Physics Master Course:

Numerical Methods 2007

Hans Maassen

Radboud Universiteit Nijmegen
Onderwijsinstituut Wiskunde, Natuur- en Sterrenkunde
Toernooiveld 1
6525 ED Nijmegen

September 2007

1. Introduction

In these notes we introduce some of the main methods of numerical analysis: numerical solution of equations, polynomial interpolation and approximation, numerical integration and solution of ordinary differential equations.

References

[BuF] Richard. L. Burden, J. Douglas Faires: *Numerical Analysis*, PWS Publishing Company, Boston, 1993.

[Sto] J. Stoer: *Numerische Mathematik 1*, Springer, 1980.

[Jeu] M. de Jeu: *Numerieke Wiskunde 1, 2003-2004*, Dictaat Mathematisch Instituut, Universiteit Leiden, januari 2004.

2. Solving Equations

In this chapter we treat several methods for the solution of equations in one variable.

The most basic method, repeated bisection, has already been treated by Andreas Gürtler in his introduction to Matlab. This method is easy to use and practically always works. It has the disadvantages, however, of being slow, and not generalising easily to higher dimensions.

We shall now discuss two other common methods: iteration of a function, and the method of Newton-Raphson.

2.1. Iteration

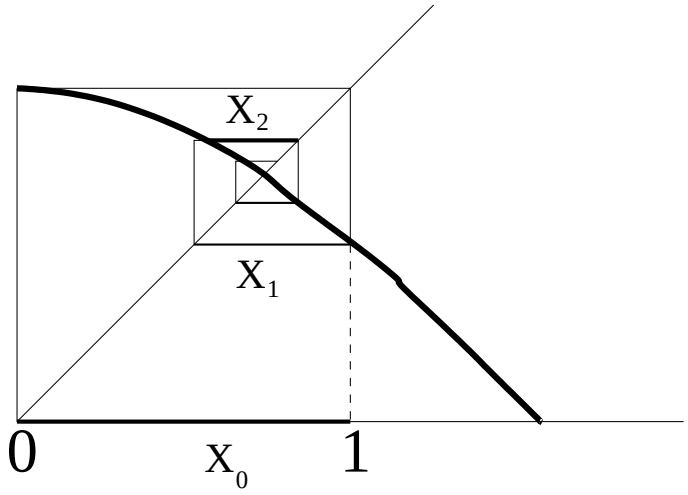
Suppose we wish to solve the equation

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

This can be easily achieved on a pocket calculator as follows:

- Enter any real number (for instance 0).
- Press the "cosine" button repeatedly.

The number in display will converge to the fixed point of the cosine function, i.e. the solution of our equation. How far we are still removed from the solution (the "error") can be estimated from the change in the display upon pressing the button: in this example the change is always *larger* than the error.



Theorem 1. *Let $g : [a, b] \to [a, b]$ be differentable with continuous derivative. Suppose that there exists $k < 1$ such that for all $x \in [a, b]$:*

$$|g'(x)| \leq k \ .$$

Then g has a unique fixed point p, and for any choice of a starting point $x_0 \in [a, b]$ the sequence defined by

$$x_{n+1} := g(x_n), \qquad (n \in \mathbb{N})$$

converges to p.

We obtain the above example by choosing $g : [-1, 1] \to [-1, 1] : x \mapsto \cos(x)$. Indeed, for $-1 \leq x \leq 1$ we have $|\cos'(x)| = |\sin(x)| \leq \sin(1) < 1$. (Note that for a general starting point) $x_0 \in \mathbb{R}$ outside $[-1, 1]$ the first iteration $x_1 = \cos(x_0)$ already lies in $[-1, 1]$.)
We shall write the n-fold composition $g \circ g \circ g \circ \cdots \circ g$ as $g^{\circ n}$.

Proof of Theorem 1. By the mean value theorem, for all $x, y \in [a, b]$ there exists z between x and y such that

$$|g(x) - g(y)| = |g'(z)| \cdot |x - y| \leq k|x - y| \ .$$

Therefore, for all $n \in \mathbb{N}$,

$$|g^{\circ n}(x) - g^{\circ n}(y)| \leq k^n |x - y| \leq k^n (b - a) \ .$$

Now let X_n denote the set of n-th iterates:

$$X_n := \left\{ \, g^{\circ n}(x) \, \middle| \, a \leq x \leq b \, \right\} .$$

We claim that as n increases, these sets shrink to a point, which must then be the unique fixed point of g. Indeed, let

$$a_n := \inf(X_n) \quad \text{and} \quad b_n := \sup(X_n) \ .$$

Since $X_0 \supset X_1 \supset X_2 \supset \ldots$, the sequence $a_0, a_1, a_2, \ldots$ must be increasing, say with limit p, and the sequence $b_0, b_1, b_2, \ldots$ must be decreasing, say with limit p'. But since

$$b_n - a_n \leq k^n (b - a) \ ,$$

p' must be equal to p. Now choose any $x_0 \in [a, b]$. Then, since $x_n := g^{\circ n}(x_0) \in X_n$,

$$a_n \leq x_n \leq b_n \ .$$

Hence the sequence $x_0, x_1, x_2, x_3, \ldots$ tends to p. This limit p must be a fixed point since g is continuous:

$$g(p) = g\left(\lim_{n\to\infty} x_n\right) = \lim_{n\to\infty} g(x_n) = \lim_{n\to\infty} x_{n+1} = p \ .$$

Finally, if $q \in [a, b]$ is a fixed point of g, then $q = g^{\circ n}(q)$. Hence

$$q = \lim_{n\to\infty} g^{\circ n}(q) = p \ .$$

So the fixed point p is unique. □

Many equations can be cast in the form of a fixed point equation. For example, the equation

$$x^3 + 4x^2 - 10 = 0 \tag{1}$$

has a unique root in the interval $[1, 2]$. It can be rewritten as

$$4x^2 = 10 - x^3, \quad \text{so} \quad x = \tfrac{1}{2}\sqrt{10 - x^3} \ .$$

However, at first the function $g : [1, 2] \to \mathbb{R} : x \mapsto \frac{1}{2}\sqrt{10 - x^3}$ does not meet our conditions, since it does not map $[1, 2]$ into itself, and has too large a derivative:

$$g'(x) = \frac{-3x^2}{4\sqrt{10 - x^3}} \ ,$$

varying from $g'(1) = -\frac{1}{4}$ to $g'(2) = -\frac{3}{2}\sqrt{2}$. However, when we restrict the function to $[1, \frac{3}{2}]$, it satisfies

$$|g'(x)| \leq |g'(\tfrac{3}{2})| \approx 0{,}66 \ .$$

So with any starting point in $[1, \frac{3}{2}]$ convergence is assured. With some inventiveness we can improve this procedure further: we can reformulate the equation (1) as

$$x^2(x + 4) = 10, \quad \text{so} \quad x = \sqrt{\frac{10}{x + 4}} \ .$$

It is not difficult to check that the function $g(x) := \sqrt{10/(x + 4)}$ maps the interval $[1, 2]$ into itself, and has quite a small derivative there. The convergence is accordingly faster. Roughly, the error in the approximation of p decreases by a factor $|g'(p)|$ each step once we are close to p. This means that we gain a digit of accuracy every ${}^{10}\log|g'(p)|$ steps. Not knowing p in advance, we can estimate this speed of convergence by ${}^{10}\log k$.

Exercise 1. Calculate the root of equation (1) to 9 decimal places, using both iteration procedures described above. Keep count of the number of iterations you need in each case.

2.2. The method of Newton-Raphson

The following very powerful method for the calculation of square roots is built into the hardware of pocket calculators. Let c be a positive number, and define

$$x_0 := c \quad \text{and} \quad x_{n+1} := \frac{1}{2}\left(x_n + \frac{c}{x_n}\right) .$$

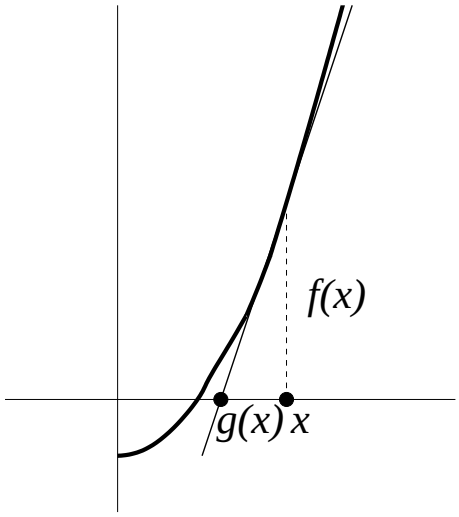
Then the sequence $x_0, x_1, x_2, \ldots$ converges to $\sqrt{c}$ quite fast. Once we are close to the square root, the number of accurate digits roughly *doubles* with every step!
This is a special case of the *method of Newton-Raphson*, which works as follows.

Theorem 2. *Let $f : [a, b] \to \mathbb{R}$ be twice differentiable with $f(a) \leq 0$ and $f(b) > 0$. Suppose that for all $x \in [a, b]$ both $f'(x)$ and $f''(x)$ are strictly positive and bounded from above. Let $x_0 \in [a, b]$ be such that $f(x_0) > 0$, and define for $n = 0, 1, 2, \ldots$:*

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

Then the sequence $x_0, x_1, x_2, \ldots$ converges to the unique root of the equation

$$f(p) = 0 .$$



We get the example above, the algorithm to calculate $\sqrt{c}$, by choosing

$$f : [\sqrt{c}, c] \to \mathbb{R} : x \mapsto x^2 - c .$$

Clearly, f' and f'' are strictly positive on this interval. Moreover, the function

$$g : [\sqrt{c}, c] \to [\sqrt{c}, c] : x \mapsto x - \frac{f(x)}{f'(x)}$$

can also be written as

$$g(x) = x - \frac{x^2 - c}{2x} = \frac{1}{2}\left(x + \frac{c}{x}\right) .$$

We can prove that the algorithm works in this case using Theorem 1! Indeed, for $x \geq \sqrt{c}$:

$$g'(x) = \frac{1}{2}\left(1 - \frac{c}{x^2}\right) \qquad \in [0, \tfrac{1}{2}] .$$

It follows that the sequence $x_0 \geq \sqrt{c}, x_1 := g(x_0), x_2 := g(g(x_0)), \ldots$ converges to the unique fixed point of g, which is $\sqrt{c}$.

Proof of Theorem 2. Since $f(x_0) \geq 0$, we have $x_0 \geq p$. Suppose that $x_n \geq p$ for some n. Then by the mean value theorem, $f(x_n) = f(p) + f'(y)(x_n - p)$, and since $f(p) = 0$ and f' is increasing, $f(x_n) \leq f'(x_n)(x_n - p)$, from which it follows that

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

By induction, the whole sequence $x_0, x_1, x_2, \ldots$ lies above p, so that $f(x_n) \geq 0$, and the sequence is decreasing. So it has a limit l, which must be such that $l = l - f(l)/f'(l)$, hence $f(l) = 0$ and $l = p$. □

Theorem 2 and its proof do not make clear why the convergence is so fast. We shall discuss this point separately.

Theorem 3. *In the situation of Theorem 2 there is a positive constant γ such that for $n \geq N$:*

$$|x_{n+1} - p| \leq \gamma |x_n - p|^2 .$$

This theorem says that, roughly speaking, when we are getting close to p the number of accurate digits doubles with every step.

Proof of Theorem 3. We can view the Newton-Raphson method as the iteration of

$$g : [p, b] \to [p, b] : x \mapsto x - \frac{f(x)}{f'(x)} .$$

Now note that

$$|g'(x)| = \frac{f(x)f''(x)}{f'(x)^2} \leq \alpha f(x) ,$$

for some properly chosen constant α. Also, $f'(x) \leq \beta$ for some β. It follows that there are y and z with $p \leq z \leq y \leq x$ such that.

$$|g(x) - p| = |g'(y)| \cdot |x - p| \leq \alpha f(y)|x - p| = \alpha f'(z)|x - p|^2 \leq \alpha\beta |x - p|^2 .$$

□

Exercise 2. Calculate the fixed point of the cosine function to 11 decimal places using

(a) bisection;
(b) iteration;
(c) Newton-Raphson.

Keep count of the number of steps required in each method.

Exercise 3.

(a) Use the method of Newton-Raphson to find the solution accurate to within 10^{-12} of the equation $(x-2)^2 - \log(x) = 0$, for $x \in [1, 2]$ and for $x \in [e, 4]$.

(b) Use the method of Newton-Raphson to approximate, to within 10^{-10}, the value of x that produces the point on the graph of $y = \frac{1}{x}$ that is closest to $(2, 1)$.

3. Polynomial Approximation and Interpolation

Polynomials are easily evaluated by computers. They are, moreover, easy to manipulate: addition, multiplication and differentiation are straightforward. This makes them ideal tools for the approximation of functions and the interpolation of data by smooth curves.

3.1. Approximating continuous functions

It is an encouraging fact that continuous functions can be approximated by polynomials arbitrarily well. This is the content of the following Theorem.

Theorem 4. *(***Weierstraß***) Let $f : [a, b] \to \mathbb{R}$ be continuous and let $\varepsilon > 0$. Then there exists a polynomial p such that for all $x \in [a, b]$:*

$$|f(x) - p(x)| < \varepsilon \, .$$

For this standard theorem of functional analysis we shall give a constructive proof with a probabilistic flavour.

Proof. Without loss of generality we may assume that $[a, b] = [0, 1]$. For $n \in \mathbb{N}$, let B_n be the *Bernstein polynomial* of degree n based on f:

$$B_n(x) := \sum_{k=0}^{n} \binom{n}{k} f\Big(\frac{k}{n}\Big) \, x^k (1-x)^{n-k} \, , \qquad (0 \le x \le 1).$$

We claim that, for n large enough, B_n approximates f as required.
Since f is a continous function on a closed interval, it is uniformly continuous, and hence there exists $\delta > 0$ such that for all $x, y \in [a, b]$:

$$|x - y| < \delta \quad \Longrightarrow \quad |f(x) - f(y)| < \tfrac{1}{2}\varepsilon \, .$$

We rewrite our polynomial in probabilistic language as follows. Let us fix $x \in [0, 1]$, and let X be a random variable, having binomial distribution with parameters n and x. Then

$$B_n(x) = \mathbb{E}\left(f\Big(\frac{1}{n} X\Big)\right) \, .$$

Now note that $\mathbb{E}(X/n) = x$ and $\mathrm{Var}\,(X/n) = x(1-x)/n$. So by Chebyshev's inequality

$$\mathbb{P}\left[\left|\frac{1}{n} X - x\right| \ge \delta\right] \le \frac{1}{\delta^2} \cdot \frac{x(1-x)}{n} \, .$$

We choose n so large that

$$\frac{1}{n\delta^2} < \frac{\varepsilon}{\|f\|} \, ,$$

where $\|f\|$ denotes the maximal value of $|f|$ on $[a,b]$. Then, since $x(1-x) \le \frac{1}{4}$,

$$\begin{aligned}
|B_n(x) - f(x)| &= \left| \mathbb{E}\left(f\Big(\frac{1}{n}X\Big)\right) - f(x)\right| \le \mathbb{E}\left| f\Big(\frac{1}{n}X\Big) - f(x)\right| \\
&= \mathbb{E}\left(\left| f\Big(\frac{1}{n}X\Big) - f(x)\right| \,\middle|\, \left|\frac{1}{n}X - x\right| < \delta \right) \cdot \mathbb{P}\left[\left|\frac{1}{n}X - x\right| < \delta \right] \\
&\quad + \mathbb{E}\left(\left| f\Big(\frac{1}{n}X\Big) - f(x)\right| \,\middle|\, \left|\frac{1}{n}X - x\right| \ge \delta \right) \cdot \mathbb{P}\left[\left|\frac{1}{n}X - x\right| \ge \delta \right] \\
&\le \tfrac{1}{2}\varepsilon + 2\,\|f\| \cdot \frac{\varepsilon}{4\,\|f\|} = \varepsilon\,.
\end{aligned}$$

□

In practice, the Bernstein polynomials are not very useful, since their convergence is slow. They have, however, the great advantage that they are quite stable: if f is changed only slightly, the Bernstein polynomials also move little. Their great competitors, the Lagrange polynomials, are highly sensitive to small changes in f.

Exercise 4. Calculate numerically the Bernstein polynomial for the function $f : [-1,1] \to \mathbb{R} : x \mapsto |x|$. How large a value of n do you need in order that $\|f - B_n\| < 0.08$? Compare this with the estimate in the proof of Theorem 4.

3.2. Lagrange's Interpolation Formula

Through two points in a plane passes a single line. Accordingly, through two points (ξ_0, α_0) and (ξ_1, α_1) in $\mathbb{R}^2$ (with different x-coordinates: $\xi_0 \neq \xi_1$), passes the graph of a unique linear function. This function can be written in different ways: either as

$$L_1(x) = \alpha_0 + (\alpha_1 - \alpha_0)\frac{x - \xi_0}{\xi_1 - \xi_0}\,,$$

or as

$$L_1(x) = \alpha_0 \frac{x - \xi_1}{\xi_0 - \xi_1} + \alpha_1 \frac{x - \xi_0}{\xi_1 - \xi_0}\,.$$

(Check that these functions indeed pass through the given points, and that they are the same.) This is the linear *Lagrange polynomial* through the two points. The first formula is known as *Newton's linear interpolation formula* and the second as *Lagrange's linear interpolation formula.*

Through three points (ξ_0, α_0), (ξ_1, α_1), and (ξ_2, α_2), passes the graph of a unique *quadratic* function L_2. It can alternatively be written (in 'Newton's way') as

$$L_2(x) = \alpha_0 + (\alpha_1 - \alpha_0)\frac{x - \xi_0}{\xi_1 - \xi_0} + (\alpha_2 - L_1(\xi_2)) \cdot \frac{(x - \xi_0)(x - \xi_1)}{(\xi_2 - \xi_0)(\xi_2 - \xi_1)}\,,$$

or (in 'Lagrange's way') as

$$L_2(x) = \alpha_0 \frac{(x - \xi_1)(x - \xi_2)}{(\xi_0 - \xi_1)(\xi_0 - \xi_2)} + \alpha_1 \frac{(x - \xi_0)(x - \xi_2)}{(\xi_1 - \xi_0)(\xi_1 - \xi_2)} + \alpha_2 \frac{(x - \xi_0)(x - \xi_1)}{(\xi_2 - \xi_0)(\xi_2 - \xi_1)}\,.$$

(Check again that these functions indeed pass through the given points, and that they are the same.)
In this way we may continue. First we follow Lagrange's line; we shall come back to Newton's *recursive* interpolation procedure in the next section.
Lagrange's approach is based on linear algebra: the set $\mathcal{P}_n$ of all polynomials of degree $\leq n$ can be viewed as a vector space under ordinary addition and multiplication by scalars. The dimension of this space is $n+1$; a basis is formed by the set of functions 1, x, x^2, ..., x^n. If a set of real numbers ξ_0, ξ_1, ..., ξ_n is given, a useful alternative basis is the *Lagrange basis* $\{\lambda_0, \lambda_1, \ldots, \lambda_n\}$, given by

$$\lambda_i(x) = \prod_{k \neq i} \frac{x - \xi_k}{\xi_i - \xi_k} \ .$$

It is easily seen that indeed these are polynomials of degree n and that

$$\lambda_i(\xi_j) = \delta_{ij} := \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{if } i \neq j. \end{cases}$$

Theorem 5. *(***Lagrange's interpolation formula***). Through $n+1$ points (ξ_0, α_0), (ξ_1, α_1), ..., (ξ_n, α_n) passes the graph of a unique polynomial L_n of degree $\leq n$. On the Lagrange basis it is given by*

$$L_n(x) = \sum_{i=0}^{n} \alpha_i \lambda_i(x) \ .$$

Proof. Uniqueness: Suppose that the graphs of polynomials p and q of degree $\leq n$ pass through our $n+1$ points:

$$p(\xi_i) = \alpha_i, \quad q(\xi_i) = \alpha_i, \quad \text{for} \quad i = 0, 1, \ldots n \ .$$

Now, a nonvanishing polynomial of degree $\leq n$ cannot have more than n zeroes. Therefore the polynomial $p - q$, being zero in $n+1$ points, namely $\xi_0, \ldots \xi_n$, must vanish identically, i.e. $p = q$.
Existence: Clearly, for $j = 0, 1, \ldots, n$,

$$L_n(\xi_j) = \sum_{i=0}^{n} \alpha_i \lambda_i(\xi_j) = \sum_{i=0}^{n} \alpha_i \delta_{ij} = \alpha_j \ .$$

So L_n fulfills our requirements. □

3.3. Newton's Interpolation Method

Suppose that we have fitted a polynomial through a number of data points, and that someone does an extra measurement, giving us a new point. If we were to use Lagrange's interpolation method described above, we would have to start all over again, calculating an entirely new Lagrange basis. Newton's method allows us to incorporate the new point, still using the results of previous calculations.
The result is, of course, the same. Here is a recipe.

Procedure. *Let the polynomial L_n of degree $\leq n$ passing through the points (ξ_0, α_0), (ξ_1, α_1), ..., (ξ_n, α_n) be given. Let $(\xi_{n+1}, \alpha_{n+1})$ be a new point, with $\xi_{n+1} \neq \xi_i$ for $i = 0, \ldots, n$. Then the polynomial of degree $\leq n+1$ passing through all the old points and the new point is given by*

$$L_{n+1}(x) := L_n(x) + (\alpha_{n+1} - L_n(\xi_{n+1})) \cdot \prod_{i=0}^{n} \frac{x - \xi_i}{\xi_{n+1} - \xi_i} \ . \tag{2}$$

Indeed, the extra term changes nothing in the points ξ_0, ..., ξ_n, and sets the value at ξ_{n+1} to α_{n+1}.

3.4. Approximation errors

We now return to function approximation. Let $f : [a, b] \to \mathbb{R}$ and a set of real numbers $a \leq \xi_0 < \xi_1 < \ldots < \xi_n \leq b$ be given. We may approximate f by the Lagrange polynomial through the points $(\xi_0, f(\xi_0))$, $(\xi_1, f(\xi_1))$, ..., $(\xi_n, f(\xi_n))$:

$$L_n(x) := \sum_{i=0}^{n} f(\xi_i)\lambda_i(x) \ .$$

How good is this approximation? That depends on the smoothness of f. In principle, the values of f outside the set $\{\xi_0, \xi_1, \ldots, \xi_n\}$ bear no relation to the values *in* these points, but if f is a few times differentiable, and a bound is known on the derivative, then L_n may be proved to lie quite close to f everywhere. It is this kind of result which we are after in this section.

We start with a kind of n-th order mean value theorem.

Theorem 6. *Let $f : [a, b] \to \mathbb{R}$ be $n+1$ times differentiable with continuous $n+1$-st derivative $f^{(n+1)}$. Let $a \leq \xi_0 < \xi_1 < \ldots < \xi_n \leq b$ and let L_n denote the Lagrange interpolation of f in the points ξ_0, ξ_1, ..., ξ_n. Then for every $x \in [a, b]$ there exists $q_x \in [a, b]$ such that*

$$f(x) - L_n(x) = \frac{1}{(n+1)!}(x - \xi_0)(x - \xi_1) \cdots (x - \xi_n) f^{(n+1)}(q_x) \ .$$

Proof. Fix a number $\overline{x} \in [a, b]$, and let L_{n+1} denote the Lagrange interpolation of f in ξ_0, ξ_1, ..., ξ_n, and $\overline{x}$. Differentiating (2), with ξ_{n+1} replaced by $\overline{x}$, $n+1$ times, we find that $L_{n+1}^{(n+1)}$ is a constant, namely

$$\frac{\big(f(\overline{x}) - L_n(\overline{x})\big)(n+1)!}{(\overline{x} - \xi_0) \cdots (\overline{x} - \xi_n)} \ .$$

We are done if we show that this constant value of $L_{n+1}^{(n+1)}$ is taken by $f^{(n+1)}$ in at least one point. We argue as follows. The difference $f - L_{n+1}$ has (at least) $n+2$ zeroes: ξ_0, ξ_1, ..., ξ_n, and $\overline{x}$. By Rolle's Theorem, between every pair of zeroes of $f - L_{n+1}$ there must lie at least one zero of $f' - L'_{n+1}$, hence $f' - L'_{n+1}$ must have at least $n+1$ zeroes. Continuing in this way, we conclude that $f^{(n+1)} - L_{n+1}^{(n+1)}$ indeed has at least one zero. □

Corollary 7. *Under the conditions of Theorem 6 we have*

$$|f(x) - L_n(x)| \leq \frac{1}{(n+1)!} \left|(x-\xi_0)(x-\xi_1)\cdots(x-\xi_n)\right| \cdot \left\|f^{(n+1)}\right\| \ .$$

Sometimes we may be satisfied with the rough and simple estimate:

$$\|f - L_n\| \leq \frac{(b-a)^{n+1}}{(n+1)!} \left\|f^{(n+1)}\right\| \ . \tag{3}$$

Warning. The above results may seem to indicate that Lagrange polynomial approximations of smooth functions converge reliably and quickly. Does not the factor $1/(n+1)!$ guarantee this? The following example shows that this suggestion is misleading.
Consider the function $f(x) = \frac{1}{1+x^2}$ on $[-5, 5]$. Let L_n denote the Lagrange interpolation of f in $n+1$ equidistant points $-5 = \xi_0 < \xi_1 < \ldots < \xi_n = 5$. Then for every $x \in (-5, 5)$ the sequence $L_1(x)$, $L_2(x)$, $L_3(x)$, ... *diverges.*

Exercise 5. Show that the estimate in Corollary 7 does not guarantee convergence of the sequence $(L_n)_{n\in\mathbb{N}^*}$ in the above example in any point x of the interval $[-5, 5]$ except the endpoints. Explain why this is not a counterexample to Theorem 4.

Exercise 6. Find the 4-th order Lagrange polynomial L_4 for the function $\log : [1, 2] \to \mathbb{R}$ with points of support 1, $\frac{5}{4}$, $\frac{3}{2}$, $\frac{7}{4}$, and 2. Evaluate this polynomial in the point $\frac{11}{8}$, and the error of this approximation, $\left|L_4\left(\frac{11}{8}\right) - \log\left(\frac{11}{8}\right)\right|$. Compare this error to the error bound of Corollary 7.

3.5. Piecewise Approximation

The example in Exercise 5 above shows that Lagrange approximation of functions by polynomials of higher and higher degree is not always a good idea. For this reason it is general practice to keep the degree low, say just 1, 2 or 3, and to approximate functions piecewise. This method will be the basis of the next Chapter, numerical integration.

4. Numerical Integration

The goal of this chapter is to describe methods for numerical approximation of the integral $\int_a^b f(x)dx$, given some more or less smooth function $f : [a, b] \to \mathbb{R}$.
The theory of Riemann integration gives such a method: calculating Riemann sums. However, the convergence properties of this method are poor. Under mild smoothness assumptions on f considerably better procedures are available.
Our strategy will be to first approximate f by a polynomial p_n, and then to integrate p_n. This may be done on the whole interval $[a, b]$, leading to a diversity of *quadrature rules* (rules for calculating areas, i.e. integrals). Or it may be done after subdivision of the interval, and piecewise approximation by polynomials, leading to *composite quadrature rules*.

4.1. Basic Quadrature Rules

We start with two simple quadrature rules.

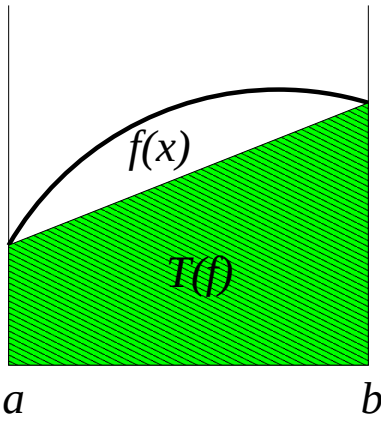
Trapezium rule.

If $f : [a, b] \to \mathbb{R}$ is twice differentiable with continuous second derivative, then the expression

$$T(f) := \tfrac{1}{2}(b-a)\big(f(a)+f(b)\big)$$

approximates the integral of f with an accuracy given by

$$\left| T(f) - \int_a^b f(x)dx \right| \le \frac{(b-a)^2}{12} \|f''\| \ .$$



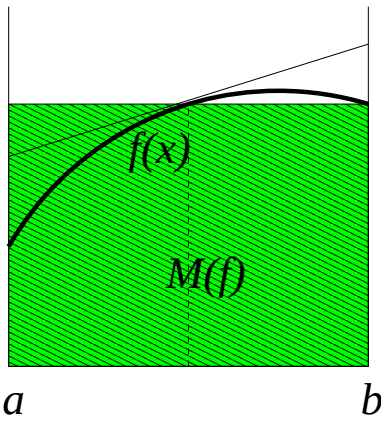
Midpoint rule.

If $f : [a, b] \to \mathbb{R}$ is twice differentiable with continuous second derivative, then the expression

$$M(f) := (b-a)\ f\big(\tfrac{a+b}{2}\big)$$

approximates the integral of f with an accuracy given by

$$\left| M(f) - \int_a^b f(x)dx \right| \le \frac{(b-a)^2}{24} \|f''\| \ .$$



Before we proceed to prove these statements, we note that the apparently more primitive Midpoint Rule is a factor 2 better then the Trapezium Rule! What is more, the figure below indicates that, if the Trapezium Rule yields a number that is too large, the Midpoint Rule usually comes out too small, and conversely. This suggests the, often much better, approximation described below.

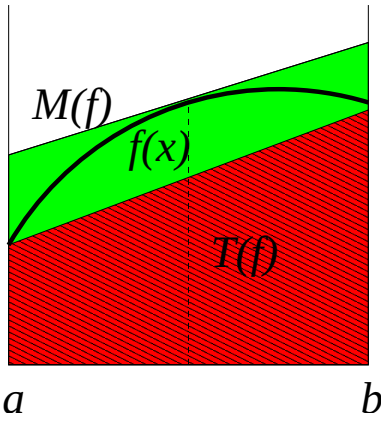
Simpson Rule.

If $f : [a,b] \to \mathbb{R}$ is four times differentiable with continuous fourth derivative, then the expression

$$S(f) := \tfrac{1}{3}\big(T(f)+2M(f)\big) = \tfrac{1}{6}\left(f(a)+4f\big(\tfrac{a+b}{2}\big)+f(b)\right)$$

approximates the integral of f with an accuracy given by

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



4.2. Quadrature Rules in General

We shall prove the results in the preceeding section by a lemma, based on Theorems 5 and Corollary 7 of Chapter 3. The lemma says that any set of $n+1$ points, chosen in some interval, leads to a quadrature rule which is exact for all polynomials of degree n or less.

Lemma 8. *Let $f : [a,b] \to \mathbb{R}$ be $n+1$ times differentiable with continuous $n+1$-st derivative $f^{(n+1)}$. Let $a \le \xi_0 < \xi_1 < \ldots < \xi_n \le b$ and let $\lambda_0, \lambda_1, \ldots, \lambda_n$ be the Lagrange basis associated to these points. Then the expression*

$$I_n(f) := \sum_{j=0}^{n} c_j f(\xi_j) \quad \text{where} \quad c_j := \int_a^b \lambda_j(x)dx$$

defines an approximation to the integral of f satisfying

$$\left|I_n(f)-\int_a^b f(x)dx\right| \le \frac{1}{(n+1)!}\left\|f^{(n+1)}\right\| \cdot \int_a^b |(x-\xi_0)\cdots(x-\xi_n)|\,dx .$$

Proof. Let L_n be the Lagrange polynomial of degree $\le n$ passing through the points $(\xi_0, f(\xi_0)), \ldots, (\xi_n, f(\xi_n))$. It follows from Theorem 5 that

$$\int_a^b L_n(x)dx = \sum_{j=0}^{n} f(\xi_j)\int_a^b \lambda_j(x)dx = \sum_{j=0}^{n} c_j f(\xi_j) = I_n(f) .$$

Hence

$$\left|I_n(f)-\int_a^b f(x)dx\right| = \left|\int_a^b L_n(x)dx - \int_a^b f(x)dx\right| \le \int_a^b |L_n(x)-f(x)|dx .$$

By Theorem 7 the statement follows. □

Proof of the Trapezium Rule. In the above lemma put $n=1$, $\xi_0 = a$, $\xi_1 = b$, and note that

$$c_0 = \int_a^b \frac{x-b}{a-b}dx = \tfrac{1}{2}(b-a) \quad \text{and} \quad c_1 = \int_a^b \frac{x-a}{b-a}dx = \tfrac{1}{2}(b-a) .$$

So in this case, $I_2(f) = \frac{1}{2}\big(b-a)(f(a)+f(b)\big) = T(f)$. Moreover, changing to the variable $u := (x-a)/(b-a)$:

$$\int_a^b |(x-a)(x-b)|\,dx = (b-a)^3 \int_0^1 u(1-u)\,du = \tfrac{1}{6}(b-a)^3 \ .$$

Hence by the lemma

$$\left| T(f) - \int_a^b f\,dx \right| \le \tfrac{1}{2}\,\|f''\| \int_a^b |(x-a)(x-b)|dx = \tfrac{1}{12}\,\|f''\|\,(b-a)^3 \ .$$

□

Proof of the Midpoint Rule. Here we have $n = 0$, $\xi_0 = \frac{1}{2}(a+b)$. Substituting these data into Lemma 8 we obtain

$$\left| M(f) - \int_a^b f dx \right| \le \|f'\| \int_a^b |x-\xi_0| dx = (b-a)^2\,\|f'\| \int_0^1 |u-\tfrac{1}{2}|\,du = \tfrac{1}{4}(b-a)^2\,\|f'\| \ .$$

However, we get a much better estimate by observing that, if we raise n to 1, and pick an extra point $\xi_1 \neq \xi_0$ somewhere in the interval $[a,b]$, our Lagrange polynomial changes, according to (2), by

$$L_1 - L_0 = \mathrm{cst}\cdot(x-\xi_0) \ .$$

It follows that the quadrature rule does not change at all:

$$I_1(f) - I_0(f) = \mathrm{cst}\cdot \int_a^b (x-\xi_0)dx = 0 \ .$$

Applying Lemma 8 to this situation, we obtain:

$$\left| M(f) - \int_a^b f dx \right| \le \tfrac{1}{2}\,\|f''\| \int_a^b |(x-\xi_0)(x-\xi_1)| dx \ .$$

This holds for all $\xi_1 \neq \xi_0 = \frac{1}{2}(a+b)$, but we may take the limit $\xi_1 \to \xi_0$:

$$\left| M(f) - \int_a^b f dx \right| \le \tfrac{1}{2}\,\|f''\| \int_a^b (x-\xi_0)^2 dx = (b-a)^3 \int_0^1 (u-\tfrac{1}{2})^2 du = \frac{1}{24}\,\|f''\|\,(b-a)^3 \ .$$

□

Proof of the Simpson Rule. This time, take $n = 3$, $\xi_0 = a$, $\xi_1 = \frac{1}{2}(a+b)$, $\xi_2 = b$, and ξ_3 arbitrary. The Lagrange quadrature rule I_2 applied to the points ξ_0, ξ_1 and ξ_2 only, is precisely the Simpson Rule: $S(f) = I_2(f)$. Indeed,

$$c_1 = \int_a^b \frac{(x-a)(x-b)}{(\xi_1-a)(\xi_1-b)}\,dx = 4(b-a)\int_0^1 u(1-u)\,du = \tfrac{2}{3}(b-a),$$

and $c_0 = c_2 = \frac{1}{6}(b-a)$ since $c_0 + c_1 + c_2 = I_2(1) = b - a$ and $c_0 = c_2$ by symmetry. When we add the fourth point ξ_3, the Lagrange polynomial changes from L_2 to L_3, where (see (2)),

$$L_3(x) - L_2(x) = \mathrm{cst} \cdot (x-a)(x - \tfrac{1}{2}(a+b))(x-b) \ ,$$

so that the Lagrange quadrature does not change at all:

$$I_3 - I_2 = \int_a^b \big(L_3(x) - L_2(x)\big)dx = 0 \ .$$

Hence I_3 does not depend on ξ_3, and we may take $\xi_3 \to \xi_1$ in our application of Lemma 8:

$$\left| S(f) - \int_a^b f dx \right| \leq \frac{1}{4!} \left\| f^{(4)} \right\| (b-a)^5 \int_0^1 u(1-u)(u-\tfrac{1}{2})^2 du = \frac{(b-a)^5}{2880} \left\| f^{(4)} \right\| \ .$$

□

4.3. Composite Quadrature Rule and Order

The Lagrangian quadrature rule I_n of Lemma 8 is based on $n+1$ points, and yields the exact integral for polynomial functions of degree n or less. However, we have seen that it may as well be exact for all polynomial functions of an even higher degree.

Definition. The *order* of a quadrature rule is the lowest value of k for which the rule is exact on all polynomials of degree less than k, but gives a wrong result for some polynomial of degree k.

For example, the Trapezium and Midpoint rules have order 2, and the Simpson rule has order 4.

We have seen at the end of Chapter 3 that, in order to increase the accuracy of a quadrature rule, it may not be wise to increase the order indefinitely: $\left\| f^{(n)} \right\|$ may go up too fast. Instead, we fix a quadrature of low order, but we chop the interval $[a,b]$ into more and more, usually equal, parts. This leads to *composite quadrature rules.* A modern refinement is *adaptive quadrature*, where the mesh of the partition is made to depend on the function f: where $f^{(p)}$ is large, the subdivision is made finer.

We now list three obvious composite quadrature rules, with their accuracy, as they follow from the basic rules given in the previous sections.

Composite Trapezium Rule. *Let $f : [a,b] \to \mathbb{R}$ be twice continuously differentable, and let $n \in \mathbb{N}$. Define $h := (b-a)/n$ and let T_n be the quadrature rule given by*

$$T_n(f) := h\big(\tfrac{1}{2}f(a) + f(a+h) + f(a+2h) + \cdots + f(a+(n-1)h) + \tfrac{1}{2}f(b)\big) \ .$$

Then

$$\left| T_n(f) - \int_a^b f dx \right| \leq \frac{1}{12} \left\| f'' \right\| \frac{(b-a)^3}{n^2} \ .$$

Composite Midpoint Rule. *Let $f : [a,b] \to \mathbb{R}$ be twice continuously differentable, and let $n \in \mathbb{N}$. Define $h := (b-a)/n$ and let M_n be the quadrature rule given by*

$$M_n(f) := h\big(f(a + \tfrac{1}{2}h) + f(a + \tfrac{3}{2}h) + \cdots + f(a + \tfrac{2n-1}{2}h)\big) \ .$$

Then

$$\left| M_n(f) - \int_a^b f dx \right| \leq \frac{1}{24} \|f''\| \frac{(b-a)^3}{n^2} .$$

Composite Simpson Rule. *Let $f : [a, b] \to \mathbb{R}$ be four times continuously differentable, and let $n \in \mathbb{N}$, $h := (b-a)/n$. Let S_n be the quadrature rule given by*

$$S_n(f) := \tfrac{1}{6}h\Big(f(a)+4f(a+\tfrac{1}{2}h)+2f(a+h)+4f(a+\tfrac{3}{2}h)+\cdots+2f(a+(n-1)h)+4f(a+\tfrac{2n-1}{2}h)+f(b) \Big)$$

Then

$$\left| S_n(f) - \int_a^b f dx \right| \leq \frac{1}{2880} \|f''''\| \frac{(b-a)^5}{n^4} .$$

Remark. The increase in accuracy of a composite quadrature rule goes as n^p, where p is the order of the elementary quadrature rule.

Exercise 7. Calculate π to 6 digits accuracy by evaluating the integral

$$\int_0^1 \frac{4}{1+x^2}\, dx ,$$

(a) using the composite Trapezium Rule;
(b) using the composite Midpoint Rule;
(c) using the composite Simpson Rule.

What is in each case the minimal value of n (number of subintervals) you need?

4.4. Gaussian Quadrature

As we have seen, the Midpoint rule, which is based on a single point only, is nevertheless of order 2. The Trapezium Rule is based on two points, and also has order 2. However, it turns out to be possible to find a different pair of points in the interval, so that the order of the linear ('trapezium-like') Lagrange quadrature goes up to 4.
In the same way, the Simpson rule, which is based on three points, is of order 4, but by a clever rearrangement of the supporting points the order can be raised to 6.
Generally, for each $n \in \mathbb{N}$ there exists a (unique) set of n points inside $[a, b]$, such that the Lagrange quadrature based on those points is of order $2n$. No higher order is achieveble on n points.
We shall not prove these statements here, but refer to the literature. Let us just note that the supporting points used by Gauß are the zeroes of the n-th Legendre orthogonal polynomial on $[a, b]$.

Exercise 8. Determine the weights c_0, c_1 and c_2 of the Lagrange quadrature rule on the interval $[-1, 1]$, based on the points $\xi_0 := -\sqrt{\frac{3}{5}}$, $\xi_1 := 0$, and $\xi_2 := \sqrt{\frac{3}{5}}$. What is the order p of this quadrature rule? (Prove your answer.) Give a sharp upper bound for the error of this rule in terms of $\left\|f^{(p)}\right\|$ and $b - a$.

5. Ordinary Differential Equations

In this chapter we shall treat numerical methods for the solution of ordinary differential equations, i.e. differential equations in one variable. In a sense this is a generalisation of the previous chapter on numerical integration.
We are given a continuous function $f : [a, b] \times \mathbb{R} \to \mathbb{R}$ and a real number y_0, and we are looking for solutions $\varphi : [a, b] \to \mathbb{R}$ of the differential equation

$$\varphi'(x) = f(x, \varphi(x)) \tag{4}$$

with initial condition $\varphi(a) = y_0$.
The feature that complicates this situation considerably in comparison with Chapter 4 is the dependence of f on the second coordinate, i.e. on φ itself.

Let us first note that equation (4) can alternatively be written as an integral equation:

$$\varphi(x) = y_0 + \int_a^x f(u, \varphi(u))\, du\ . \tag{5}$$

Now the 'new' value $\varphi(x)$ is expressed in terms of 'earlier' values $\varphi(u)$, and in order to evaluate it we may apply the integration techniques of Chapter 4 to the right hand side, keeping in mind that we should only use data that have already been calculated.

5.1. Euler's Method

The first idea that comes to mind is to use *first* order quadrature. This leads to the oldest and most primitive method, pioneered by Euler. For a positive integer n we keep the standard notation of last chapter:

$$h := \frac{b-a}{n} \quad \text{and} \quad x_i := a + ih\ , \quad (i = 0, \ldots, n)\ .$$

Proposition 9. *Let $f : [a, b] \times \mathbb{R} \to \mathbb{R}$ be differentiable with continuous and bounded derivatives. Define a sequence $w_0, w_1, w_2, \ldots, w_n$ by $w_0 := y_0$ and*

$$w_{i+1} := w_i + hf(x_i, w_i)\ .$$

Then w_i is an approximation of $y_i := \varphi(x_i)$ satisfying

$$|y_i - w_i| \le \frac{b-a}{2n} \cdot M \cdot \frac{e^{(x_i-a)L} - 1}{L}\ ,$$

where

$$L := \left\| \frac{\partial f}{\partial y} \right\| \text{ and } M := \left\| \frac{\partial f}{\partial x} \right\| + \|f\| \cdot \left\| \frac{\partial f}{\partial y} \right\|\ .$$

Of all the details in this statement the most important ones are the simple rule by which the Euler approximations are produced and the factor $1/n$ in the estimate, which shows that convergence to the true solution is very slow.
For the proof we need the following lemma.

Lemma 10. *Let $a_0, a_1, a_2, \ldots$ be a sequence of positive numbers satisfying*

$$a_{i+1} \leq \gamma a_i + \beta \ .$$

For some $\alpha, \gamma > 0$. Then for all $i \in \mathbb{N}$:

$$a_i \leq \gamma^i a_0 + \beta \frac{\gamma^i - 1}{\gamma - 1} \ .$$

This can be by induction.

Proof of Proposition 9. From the general theory of differential equations we know that (4) allows a unique solution, provided that f is Lipschitz continuous in y for all x. This is guaranteed by the boundedness of $\frac{\partial f}{\partial y}$.
Note furthermore that

$$\varphi''(x) = \frac{df(x, \varphi(x))}{dx} = \frac{\partial f}{\partial x}(x, \varphi(x)) + \frac{\partial f}{\partial y}(x, \varphi(x)) \cdot f(x, \varphi(x)) \ ,$$

so that $|\varphi''(x)| \leq M$. By Taylor's Theorem we have, for some $\xi \in (x_i, x_{i+1})$,

$$\varphi(x_{i+1}) = \varphi(x_i) + h\varphi'(x_i) + \tfrac{1}{2}h^2\varphi''(\xi) \ .$$

Therefore

$$y_{i+1} - w_{i+1} = \big(y_i - w_i\big) + h\big(f(x_i, y_i) - f(x_i, w_i)\big) + \tfrac{1}{2}h^2\varphi''(\xi) \ ,$$

so that

$$|y_{i+1} - w_{i+1}| \leq |y_i - w_i|(1 + hL) + \tfrac{1}{2}h^2 M \ .$$

The statement follows from Lemma 10 if we observe that

$$(1 + hL)^i \leq e^{(x_i - a)L} \ .$$

□

5.2. Runge Kutta Methods of Second Order

By using a higher order quadrature the approximation method can be inmproved. The relation

$$y_{i+1} = y_i + \int_{x_i}^{x_i + h} f(u, \varphi(u))\, du$$

can be approximated by the 'Ansatz'

$$w_{i+1} = w_i + c_0 k_0 + c_1 k_1 \ ,$$

where

$$k_0 := hf(x_i, w_i) \quad \text{and} \quad k_1 := hf(x_i + \xi h, w_i + \xi k_0) \ .$$

If we choose the values of c_0, c_1 as the weights of a Lagrange quadrature supported by the points 0 and $\xi \in [0, 1]$,

$$c_0 = 1 - \frac{1}{2\xi} \quad \text{and} \quad c_1 = \frac{1}{2\xi} \ ,$$

then the above scheme yields approximations w_i to $\varphi(x_i)$ that are correct to second order:

$$|w_i - \varphi(x_i)| \leq \frac{C}{n^2} \ .$$

In this second order Runge-Kutta scheme ξ is a free parameter. Some particular values have acquired special names:

$$\begin{aligned} \xi = 1 : &\qquad \text{Improved Euler Method;} \\ \xi = \tfrac{2}{3} : &\qquad \text{Heun's Method;} \\ \xi = \tfrac{1}{2} : &\qquad \text{Midpoint Method.} \end{aligned}$$

5.3. Runge Kutta Method of Fourth Order

The most popular Runge-Kutta scheme, for many *the* Runge Kutta Method, is the following 4-th order method for the solution of equation (4), to be executed n times on the interval $[a, b]$:

$$\begin{aligned} k_0 &= hf(x_i, w_i) \ ; \\ k_1 &= hf(x_i + \tfrac{1}{2}h, w_i + \tfrac{1}{2}k_0) \ ; \\ k_2 &= hf(x_i + \tfrac{1}{2}h, w_i + \tfrac{1}{2}k_1) \ ; \\ k_3 &= hf(x_i + h, w_i + k_2) \ ; \\ w_{i+1} &= w_i + \tfrac{1}{6}\big(k_0 + 2k_1 + 2k_2 + k_3\big) \ . \end{aligned}$$

It is one of a large family of fourth order methods, which in its turn is part of a hierarchy of methods of all orders. This particular one combines a relatively simple Simpson-type calculation based on only 4 function evaluations, with fast convergence:

$$|w_i - \varphi(x_i)| \leq \frac{C}{n^4} \ .$$

Exercise 9. Consider the differential equation

$$\varphi'(x) = x - \varphi(x)$$

on the interval $[0, 4]$ with initial condition $\varphi(0) = 3$. Calculate the exact solution and compare it with the following approximations:

(a) Euler's Method;
(b) Heun's Method;
(c) the Midpoint Method;
(d) the 4-th order Runge-Kutta scheme.

How many iterations do you need in each case in order to approximate $\varphi(4)$ to within 10^{-3}?

6. Gaussian Elimination and Matrix Factorisation

In this chapter we consider solution strategies for equations of the form

$$Ax = b \ ,$$

where A is a real $n \times n$ matrix and both b and the unknown x are column vectors of lenght n. The solution exists and is unique if and only if A is regular, i.e. $\det(A) = 0$.
There is a variety of ways to solve this equation, exactly or approximately. Cramer's rule, for one, yields an answer in principle, but is in practice only useful for small values of n, since the number of operations required grows as $n!$. Gaussian elimination is a much more efficient method, taking roughly $\frac{2}{3}n^3$ operations. We shall discuss Gaussian elimination and some of its refinements in special cases. It is a *direct* method in the sense that, apart from rounding off errors, it is exact.
For large n, it may be profitable to use an approximate, iterative method, which in each step uses about n^2 operations. This makes sense if the number of iterations required is of the order of n or less.

6.1. *LU*-FACTORISATION

We recall that a matrix L is called *lower triangular* if it is of the form

$$L = \begin{pmatrix} l_{11} & & \mathbf{0} \\ \vdots & \ddots & \\ l_{n1} & \cdots & l_{nn} \end{pmatrix} \ ,$$

i.e. if all nonzero entries lie on or below the diagonal. Similarly, we call a matrix U *upper diagonal* if all nonzero entries lie on or above the diagonal. Clearly, the determinant of such matrices is the product of their diagonal entries, so that they are regular iff these are all nonzero. All regular lower triangular matrices form a group $\mathcal{L}$; the regular upper triangular matrices form a group $\mathcal{U}$.
We shall now show that Gaussian elimination (without row permutation) is actually factorisation into a lower and an upper triangular matrix.
By E_{ij} we denote the $n \times n$-matrix having a 1 at the ij-position, and 0 everywhere else.

Theorem 11. (*LU*-factorisation) *Let A be a regular $n \times n$ matrix. Suppose that A can be transformed into an upper triangular matrix U by Gaussian elimination. Then*

$$A = LU \ ,$$

with

$$L = \left(\mathbb{1} + \sum_{i>j} \lambda_{ij} E_{ij} \right) \ ,$$

where λ_{ij} is the multiplier used when cleaning the j-th column with respect to the i-th row (and $\mathbb{1}$ denotes the identity matrix).

Proof. Let $A^{(1)} := A$ be the original matrix

$$\begin{pmatrix} a_{11}^{(1)} & \cdots & a_{1n}^{(1)} \\ \vdots & & \vdots \\ a_{n1}^{(1)} & \cdots & a_{nn}^{(1)} \end{pmatrix} \ .$$

Let $\lambda_{i1} := a_{i1}^{(1)}/a_{11}^{(1)}$ for $2 \leq i \leq n$. Then 'sweeping clean' the first column in the downward direction is described by the equation

$$A^{(2)} := (\mathbb{1} - \lambda_{n1}E_{n1}) \cdots (\mathbb{1} - \lambda_{31}E_{31})(\mathbb{1} - \lambda_{21}E_{21})A^{(1)} = \begin{pmatrix} a_{11}^{(2)} & a_{12}^{(2)} & \cdots & a_{1n}^{(2)} \\ 0 & a_{22}^{(2)} & \cdots & a_{2n}^{(2)} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & a_{n2}^{(2)} & \cdots & a_{nn}^{(2)} \end{pmatrix} .$$

Subsequently we define $\lambda_{i2} := a_{i2}^{(2)}/a_{22}^{(2)}$ for $3 \leq i \leq n$ and

$$A^{(3)} := (\mathbb{1} - \lambda_{n2}E_{n2}) \cdots (\mathbb{1} - \lambda_{42}E_{42})(\mathbb{1} - \lambda_{32}E_{32})A^{(2)} = \begin{pmatrix} a_{11}^{(3)} & a_{12}^{(3)} & a_{13}^{(3)} & \cdots & a_{1n}^{(3)} \\ 0 & a_{22}^{(3)} & a_{23}^{(3)} & \cdots & a_{2n}^{(3)} \\ 0 & 0 & a_{32}^{(3)} & \cdots & a_{3n}^{(3)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & a_{n3}^{(3)} & \cdots & a_{nn}^{(3)} \end{pmatrix} .$$

If this can be continued without ever hitting on a division by zero (i.e. $a_{kk}^{(k)} = 0$ at some stage k), we eventually arrive at an upper triangular matrix $U := A^{(n)}$, which is related to A by the equation $U = L_{n-1}L_{n-2} \cdots L_1 A$, where

$$L_k := (\mathbb{1} - \lambda_{nk}E_{nk}) \cdots (\mathbb{1} - \lambda_{k+1,k}E_{k+1,k}) , \qquad (1 \leq k \leq n-1) .$$

The rules of calculation in the group $\mathcal{L}$ now yield the factorisation $A = LU$, where $L = L_1^{-1}L_2^{-1} \cdots L_{n-1}^{-1}$ is given by the expression in the theorem. □

In general it may happen that in the above calculation a *pivot* element $a_{kk}^{(k)}$ is found to be zero. In that case Gaussian elimination requires that a permutation of rows is applied in order to put a nonzero element in its place. This permutation of rows is realised by a permutation matrix P, i.e. a matrix having precisely one 1 in every row and in every column, and zeroes everywhere else. In other words: an arbitrary regular matrix A satisfies $PA = LU$, so it can be cast in the form $A = P^*LU$, where $L \in \mathcal{L}$, $U \in \mathcal{U}$ and $P \in \mathcal{P}$, the group of permutation matrices.
The Matlab command

```
[L,U,P]=lu(A)
```

returns a lower triangular matrix L, upper triangular matrix U, and permutation matrix P such that $PA = LU$.

Remark. Suppose no exchanges of rows are needed in the elimination process. Then A can be factorised as $A = LU$ with $L \in \mathcal{L}$, $U \in \mathcal{U}$. Let us see in how far this factorisation is unique. If D is a regular diagonal matrix, then $L' := LD$ and $U' := D^{-1}U$ form another decompostion of A. In this way decompositions can be constructed having arbitrary diagonal entries for L. Conversely, if a second decomposition $A = L'U'$ exists, then $LU = L'U'$, so that $U(U')^{-1} = L^{-1}L'$. These products must be at the same time lower triangular and upper triangular, hence they are given by a diagonal matrix D, and we find $L' = LD$ and $U' = D^{-1}U$. This D is fixed by, for instance, the requirement that L has diagonal entries 1.

Proposition 12. *Let A be a regular matrix. Then the following are equivalent.*

1. *A can be transformed by Gaussian elimination into upper triangular form without exchange of rows;*
2. *$A = LU$ for some $L \in \mathcal{L}$, $U \in \mathcal{U}$;*
3. *the determinant of each main minor of A is nonzero.*

We omit the proof.
The decomposition of A as LU costs approximately $\frac{2}{3}n^3$ operations.

6.2. Pivoting

In numerical anlysis it is sometimes advisable to apply row exchange not only if a pivot element $a_{kk}^{(k)}$ is zero, but also if it is small. Indeed, division by these small numbers may cause unacceptable errors in the calculation.
A popular strategy to avoid division by small numbers as much as possible, is *column pivoting*, i.e. searching for the largest entry in each column (in absolute value), and interchanging it with the row whose 'turn' it is to serve as a pivot. An more radical approach is *total pivoting*, looking for the largest entry in the whole block still to be treated, and then interchanging two rows and two columns in order to put this entry into the next pivot position. Each method has its advantages and disadvantages, and it is not known if there exists a strategy that serves best under all conditions.

6.3. Special cases of *LU*-factorisation

Theorem 13. (Sylvester) *A symmetric matrix is strictly positive definite iff the determinant of each main minor is strictly positive.*

In particular, every strictly positive definite matrix A has an LU factorisation. In fact, it has many. The following theorem states, however, that one of these is particularly nice.

Theorem 14. (Cholesky decomposition) *Let A be a strictly positive definite $n \times n$ matrix. Then there exists a unique lower triangular matrix L such that $A = LL^*$and $l_{ii} > 0$ for $i = 1, \ldots, n$.*

We do not prove these theorems from linear algebra here. The Cholesky decomposition is found in approximately $\frac{1}{3}n^2$ operations.

6.4. Iterative Methods

If n is very large, even Gaussian elimination may be too much work. In that case we may have to resort to approximation of the solution by iteration. This requires that we write our solution as the fixed point of some contraction. One such method is that of Jacobi.

Definition. A complex $n \times n$ matrix is called *strictly diagonally dominated* if $|a_{ii}| > \sum_{j=1, j\neq i}^{n} |a_{ij}|$ for $i = 1, \ldots n$.

Lemma 15. *A strictly diagonally dominated matrix is regular and has no zeroes on its diagonal.*

Proof. The second point is obvious. For the first, suppose that $Ax = 0$. Then in particular $(Ax)_i = 0$, i.e. $a_{ii}x_i = -\sum_{j=1,j\neq i}^{n} a_{ij}x_j$. Taking absolute values we find that $|a_{ii}| \|x\|_\infty \leq \sum_{j=1,j\neq i}^{n} |a_{ij}| \|x\|_\infty$. This contradicts the assumption unless $\|x\|_\infty = 0$, and therefore $x = 0$. □

Theorem 16. *(***Jacobi Method***) Let A be strictly diagonally dominated. Write $A = D + O$, with D diagonal and O off-diagonal (i.e. $O_{ii} = 0$ for all i). For a given b and starting point x^0 the sequence $x^0, x^1, x^2, \ldots$, defined by*

$$x^{k+1} := D^{-1}\left(-Ox^k + b\right)$$

converges to the solution of the equation $Ax = b$. Moreover, if we define

$$\vartheta := \max_i \sum_{j\neq i} \frac{|a_{ij}|}{|a_{ii}|} ,$$

then the error in the approximation x^k satisfies

$$\left\|x^k - x^\infty\right\| \leq \frac{\vartheta}{1-\vartheta} \left\|x^k - x^{k-1}\right\| .$$

Proof. The matrix $-D^{-1}O$ is a contraction with norm ϑ, which is less than 1 by the assumption of diagonal dominance. By a similar argument as in Theorem 1 it follows that the sequence converges to the fixed point of the map $x \mapsto D^{-1}\left(-Ox + b\right)$, which is our solution.
Moreover we have

$$\left\|x^k - x^\infty\right\|_\infty \leq \sum_{j=k}^{\infty} \left\|x^{j+1} - x^j\right\| \leq \sum_{j=k}^{\infty} \vartheta^{j+1-k} \left\|x^k - x^{k-1}\right\| = \frac{\vartheta}{1-\vartheta} \left\|x^k - x^{k-1}\right\| .$$

□

The following refinement of the Jacobi method widens the class of admissible A's in a useful way, speeding up the convergence for the A's we already had. We state the theorem without proof.

Theorem. *(***Gauß-Seidel***) Suppose that A is strictly diagonally dominated or strictly positive definite. For a given b and starting point x^0 the sequence $x^0, x^1, x^2, \ldots$, defined by*

$$x_i^{k+1} = \frac{1}{a_{ii}} \left(-\sum_{j=1}^{i-1} a_{ij}x_j^{k+1} - \sum_{j=i+1}^{n} a_{ij}x_j^k + b_i \right)$$

converges to the solution of the equation $Ax = b$.

Exercise 10. Suppose $A = P^*LU$, where P is a permutation matrix, L a lower triangular matrix with ones on the diagonal, and U upper triangular.

(a) Count the number of operations needed to compute the factorisation $A = P^*LU$ for a given matrix A.
(b) Count the number of operations needed to calculate $\det(A)$ by factoring.
(c) Compute $\det(A)$ and count the number of operations when

$$A = \begin{pmatrix} 0 & 2 & 1 & 4 & -1 & 3 \\ 1 & 2 & -1 & 3 & 4 & 0 \\ 0 & 1 & 1 & -1 & 2 & -1 \\ 2 & 3 & -4 & 2 & 0 & 5 \\ 1 & 1 & 1 & 3 & 0 & 2 \\ -1 & -1 & 2 & -1 & 2 & 0 \end{pmatrix}.$$

Exercise 11. Let A be the matrix given in exercise 10. Find the solution of the equation $Ax = b$, where $b = (2, 3, 0, 1, 1, 3)^T$, in two ways:

(a) By factorisation;
(b) by premultiplying the equation on both sides with A^*, and using Gauß-Seidel iteration on the positive definite matrix $B := A^*A$. What is the number of iterations needed for an accuracy of 10^{-3} in each component?