = \begin{bmatrix} 0.98 \\ 2.02 \\ 2.98 \end{bmatrix}$$

$$\max\left(|0.98 - 0.96|, |2.02 - 2.03|, |2.98 - 2.97|\right) = 0.02 > 0.01 = \varepsilon$$

Iteration 8

$$x^{(8)} = Jx^{(7)} + C = \begin{bmatrix} 0 & -1/3 & 1/3 \\ -1/5 & 0 & -2/5 \\ 1/3 & -1/6 & 0 \end{bmatrix} \begin{bmatrix} 0.98 \\ 2.02 \\ 2.98 \end{bmatrix} + \begin{bmatrix} 2/3 \\ 17/5 \\ 3 \end{bmatrix} = \begin{bmatrix} 0.99 \\ 2.01 \\ 2.99 \end{bmatrix}$$

$$\max\left(|0.99 - 0.98|, |2.01 - 2.02|, |2.99 - 2.98|\right) = 0.01 \leq 0.01 = \varepsilon$$

Conclusion We stop the Jacobi process, and the approximate solution is:

$$x \approx \begin{bmatrix} 0.99 \\ 2.01 \\ 2.99 \end{bmatrix}.$$

The exact solution is:

$$x = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}.$$

6.4 Gauss-Seidel method

The Gauss-Seidel method is an improved variant of the Jacobi method. For simplicity, we consider a system of three equations with three unknowns and an initial guess

$$x^{(0)} = \begin{bmatrix} x_1^{(0)} \\ x_2^{(0)} \\ x_3^{(0)} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}.$$

First Iteration

In the first iteration, we substitute $x_2^{(0)}$ and $x_3^{(0)}$ into the first equation, giving:

$$x_1^{(1)} = \frac{1}{a_{11}} \left(b_1 - a_{12} x_2^{(0)} - a_{13} x_3^{(0)} \right).$$

It is this new value $x_1^{(1)}$, and not $x_1^{(0)}$, that is then substituted into the second equation of the system to compute $x_2^{(1)}$.

$$x_2^{(1)} = \frac{1}{a_{22}} \left(b_2 - a_{21} x_1^{(1)} - a_{23} x_3^{(0)} \right)$$

Similarly, in the third equation, we substitute $x_1^{(1)}$ and $x_2^{(1)}$ (not the initial values $x_1^{(0)}$ and $x_2^{(0)}$) to obtain:

$$x_3^{(1)} = \frac{1}{a_{33}} \left(b_3 - a_{31} x_1^{(1)} - a_{32} x_2^{(1)} \right)$$

This iterative process continues, which ensures much faster convergence compared to the Jacobi method.

General System with n Equations

For a system of n equations with n unknowns:

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k)} \right), \quad a_{ii} \neq 0$$

Otherwise, we swap two equations.

Matrix Form

The Gauss-Seidel process can be expressed in matrix form:

$$x^{(k+1)} = D^{-1}\big(b - L x^{(k+1)} - U x^{(k)}\big), \quad \text{where } A = D + L + U$$

Or equivalently:

$$x^{(k+1)} = G x^{(k)} + F$$

with

$$G = -(D + L)^{-1}U, \quad F = (D + L)^{-1}b$$

G is called the **iteration matrix** of the Gauss-Seidel method.

6.4.1 Convergence conditions of Gauss-Seidel Method

The sufficient conditions for the Gauss-Seidel process to converge are the same as for the Jacobi method:

- $\|G\|_\infty < 1$
- $\|G\|_1 < 1$
- The matrix A of the system is strictly diagonally dominant (SDD)

In cases where the sufficient conditions are not met, the Gauss-Seidel process still converges for any initial guess $x^{(0)}$ if:

$$\rho(G) < 1$$

where $\rho(G)$ is the spectral radius of G.

Remark 2: If the matrix A of the system $Ax = b$ is symmetric and positive definite, then the Gauss-Seidel method converges regardless of the initial guess $x^{(0)}$.

6.4.2 Number of Necessary Iterations

One can estimate the number of iterations required to achieve a given accuracy ε:

$$k \geq \frac{\log\left(\frac{\|x^{(k)} - x^*\|}{\|x^{(0)} - x^*\|}\right)}{\log(\rho(G)^{-1})}$$

where G is the iteration matrix and $\rho(G)$ is its spectral radius.

6.4.3 Gauss-Seidel Method: Termination Criterion

We stop the iterations when the maximum change between successive iterations is less than or equal to the desired accuracy ε:

$$\max_{1 \leq i \leq n} \left| x_i^{(k+1)} - x_i^{(k)} \right| \leq \varepsilon$$

The approximate solution is then:

$$x^{(k+1)} = \begin{bmatrix} x_1^{(k+1)} \\ x_2^{(k+1)} \\ \vdots \\ x_n^{(k+1)} \end{bmatrix}$$

6.4.4 Example: Gauss-Seidel Method

Consider the system:

$$Ax = b$$

where

$$A = \begin{bmatrix} 3 & 1 & -1 \\ 1 & 5 & 2 \\ 2 & -1 & -6 \end{bmatrix}, \quad x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}, \quad b = \begin{bmatrix} 2 \\ 17 \\ -18 \end{bmatrix}.$$

Solve this system using the Gauss-Seidel method with an accuracy $\varepsilon = 10^{-2}$ and initial guess

$$x^{(0)} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}.$$

Solution

Decompose A as $A = D + L + U$, where:

$$D = \begin{bmatrix} 3 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & -6 \end{bmatrix}, \quad L = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 2 & -1 & 0 \end{bmatrix}, \quad U = \begin{bmatrix} 0 & 1 & -1 \\ 0 & 0 & 2 \\ 0 & 0 & 0 \end{bmatrix}.$$

The iteration matrix G is defined as:

$$G = -(D + L)^{-1}U = -\begin{bmatrix} 1/3 & 0 & 0 \\ -1/15 & 1/5 & 0 \\ 11/90 & -1/30 & -1/6 \end{bmatrix} \begin{bmatrix} 0 & 1 & -1 \\ 0 & 0 & 2 \\ 0 & 0 & 0 \end{bmatrix}.$$

The iteration matrix G and vector C are given by:

$$G = \begin{bmatrix} 0 & -1/3 & 1/3 \\ 0 & 1/15 & -7/15 \\ 0 & -11/90 & 17/90 \end{bmatrix}, \quad C = (D + L)^{-1}b = \begin{bmatrix} 1/3 & 0 & 0 \\ -1/15 & 1/5 & 0 \\ 11/90 & -1/30 & -1/6 \end{bmatrix} \begin{bmatrix} 2 \\ 17 \\ -18 \end{bmatrix} = \begin{bmatrix} 2/3 \\ 49/15 \\ 241/90 \end{bmatrix}.$$

The Gauss-Seidel iteration is:

$$x^{(k+1)} = Gx^{(k)} + C.$$

Convergence Conditions

The sufficient conditions for convergence are:

- $\|G\|_\infty = \max(2/3, 8/15, 28/90) = 0.67 < 1$,
- $\|G\|_1 = \max(0, 48/90, 89/90) = 0.98 < 1$,
- The matrix A is strictly diagonally dominant (SDD): $|3| > |1| + |-1|$, $|5| > |1| + |2|$, $|6| > |2| + |-1|$.

All three sufficient conditions are satisfied, thus the Gauss-Seidel process converges for any initial guess $x^{(0)} \in \mathbb{R}$.

Spectral Radius Check

$$\det(G - \lambda I) = \begin{vmatrix} -\lambda & -1/3 & 1/3 \\ 0 & -\lambda + 1/15 & -2/5 \\ 0 & -11/90 & -\lambda + 17/90 \end{vmatrix} = 0$$

Eigenvalues:

$$\lambda_1 = 0, \quad \lambda_2 = 1/15 \approx 0.0667, \quad \lambda_3 = 17/90 \approx 0.189$$

$$\rho(G) = \max(0, 0.0667, 0.189) = 0.189 < 1,$$

so the Gauss-Seidel process converges to x^* for any $x^{(0)}$.

Iteration 1

$$x^{(1)} = Gx^{(0)} + C = \begin{bmatrix} 0 & -1/3 & 1/3 \\ 0 & 1/15 & -7/15 \\ 0 & -11/90 & 17/90 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 2/3 \\ 49/15 \\ 241/90 \end{bmatrix} = \begin{bmatrix} 2/3 \\ 49/15 \\ 241/90 \end{bmatrix}.$$

Iteration matrix G and vector C:

$$G = \begin{bmatrix} 0 & -1/3 & 1/3 \\ 0 & 1/15 & -7/15 \\ 0 & -11/90 & 17/90 \end{bmatrix}, \quad C = \begin{bmatrix} 2/3 \\ 49/15 \\ 241/90 \end{bmatrix}.$$

The iterative process is:

$$x^{(k+1)} = Gx^{(k)} + C$$

Iterations

Iteration 1:

$$x^{(1)} = Gx^{(0)} + C = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \Rightarrow x^{(1)} = \begin{bmatrix} 0.67 \\ 3.26 \\ 2.67 \end{bmatrix}, \quad \max\left(|x^{(1)} - x^{(0)}|\right) = 3.26 > 0.01$$

Iteration 2:

$$x^{(2)} = Gx^{(1)} + C = \begin{bmatrix} 0.47 \\ 2.23 \\ 2.78 \end{bmatrix}, \quad \max\left(|x^{(2)} - x^{(1)}|\right) = 1.03 > 0.01$$

Iteration 3:

$$x^{(3)} = Gx^{(2)} + C = \begin{bmatrix} 0.85 \\ 2.12 \\ 2.93 \end{bmatrix}, \quad \max\left(|x^{(3)} - x^{(2)}|\right) = 0.38 > 0.01$$

Iteration 4:

$$x^{(4)} = Gx^{(3)} + C = \begin{bmatrix} 0.94 \\ 2.04 \\ 2.97 \end{bmatrix}, \quad \max\left(|x^{(4)} - x^{(3)}|\right) = 0.09 > 0.01$$

Iteration 5:

$$x^{(5)} = Gx^{(4)} + C = \begin{bmatrix} 0.97 \\ 2.03 \\ 2.98 \end{bmatrix}, \quad \max\left(|x^{(5)} - x^{(4)}|\right) = 0.04 > 0.01$$

Iteration matrix G and vector C:

$$G = \begin{bmatrix} 0 & -1/3 & 1/3 \\ 0 & 1/15 & -7/15 \\ 0 & -11/90 & 17/90 \end{bmatrix}, \quad C = \begin{bmatrix} 2/3 \\ 49/15 \\ 241/90 \end{bmatrix}.$$

The iterative process:

$$x^{(k+1)} = Gx^{(k)} + C$$

Iterations

Iteration 1:

$$x^{(1)} = Gx^{(0)} + C = \begin{bmatrix} 0.67 \\ 3.26 \\ 2.67 \end{bmatrix}, \quad \max(|x^{(1)} - x^{(0)}|) = 3.26 > 0.01$$

Iteration 2:

$$x^{(2)} = \begin{bmatrix} 0.47 \\ 2.23 \\ 2.78 \end{bmatrix}, \quad \max(|x^{(2)} - x^{(1)}|) = 1.03 > 0.01$$

Iteration 3:

$$x^{(3)} = \begin{bmatrix} 0.85 \\ 2.12 \\ 2.93 \end{bmatrix}, \quad \max(|x^{(3)} - x^{(2)}|) = 0.38 > 0.01$$

Iteration 4:

$$x^{(4)} = \begin{bmatrix} 0.94 \\ 2.04 \\ 2.97 \end{bmatrix}, \quad \max(|x^{(4)} - x^{(3)}|) = 0.09 > 0.01$$

Iteration 5:

$$x^{(5)} = \begin{bmatrix} 0.98 \\ 2.01 \\ 2.99 \end{bmatrix}, \quad \max(|x^{(5)} - x^{(4)}|) = 0.03 > 0.01$$

Iteration 6:

$$x^{(6)} = \begin{bmatrix} 0.992 \\ 2.012 \\ 2.99 \end{bmatrix}, \quad \max(|x^{(6)} - x^{(5)}|) = 0.01 \leq 0.01$$

Convergence: The process stops as the desired accuracy $\epsilon = 0.01$ is reached.

$$\text{Approximate solution: } x \approx \begin{bmatrix} 0.992 \\ 2.012 \\ 2.99 \end{bmatrix}$$

$$\text{Exact solution: } x = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

BIBLIOGRAPHY

[1] Abramowitz, M. and Stegun, I., Handbook of Mathematical Functions, Dover Press, New York, 1964.

[2] Akai, T.J., Applied Numerical Methods for Engineers, Wiley, New York, NY, 1993.

[3] Al-Khafaji, A.W. and Tooley, J.R., Numerical Methods in Engineering Practice, Holt, Rinehart and Winston, New York, 1986.

[4] Allen, M. and Iaacson, E., Numerical Analysis for Applied Science, Wiley, New York, 1998.

[5] Atkinson, L.V. and Harley, P.J., Introduction to Numerical Methods with PASCAL, Addison Wesley, Reading, MA, 1984.

[6] Ayyub, B.M. and McCuen, R.H., Numerical Methods for Engineers, Prentice Hall, Upper Saddle River, New Jersey, NJ, 1996.

[7] Balagurusamy, E., Numerical Methods, Tata McGraw-Hill, New Delhi, India, 2002.

[8] Bjorck, A., Numerical Methods for Least Squares Problems, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1996.

[9] Bathe, K.J. and Wilson, E.L., Numerical Methods in Finite Element Analysis, Prentice Hall, Englewood Cliffs, NJ, 1976.