An Introduction to Scientific Computing

CARL CHRISTIAN KJELGAARD MIKKELSEN

Department of Computing Science, Umeå University

Contents

Chapter 1

Introduction

This textbook is a work in progress. Be advised that the text contains typographical errors and that new sections and problems will be added continuously. The goal is to create a single text suitable for the course 5DV005. Definitions, lemmas, theorems, and corollaries are numbered by chapter using the same counter. `MATLAB` scripts and functions are enclosed in grey boxes. The names of `MATLAB` functions are written using a typewriter font, ex. `newton`. All named functions are either standard functions or available through the course repository.

Chapter 2

Motivating examples

2.1 Basic examples

This section contains three examples which illustrates the profound difference between exact arithmetic and finite precision arithmetic. Most users are blissfully unaware of these differences and are incapable of writing programs which perform even simple computations in a reliable manner.

Example 2.1 The average of two real number a and b is given by

$$\mu = \frac{a+b}{2}. \tag{2.1}$$

Compute the average of $a = 6.377$ and $b = 6.379$ using exact arithmetic. What is the computed average if each intermediate result is rounded to 4 significant figures?

Solution The exact sum s of a and b is $s = a + b = 12.756$. The exact average is $\mu = 6.378$. However, if each intermediate result is rounded to 4 significant figures, then the computed sum is $\hat{s} = 12.76$ and the computed average is $\hat{\mu} = 6.380$. The computed average is larger than both a and b! ■

The next example shows that the computed result of a sum depends on the summation order. This is radically different from exact arithmetic where the sum is independent of the order of the terms.

Example 2.2 The harmonic numbers H_n are given by

$$H_n = \sum_{j=1}^{n} \frac{1}{j}. \tag{2.2}$$

Compute H_n for $n = 2^{22}$ using single precision arithmetic. Sum the terms in both increasing and decreasing order and compare the results.

Solution We can sum the terms in decreasing order as follows:

```
>> h=single(0); for j=1:2^22 h=h+1/j; end
>> h
h =

  single

   15.4037
```

By default, MATLAB does all computations using double precision floating point arithmetic. Initializing the variable `h` using the `single` command forces the summation to proceed in single precision. We can sum the terms in increasing order as follows:

```
>> h=single(0); for j=2^22:-1:1 h=h+1/j; end
>> h
h =

  single

   15.8296
```

It is clear that the two results are very different! The exact result rounded to 6 significant figures is $H_n \approx 15.8265$. ■

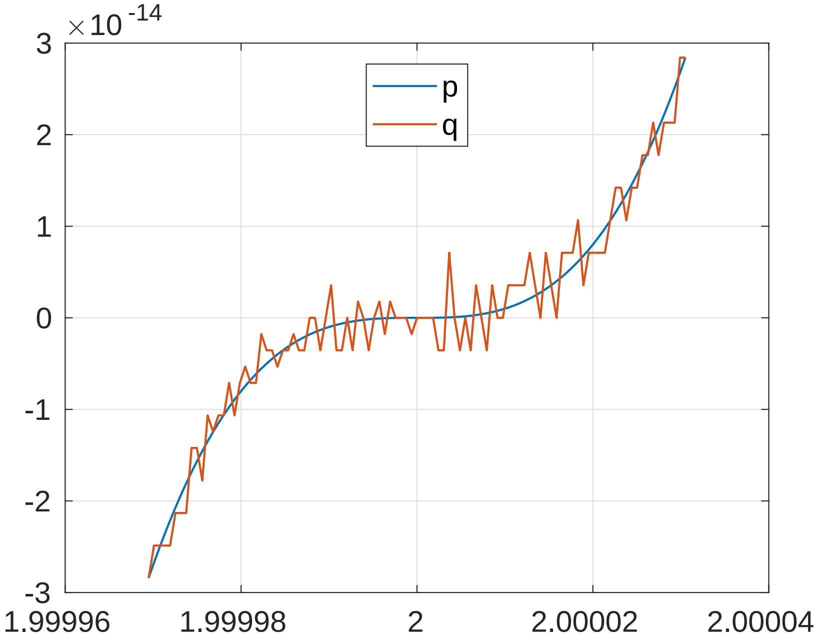

Figure 2.1: The result obtained by plotting two different representations of polynomial $p(x) = (x-2)^3$.

The third example show that it be difficult to evaluate even simple functions such as polynomials.

Example 2.3 Consider the polynomials p and q given by

$$p(x) = (x-2)^3, \quad q(x) = x^3 - 6x^2 + 12x - 8 \tag{2.3}$$

Show that $p(x) = q(x)$ and plot both polynomials in a small interval around zero using MATLAB.

Solution A direct calculation establishes that $p(x) = q(x)$. The following commands can be used to define and plot the polynomials using MATLAB:

```
>> p=@(x)(x-2).^3;
>> q=@(x)x.^3-6*x.^2+12*x-8;
>> x=linspace(-1,1,101)*2^(-15)+2;
>> plot(x,p(x),x,q(x),’Linewidth’,1); grid;
>> legend(’Location’,’North’,’p’,’q’);
```

The plot can be found as Figure 2.1. It is interesting to note that the graph of p is a smooth curve just as we expect, while the graph of q is very erratic and seems to have many zeros other than $x = 2$.

■

Exercises

1. Compute the average μ of the numbers $a = 6.371$ and $b = 6.373$ using equation (2.1). Compare the result μ to the value $\hat{\mu}$ obtained when the result of each arithmetic operation is rounded to 4 significant figures.
2. Compute the average μ of the numbers $a = 6.378$ and $b = 6.378$ using equation (2.1). Compare the result μ to the value $\hat{\mu}$ obtained when the result of each arithmetic operation is rounded to 4 significant figures.

Computer problems

1. Write a `C` program which computes the harmonic numbers H_n given by equation (2.2). The call sequence should be `./harmonic <n> <direction>` where `n` is the number of terms and `direction` which controls the summation order. The terms should be summed in decreasing (increasing) order if `direction` equals 1 (-1).
2. Consider the polynomials $p_i : \mathbb{R} \to \mathbb{R}$ given by

$$p_1(x) = x^4 - 4x^3 + 6x^2 - 4x + 1, \tag{2.4}$$
$$p_2(x) = (((x-4)x+6)x-4)x+1, \tag{2.5}$$
$$p_3(x) = (x-1)^4 \tag{2.6}$$

   (a) Show that $p_1(x) = p_2(x) = p_3(x)$ in exact arithmetic.

   (b) Plot the graph of three polynomials in a small neighborhood of $x = 1$. Example 2.3 illustrates how to control the interval. Use the `subplot` command to create a separate subfigure for each expression. It can be difficult to find a good interval, so feel free to experiment, but do consider the choice of $k = 20$ and $n = 2^8 + 1$.

   (c) Which of the three formulas gives the best representation of the polynomial?

   (d) What information is encoded in the expression for p_1 and p_2, but can be read off immediately from expression for p_3?

Chapter 3

Mathematical prerequisites

Contents

This chapter contains a brief review of many of the theorems which will be used to derive new results during the course. Moreover, many of the results can be used to generate test cases for numerical programs. The proofs and examples demonstrate how to write proofs which can not be faulted.

3.1 Inequalities

Theorem 3.1 Cauchy-Schwartz inequality *Let* $\mathbf{x}, \mathbf{y} \in \mathbb{R}^\mathbf{n}$*. Then the inner product* $x^T y$ *satisfies*

$$|\mathbf{x^T y}| \leq \|\mathbf{x}\|_\mathbf{2} \|\mathbf{y}\|_\mathbf{2}$$

Proof. If $\mathbf{y} = \mathbf{0}$, then the result is immediate, so assume $\mathbf{y} \neq \mathbf{0}$. Let

$$p(t) = \|\mathbf{x} + \mathbf{ty}\|_\mathbf{2}^\mathbf{2}.$$

A direct calculation shows that

$$p(t) = \|\mathbf{x}\|^\mathbf{2} + \mathbf{2x^T yt} + \|\mathbf{y}\|_\mathbf{2}^\mathbf{2}\mathbf{t}^\mathbf{2}$$

is a polynomial of degree 2 in the variable t. Since $p(t) \geq 0$, the discriminant of p must be less than or equal to zero, i.e.,

$$4(\mathbf{x^T y})^\mathbf{2} - \mathbf{4}\|\mathbf{x}\|_\mathbf{2}^\mathbf{2}\|\mathbf{y}\|_\mathbf{2}^\mathbf{2} \leq \mathbf{0}.$$

This implies

$$|\mathbf{x^T y}| \leq \|\mathbf{x}\|_\mathbf{2}\|\mathbf{y}\|_\mathbf{2}.$$

This completes the proof. ■

Theorem 3.2 The triangular inequality *Let* $\mathbf{x}, \mathbf{y} \in \mathbb{R}^\mathbf{n}$*. Then*

$$\|\mathbf{x} + \mathbf{y}\|_\mathbf{2} \leq \|\mathbf{x}\|_\mathbf{2} + \|\mathbf{y}\|_\mathbf{2}.$$

Proof. Let $\mathbf{x}, \mathbf{y} \in \mathbb{R}^\mathbf{n}$. Then by applying Cauchy-Schartz's inequality we find

$$\begin{aligned}\|\mathbf{x}+\mathbf{y}\|_\mathbf{2}^\mathbf{2} = (\mathbf{x}+\mathbf{y})^\mathbf{T}(\mathbf{x}+\mathbf{y}) = \mathbf{x}^\mathbf{T}\mathbf{x}+\mathbf{x}^\mathbf{T}\mathbf{y}+\mathbf{y}^\mathbf{T}\mathbf{x}+\mathbf{y}^\mathbf{T}\mathbf{y} &= \|\mathbf{x}\|_\mathbf{2}^\mathbf{2} + \mathbf{2}\mathbf{x}^\mathbf{T}\mathbf{y} + \|\mathbf{y}\|_\mathbf{2}^\mathbf{2} \\ &\le \|\mathbf{x}\|_\mathbf{2}^\mathbf{2} + \mathbf{2}|\mathbf{x}^\mathbf{T}\mathbf{y}| + \|\mathbf{y}\|_\mathbf{2}^\mathbf{2} \\ &\le \|\mathbf{x}\|_\mathbf{2}^\mathbf{2} + \mathbf{2}\|\mathbf{x}\|_\mathbf{2}\|\mathbf{y}\|_\mathbf{2} + \|\mathbf{y}\|_\mathbf{2}^\mathbf{2} = (\|\mathbf{x}\|_\mathbf{2} + \|\mathbf{y}\|_\mathbf{2})^\mathbf{2}.\end{aligned}$$

Since the square root function is monotone increasing, we conclude

$$\|\mathbf{x}+\mathbf{y}\|_\mathbf{2} \le \|\mathbf{x}\|_\mathbf{2} + \|\mathbf{y}\|_\mathbf{2}.$$

This completes the proof. ■

Corollary 3.1 *Let* $\mathbf{x}, \mathbf{y} \in \mathbb{R}^\mathbf{n}$. *Then*

$$|\|\mathbf{x}\|_\mathbf{2} - \|\mathbf{y}\|_\mathbf{2}| \le \|\mathbf{x}-\mathbf{y}\|_\mathbf{2}.$$

Proof. Let $\mathbf{x}, \mathbf{y} \in \mathbb{R}^\mathbf{n}$. Then

$$\mathbf{x} = (\mathbf{x}-\mathbf{y}) + \mathbf{y}$$

which implies

$$\|\mathbf{x}\|_\mathbf{2} \le \|\mathbf{x}-\mathbf{y}\|_\mathbf{2} + \|\mathbf{y}\|_\mathbf{2}$$

or equivalently

$$\|\mathbf{x}\|_\mathbf{2} - \|\mathbf{y}\|_\mathbf{2} \le \|\mathbf{x}-\mathbf{y}\|_\mathbf{2}.$$

By swapping $\mathbf{x}$ and $\mathbf{y}$ we see that

$$\|\mathbf{y}\|_\mathbf{2} - \|\mathbf{x}\|_\mathbf{2} \le \|\mathbf{y}-\mathbf{x}\|_\mathbf{2}$$

We conclude that not only is

$$\|\mathbf{x}\|_\mathbf{2} - \|\mathbf{y}\|_\mathbf{2} \le \|\mathbf{x}-\mathbf{y}\|_\mathbf{2}$$

but we also have

$$-(\|\mathbf{x}\|_\mathbf{2} - \|\mathbf{y}\|_\mathbf{2}) \le \|\mathbf{x}-\mathbf{y}\|_\mathbf{2}$$

It follows that

$$|\|\mathbf{x}\|_\mathbf{2} - \|\mathbf{y}\|_\mathbf{2}| \le \|\mathbf{x}-\mathbf{y}\|_\mathbf{2}$$

This completes the proof. ■

3.2 Mathematical induction

Mathematical induction is standard method for proving statements which involve the natural numbers. It is often used to prove that algorithms and computer programs are correct. Everyone should be able to apply this technique correctly. In this section we review the principle of mathematical induction and demonstrate how to apply it correctly.

The set of natural numbers $\mathbb{N}$ is defined as a set which satisfies (among others) the following axiom, known as the principle of mathematical induction.

Axiom 3.1 Principle of mathematical induction *If* $V \subseteq \mathbb{N}$ *satisfies the following properties*

1. $1 \in V$, *and*
2. $n \in V$ *implies that* $n+1 \in V$,

then $V = \mathbb{N}$.

The following examples show how to correctly apply the principle of mathematical induction to prove statements.

Example 3.1 Let $n \in \mathbb{N}$. Show that

$$\sum_{j=1}^{n} j = \frac{n(n+1)}{2}.$$

Solution Let $V \subseteq \mathbb{N}$ be given by

$$V = \left\{ n \in \mathbb{N} \, : \, \sum_{j=1}^{n} j = \frac{n(n+1)}{2} \right\}.$$

Our goal is to show that $V = \mathbb{N}$. It is clear that $1 \in V$ because

$$\sum_{j=1}^{1} j = 1 = \frac{1(1+1)}{2}.$$

Now assume that $n \in V$, i.e.,

$$\sum_{j=1}^{n} j = \frac{n(n+1)}{2}.$$

We have to show that $n+1 \in V$, i.e.,

$$\sum_{j=1}^{n+1} j = \frac{(n+1)(n+2)}{2}.$$

However this is easy, because by definition

$$\sum_{j=1}^{n+1} j = \left(\sum_{j=1}^{n} j \right) + (n+1),$$

and by applying our assumption ($n \in V$) we discover

$$\sum_{j=1}^{n+1} j = \frac{n(n+1)}{2} + (n+1) = \frac{n(n+1) + 2(n+1)}{2} = \frac{(n+1)(n+2)}{2},$$

and we have proved that $n+1 \in V$. By the principle of mathematical induction we conclude $V = \mathbb{N}$. This completes the proof. ■

Example 3.2 Let $n \in \mathbb{N}$. Show that

$$\sum_{j=1}^{n} j^2 = \frac{n(n+1)(2n+1)}{6}.$$

Solution Let $V \subseteq \mathbb{N}$ be given by

$$V = \left\{ n \in \mathbb{N} \, : \, \sum_{j=1}^{n} j^2 = \frac{n(n+1)(2n+1)}{6} \right\}.$$

Our goal is to show that $V = \mathbb{N}$. It is clear that $1 \in V$ because

$$\sum_{j=1}^{1} j^2 = 1 = \frac{1(1+1)(2+1)}{6} = \frac{1 \cdot 2 \cdot 3}{6}.$$

Now assume that $n \in V$, i.e.,

$$\sum_{j=1}^{n} j^2 = \frac{n(n+1)(2n+1)}{6}.$$

We have to show that $n+1 \in V$, i.e.,

$$\sum_{j=1}^{n+1} j^2 = \frac{(n+1)(n+2)(2n+3)}{6}. \tag{3.1}$$

We now simplify the left hand side of equation (3.1) using our assumption ($n \in V$) and find that

$$\begin{aligned}\sum_{j=1}^{n+1} j^2 &= \left(\sum_{j=1}^{n} j^2\right) + (n+1)^2 = \frac{n(n+1)(2n+1) + 6(n+1)^2}{6} \\ &= \frac{(n+1)\left[n(2n+1) + 6(n+1)\right]}{6} = \frac{(n+1)\left(2n^2+7n+6\right)}{6}.\end{aligned}$$

We now simplify the right hand side of equation (3.1) and find

$$\frac{(n+1)(n+2)(2n+3)}{6} = \frac{(n+1)(2n^2+7n+6)}{6}$$

We conclude that that equation (3.1) is true and $n+1 \in V$. By the principle of mathematical induction, we conclude $V = \mathbb{N}$. This completes the proof. ■

Example 3.3 Derive an algorithm for computing $n!$ and prove that it is correct.

Solution By definition, $1! = 1$ and $n! = (n-1)!n$. This suggests the following Algorithm 1.

Algorithm 1 Factorial

```
$r \leftarrow 1$
for $j = 2, 3, \ldots, n$ do
    $r \leftarrow r \cdot j$
end for
return $r$
```

It remains to prove that this algorithm is correct. Let

$$V = \{n \in \mathbb{N} \mid \text{Algorithm 1 computes } n! \text{ correcly}\}.$$

Then $V \subseteq \mathbb{N}$. We have $1 \in V$, because the algorithm returns $r = 1$ when $n = 1$. Now suppose, that $n \in V$. We have to show that $n+1 \in V$. By unrolling the last iteration of the main loop, we see that the algorithm first computes $r = n!$ (because $n \in V$), then performs the final update $r \leftarrow r \cdot (n+1)$. Therefore, the return value is $r = (n+1)!$ and $n+1 \in V$. By the principle of mathematical induction, we conclude that $V = \mathbb{N}$ and the algorithm is correct for all $n \in \mathbb{N}$. ■

3.3 Sequences and series

The following theorem is useful in the analysis of a variety of topics, including the error analysis of polynomials, functional iterations and the solution of ordinary differential equations.

Theorem 3.3 *Let $\{x_j\}_{j=0}^{\infty}$ be any sequence which satisfies*

$$|x_{j+1}| \leq \alpha |x_j| + \beta, \quad j = 0, 1, 2, \ldots .$$

Then

$$|x_n| \leq \alpha^n |x_0| + \beta \sum_{j=0}^{n-1} \alpha^j, \quad n = 1, 2, \ldots .$$

Proof. Let $V \subseteq \mathbb{N}$ be given by

$$V = \left\{ n \in \mathbb{N} \mid |x_n| \leq \alpha^n |x_0| + \sum_{j=0}^{n-1} \beta \right\}.$$

We will show $V = \mathbb{N}$ using the principle of mathematical induction. It is clear that $1 \in V$ because $|x_1| \leq \alpha|x_0| + \beta$. Now, assume that $n \in V$. We have to show that $n+1 \in V$. By assumption, we have

$$|x_{n+1}| \leq \alpha|x_n| + \beta,$$

and since $n \in V$ we can estimate

$$|x_{n+1}| \leq \alpha \left(\alpha^n |x_0| + \beta \sum_{j=0}^{n-1} \alpha^j \right) + \beta = \alpha^{n+1}|x_0| + \alpha \sum_{j=0}^{n} \beta^j.$$

This shows that $n+1 \in V$. By the principle of mathematical induction $V = \mathbb{N}$. This completes the proof. ■

Theorem 3.4 *The geometric series $s = \sum_{j=0}^{\infty} q^j$ is convergent with sum $s = \frac{1}{1-q}$ if and only if $|q| < 1$.*

Proof. If $q = 1$ then the series is clearly divergent, so assume that $q \neq 1$. We will now calculate the partial sum $s_n = \sum_{j=0}^{n-1} q^j$. We have

$$(1-q)s_n = (1-q)\sum_{j=0}^{n-1} q^j = \sum_{j=0}^{n-1} q^j - \sum_{j=0}^{n-1} q^{j+1} = \sum_{j=0}^{n-1} q^j - \sum_{j=1}^{n} q^j = 1 - q^n,$$

because the majority of the terms cancel. Since $q \neq 1$, we conclude

$$s_n = \frac{1-q^n}{1-q}.$$

It follows that the sequence of partial sums is convergent if and only if the sequence $\{q^n\}_{n=1}^{\infty}$ is convergent, i.e., if and only if $|q| < 1$, and the limit is $s = \frac{1}{1-q}$. ■

Theorem 3.5 **Integral test of convergence** *Let $f : [1, \infty) \to [0, \infty)$ be a continuous function which is monotone decreasing. Then the series*

$$s = \sum_{j=1}^{\infty} f(j)$$

is convergent if and only if the improper integral

$$\int_1^{\infty} f(x)dx$$

is convergent.

Proof. The proof hinges on the observation that

$$f(j) \leq \int_{j-1}^{j} f(x)dx \leq f(j-1), \quad j = 2, 3, 4, \ldots . \tag{3.2}$$

Here the central integral exists because f is continuous. The two inequalities follow from the fact that f is continuous and monotone decreasing. Let s_n denote the partial sum $s_n = \sum_{j=1}^{n} f(j)$

and let t_n denote the integral $t_n = \int_1^n f(x)dx$. Then by application of (3.2) we have

$$s_n - f(1) = \sum_{j=2}^{n} f(j) \leq \sum_{j=2}^{n} \int_{j-1}^{j} f(x)dx = t_n \leq \sum_{j=2}^{n} f(j-1) = \sum_{j=1}^{n-1} f(j) = s_{n-1}.$$

We see that the sequence $\{s_n\}_{n=1}^{\infty}$ is convergent if and only if the sequence $\{t_n\}_{n=1}^{\infty}$ is convergent, i.e, if and only if $\int_1^{\infty} f(x)dx < \infty$. ■

Example 3.4 Show that the series $\sum_{n=1}^{\infty} \frac{1}{n^2}$ is convergent.

Solution Let $f : [1, \infty) \to (0, \infty)$ denote the function $f(x) = \frac{1}{x^2}$. Then $\sum_{n=1}^{\infty} \frac{1}{n^2} = \sum_{n=1}^{\infty} f(n)$. Since

$$\int_1^M \frac{1}{x^2}dx = \left[-\frac{1}{x}\right]_1^M = 1 - \frac{1}{M} \to 1, \quad M \to \infty$$

we conclude that the improper integral $\int_1^{\infty} f(x)dx$ is convergent. It follows that the series $\sum_{n=1}^{\infty} \frac{1}{n^2}$ is convergent. ■

3.4 Continuous functions

Definition 3.1 *Let $A \subseteq \mathbb{R}$ and let $f : A \to \mathbb{R}$. We say that f is continuous at $x_0 \in I$, if*

$$\forall \epsilon > 0 \, \exists \delta > 0 \, \forall x \in I \, : \, |x - x_0| < \delta \Rightarrow |f(x) - f(x_0)| < \epsilon$$

We say that f is continuous, if f is continuous at each point $x_0 \in A$.

Example 3.5 Let $f : \mathbb{R} \to \mathbb{R}$ denote the identity map, i.e., i.e. the function given by $f(x) = x$. Show that f is continuous.

Solution Let $x_0 \in \mathbb{R}$ be any point and let $\epsilon > 0$ be given. We have to find $\delta > 0$, such that $|x - x_0| < \delta$ implies that $|f(x) - f(x_0)| < \epsilon$. However, since $f(x) = x$, we can use $\delta = \epsilon$. ■

The following theorem shows how continuous functions can be constructed from functions which are known to be continuous using the absolute value, basic arithmetic operations and function compositions.

Theorem 3.6 *Let $A \subseteq \mathbb{R}$. Let $f : A \to \mathbb{R}$ be continuous. Then $|f|$ is continuous and $c \cdot f$ is continuous for each $c \in \mathbb{R}$. If $g : A \to \mathbb{R}$ is continuous, then each of the functions $f + g$, $f - g$, and $f \cdot g$ are continuous.*

Example 3.6 Show that every real polynomial $p : \mathbb{R} \to \mathbb{R}$ is continuous.

Solution We will prove this statement using the principle of mathematical induction. Let $V \subseteq \mathbb{N}$ be given by

$$V = \{n \in \mathbb{N} \mid \text{all polynomials of degree at most } n \text{ are continuous}\}. \tag{3.3}$$

We have to show that $V = \mathbb{N}$. We first show that $1 \in V$, i.e, every polynomial of degree at most 1 is continuous. Let p be a real polynomial of degree at most 1, i.e., $p(x) = c_0 + c_1 x$ for real constants c_0 and c_1. From Example 3.5 we know that the function $f(x) = x$ is continuous. Since c_1 is a constant, Theorem 3.6 implies $c_1 \cdot f$ is continuous. Since the constant function $g = c_0$ is continuous, Theorem 3.6 implies $p = f + g$ is continuous. This shows that $1 \in V$. We now show that $n \in V$ implies that $n + 1 \in V$. Assume $n \in V$ and let p denote any real polynomial of degree at most $n + 1$, i.e.,

$$p(x) = \sum_{j=0}^{n+1} a_j x^j, \quad a_j \in \mathbb{R}, \quad j = 0, 1, \ldots n + 1$$

We have to show that p is continuous. To that end we rewrite p as

$$p(x) = \sum_{j=0}^{n+1} a_j x^j = a_0 + \sum_{j=1}^{n+1} a_j x^j = a_0 + x \sum_{j=0}^{n} a_{j+1} x^j = a_0 + xq(x), \tag{3.4}$$

where $q(x) = \sum_{j=0}^{n} a_{j+1} x^j$ is a polynomial of degree at most n. By assumption ($n \in V$), q is continuous. As before, Theorem 3.6 now implies $p = a_0 + x \cdot q$ is continuous. This shows that $n+1 \in V$. By the principle of mathematical induction, we conclude that $V = \mathbb{N}$. This completes the proof. ■

Theorem 3.7 *Let $A \subseteq \mathbb{R}$. Let $f, g : A \to \mathbb{R}$. Let $B = \{x \in A \mid g(x) \neq 0\}$ and let $h : B \to \mathbb{R}$ be given by $h(x) = f(x)/g(x)$. Then h is continuous.*

Theorem 3.8 *Let $A \subseteq \mathbb{R}$ and $B \subseteq \mathbb{R}$. Let $f : A \to B$ and $g : B \to \mathbb{R}$ be continuous functions. Then $h : A \to \mathbb{R}$ given by $h(x) = g(f(x))$ is continuous.*

A large number of complicated calculations hinge on our ability to solve nonlinear equations of the form

$$f(x) = 0.$$

The following theorem is frequently used to find an interval which contains at least one solution.

Theorem 3.9 **Intermediate value theorem** *Let $f : [a, b] \to \mathbb{R}$ be a continuous function. Let y denote any number between $f(a)$ and $f(b)$. Then there exists at least one $x \in (a, b)$ such that $y = f(x)$. In particular, if $f(a)$ and $f(b)$ have different sign, then f has a zero in (a, b).*

Example 3.7 Let $f : \mathbb{R} \to \mathbb{R}$ be given by $f(x) = x^3 - 4x^2 \cos(2\pi x) - x + 1$. Show that f has at least 3 zeros in the interval $[0, 2]$.

Solution We see that the function f is continuous by repeated application of Theorem 3.6. Hence, the intermediate value theorem applies to f and the problem is now a question for finding intervals where f changes sign. We deliberately select a set of points for which the computation of $\cos(2\pi x)$ is trivial and can be done exactly. We have the following table.

x	0	$\frac{1}{2}$	1	$\frac{3}{2}$	2
$f(x)$	1	$\frac{13}{8}$	-3	$\frac{95}{8}$	-9

We see that f changes sign three times. In particular, f has at least one root in each of the intervals $(\frac{1}{2}, 1)$, $(1, \frac{3}{2})$ and $(\frac{3}{2}, 2)$. Since the intervals are disjoint, the roots are necessarily distinct. This shows that f has at least three roots in the interval $[0, 2]$. ■

3.5 Differentiable functions

This is the basic definition of differentiability.

Definition 3.2 *Let $f : (a, b) \to \mathbb{R}$ and let $x_0 \in (a, b)$. We say that f is differentiable at x_0 if there exists $c \in \mathbb{R}$ such that*

$$\frac{f(x) - f(x_0)}{x - x_0} \to c, \quad x \to x_0, \quad x \neq x_0$$

We say that f is differentiable if f is differentiable for all $x_0 \in (a, b)$.

The sum, difference and product of a pair of differentiable functions is differentiable.

Theorem 3.10 *Let I be an interval and let $f : I \to \mathbb{R}$ be differentiable. Then $c \cdot f$ is differentiable for each constant $c \in \mathbb{R}$ and $(c \cdot f)' = c \cdot f'$. If $g : I \to \mathbb{R}$ is differentiable, then the functions $f + g$, $f - g$ and $f \cdot g$ are differentiable. Moreover, the derivatives are given by $f' + g'$, $f' - g'$ and $f'g + fg'$.*

Theorem 3.11 *Let I be an interval and let $f, g : I \to \mathbb{R}$. Let $J = \{x \in I \mid g(x) \neq 0\}$ and let $h : J \to \mathbb{R}$ be given by $h(x) = f(x)/g(x)$. Then h is differentiable and $h' = (f'g - fg')/g^2$.*

Theorem 3.12 Chain-rule of differentiable functions *Let I and J be intervals and let $f : I \to J$ and let $g : J \to \mathbb{R}$ be differentiable functions. Then $h : I \to \mathbb{R}$ given by $h(x) = g(f(x))$ is differentiable. Moreover, derivative h' is given by $h'(x) = g'(f(x))f'(x)$.*

The following theorem can be used to determine if the nonlinear equation

$$f'(x) = 0$$

has a solution.

Theorem 3.13 Rolle's theorem *Let $f : [a, b] \to \mathbb{R}$ be continuous for $x \in [a, b]$ and differentiable for $x \in (a, b)$. If $f(a) = f(b) = 0$, then there exists at least one $\xi \in (a, b)$ such that*

$$f'(\xi) = 0.$$

Example 3.8 Let $f : [0, 1] \to \mathbb{R}$ be given by

$$f(x) = x(x - 1)e^x.$$

Show that there exists at least one $\xi \in (0, 1)$ such that $f'(\xi) = 0$.

Solution The function f is differentiable for all $x \in [0, 1]$ because it is the product of three differentiable functions. Hence it is also continuous for all $x \in [0, 1]$. Since $f(0) = f(1) = 0$, Rolle's theorem implies that there exists a $\xi \in (0, 1)$, such that $f'(\xi) = 0$. ■

The following theorem is a very powerful extention of Rolle's theorem. In reality, the mean value theorem of differentiable function is the statement that your average speed must equal your instantaneous speed at least once during your journey.

Theorem 3.14 The mean value theorem of differentiable functions *Let $f : [a, b] \to \mathbb{R}$ be continuous for $x \in [a, b]$ and differentiable for $x \in (a, b)$. Then there exists at least one $\xi \in (a, b)$ such that*

$$\frac{f(b) - f(a)}{a - b} = f'(\xi).$$

Example 3.9 Let $f : \mathbb{R} \to \mathbb{R}$ be given by

$$f(x) = x^2 + 5x \sin(\pi x).$$

Show that there exists at least one ξ between 0 and 1, such that

$$\frac{f(1) - f(0)}{1 - 0} = f'(\xi).$$

Solution The product rule and the sum rule of differention imply that f is differentiable for all $x \in [0, 1]$. It follows, that f is continuous for all $x \in [0, 1]$. By the mean value theorem, there exists $\xi \in (0, 1)$ such that

$$\frac{f(1) - f(0)}{1 - 0} = -4 = f'(\xi).$$

■

The following notation allows us to rapidly specify how many times a given function can be differentiated.

Definition 3.3 **The classes C^m and C^∞** *Let $I, J \subset \mathbb{R}$ be intervals. Let $f : I \to J$. We say that $f \in C^0(I, J)$ if and only if f is continuous for all $x \in I$. We say that $f \in C^{m+1}(I, J)$ if f is differentiable and $f' \in C^m(I, J)$. If $f \in C^m(I, J)$ for all m, then we say that $f \in C^\infty(I, J)$. When the intervals I and J are obvious from the context, we will write $f \in C^0$, $f \in C^m$, or $f \in C^\infty$ as appropriate.*

Definition 3.4 **Taylor-polynomials** *Let $I, J \subset \mathbb{R}$ be intervals and let $f \in \mathbb{C}^m(I, J)$. Let $x_0 \in I$. Then the mth order Taylor polynomial p_m for f about x_0 is given by*

$$p_m(x) = \sum_{j=0}^{m} \frac{f^{(j)}(x_0)}{m!}(x - x_0)^j. \tag{3.5}$$

Example 3.10 Let $f : \mathbb{R} \to \mathbb{R}$ be given by $f(x) = e^x \cos(\pi x)$. Let $x_0 = 1$. Find the 2nd order Taylor polynomial for f about $x_0 = 1$.

Solution Repeated application of Theorem 3.10 shows that $f \in C^2(\mathbb{R}, \mathbb{R})$. Moreover, we have

$$\begin{aligned} f'(x) &= e^x \cos(\pi x) - \pi e^x \sin(\pi x), \\ f''(x) &= e^x \cos(\pi x) - 2\pi e^x \sin(\pi x) - \pi^2 e^x \cos(\pi x), \end{aligned}$$

and

$$f(1) = -e, \quad f'(1) = -e, \quad f''(1) = e(\pi^2 - 1).$$

Therefore

$$p_2(x) = -e - e(x-1) + e\frac{\pi^2 - 1}{2}(x-1)^2 = -e\left(x + \frac{1-\pi^2}{2}(x-1)^2\right)$$

■

The following theorem can be used to construct approximations of complicated functions. It is also the theoretical foundation for the vast majority of the results presented in the rest of these notes.

Theorem 3.15 **Taylor's formula with remainder term** *Let $f \in \mathbb{C}^{m+1}([a, b], \mathbb{R})$ and let $x_0 \in [a, b]$. Let p_m denote the Taylor polynomial of f at x_0 of degree m. Let $x \in [a, b]$. Then there exists at least one ξ between x_0 and x such that*

$$f(x) - p_m(x) = \frac{f^{(m+1)}(\xi)}{(m+1)!}(x - x_0)^{m+1}.$$

Proof. Let $x \in [a, b]$ be any point. If $x = x_0$ there is nothing to show, so assume $x \neq x_0$. Define the auxilliary function

$$g(t) = f(t) - p_m(t) - \frac{f(x) - p_m(x)}{(x - x_0)^{m+1}}(t - x_0)^{m+1}.$$

We will know derive information about g. Our goal is to show that $g^{(m+1)}$ has at least one zero ξ between x and x_0. It is clear that $g \in C^{(m+1)}$ and

$$g^{(m+1)}(t) = f^{(m+1)}(t) - \frac{f(x) - p_m(x)}{(x - x_0)^{m+1}}(m+1)!$$

because $f \in C^{(m+1)}$ and p_m is a polynomial of degree at most m. Moreover, if $g^{(m+1)}(\xi) = 0$, then

$$f(x) - p_m(x) = \frac{f^{(m+1)}(\xi)}{(m+1)!}(x - x_0)^{m+1}.$$

By design, we have

$$g(x) = g(x_0) = 0.$$

By Rolle's theorem, there is at least one ξ_1 between x_0 and x, such that $g'(\xi_1) = 0$. However, by design, we also have $g'(x_0) = 0$, i.e.,

$$g'(\xi_1) = g'(x_0) = 0.$$

This indicates that we can proceed in an inductive manner. To that end, let V denote the set given by

$$V = \{j \in \mathbb{N} \,|\, \exists\, \xi_j \text{ between } x \text{ and } x_0 \text{ such that } g'(x_j) = 0\}.$$

We have shown that $1 \in V$. We claim $V = \{1, 2, \ldots, m+1\}$. Let $j \leq m$ and $j \in V$. We will show $j + 1 \in V$. Since $j \in V$ we have ξ_j between x and x_0 such that $g^{(j)}(\xi_j) = 0$. By design, $g^{(j)}(x_0) = 0$. By Rolle's theorem, there exists ξ_{j+1} between ξ_j and x_0 such that $g^{(j+1)}(\xi_{j+1}) = 0$. This shows that $j + 1 \in V$. By the principle of mathematical induction, we conclude that $V = \{1, 2, \ldots, m+1\}$. In particular, we have $m+1 \in V$ and there exists $\xi = \xi_{m+1}$ such that $g^{(m+1)}(\xi) = 0$. This completes the proof. ■

Example 3.11 Let $f : \mathbb{R} \to \mathbb{R}$ be given by

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

Let p_m denote the mth order Taylor polynomial for f about $x_0 = 0$. Let $K > 0$. Show that the relative error $r : \mathbb{R} \to \mathbb{R}$ given by

$$r(x) = \frac{f(x) - p_m(x)}{f(x)}$$

satisfies

$$|r(x)| < \frac{K^{m+1}}{(m+1)!}$$

for all $x \in [0, K]$.

Solution Let $x \in [0, K]$. Then by Taylor's theorem, there exists ξ in $(0, x)$ such that

$$f(x) - p_m(x) = \frac{f^{(m+1)}(\xi)}{(m+1)!}(x - x_0)^{m+1} = \frac{\exp(\xi)}{(m+1)!}x^{m+1}$$

It follows, that

$$\left|\frac{f(x) - p_m(x)}{f(x)}\right| \leq \frac{\exp(\xi)}{\exp(x)} \frac{|x|^{m+1}}{(m+1)!} < \frac{K^{m+1}}{(m+1)!}.$$

The last inequality is strict, because $\exp(\xi) < \exp(x)$, since $\xi \in (0, x)$. ■

Exercises

1. Let $\alpha \in [0, 1)$ and $\beta \geq 0$. Let $\{x_n\}_{n=0}^{\infty}$ be a sequence satisfying $|x_{n+1}| \leq \alpha |x_n| + \beta$. Show that the sequence is convergent with limit x satisfying $|x| \leq \frac{\beta}{1-\alpha}$.

2. Let $f : [1, \infty) \to [0, \infty)$ be a continuous and monotone decreasing function such that the improper integral $\int_1^\infty f(x)dx$ exists. Show that

$$\sum_{j=n+1}^{\infty} f(j) \leq \int_n^\infty f(x)dx$$

3. Show the infinite series $s = \sum_{j=1}^{\infty} \frac{1}{j^2}$ is convergent and $\sum_{j=n+1}^{\infty} \frac{1}{j^2} \leq \frac{1}{n}$. Let s_n denote the sum of the first n terms. Show that the relative error $r_n = (s - s_n)/s$ satisfies the bound $|r_n| \leq \frac{1}{n}$ for all n.

4. Show the infinite series $s = \sum_{j=1}^{\infty} \frac{1}{j^3}$ is convergent and $\sum_{j=n+1}^{\infty} \frac{1}{j^3} \leq \frac{1}{2n^2}$. Let s_n denote the sum of the first n terms. Show that the relative error $r_n = (s - s_n)/s$ satisfies the bound $|r_n| \leq \frac{1}{2n^2}$ for all n.

5. Let $f : \mathbb{R} \to \mathbb{R}$ be the function given $f(x) = \exp(x)$. Let p_m denote the mth order Taylor polynomial for f about $x_0 = 0$. Let $K > 0$. Show that the relative error $r = (f - p_m)/f$ satisfies $|r(x)| \leq e^K K^{m+1}/(m+1)!$ for all $x \in [-K, 0]$. Compare with Example 14.

Chapter 4

Analysis of rounding errors

Contents

4.1 Floating point arithmetic

Any real number $x \neq 0$ can be written in the form

$$x = s(\beta_0.\beta_1\beta_2\beta_3,\dots)_b \times b^e = s\left(\sum_{j=0}^{\infty} \beta_j b^{-j}\right) b^e \tag{4.1}$$

where the sign $s \in \{-1, 1\}$, the base $b > 1$, the digits $\beta_j \in \{0, 1, 2, \dots, b-1\}$ and the exponent $e \in \mathbb{Z}$. The representation is (essentially) unique if $b_0 \neq 0$. A number written in this form is said be written in normalized scientific notation.

The most commonly used values of the base b are $b = 2$ (binary), $b = 10$ (decimal) and $b = 16$ (hexadecimal). Computers are necessarily limited to a finite number of digits and exponents. It is therefore practical to introduce that following set of numbers.

Definition 4.1 *A real number x is a member of the floating point number system $\mathcal{F} = \mathcal{F}(b,k,m,M)$ if and only if*

$$x = s\left(\sum_{j=0}^{k} \beta_j b^{-j}\right) b^e,$$

where $s \in \{-1,1\}$, $\beta_j \in \{0,1,2,\dots,b-1\}$, $\beta_0 > 0$ and $m \le e \le M$.

Each set of numbers $\mathcal{F} = \mathcal{F}(b,k,m,M)$ contains only a finite number of elements, see Exercise 1. In particular, $\mathcal{F}$ has a smallest positive number and a largest positive number.

Lemma 4.1 *Let $x_{\min}$ ($x_{\max}$) denote the smallest (largest) positive number in $\mathcal{F}(b,k,m,M)$. Then*

$$x_{\min} = b^m, \quad x_{\max} = \frac{b - b^{-k}}{b-1} b^M.$$

Proof. Left to the reader as Exercise 2. ■

Definition 4.2 *The representable range of $\mathcal{F}(b,k,m,M)$ is given by the union*

$$[-x_{\max}, -x_{\min}] \cup [x_{\min}, x_{\max}].$$

Given a real number x in the representable range of $\mathcal{F}(b,k,m,M)$ we now seek the best possible approximation $\hat{x}$ within $\mathcal{F}(b,k,m,M)$.

Theorem 4.1 *Let $x \in \mathbb{R}$ be in the representable range of $\mathcal{F} = \mathcal{F}(b,k,m,M)$. Then there exists a number $\hat{x} \in \mathcal{F}$ such that*

$$\left|\frac{x-\hat{x}}{x}\right| \le \frac{1}{2} b^{-k}. \tag{4.2}$$

Proof. Without loss of generality we can assume that $s = 1$. If $x \in \mathcal{F}$ there is nothing to show, so we can assume that $x < x_{\max}$. By assumption, x can be written in the form

$$x = (\beta_0.\beta_1\beta_2\beta_3,\dots)_b \times b^e, \quad m \le e \le M$$

The number x_- given by

$$x_- = (\beta_0.\beta_1\beta_2\beta_3,\dots,\beta_k)_b \times b^e \in \mathcal{F}$$

is the largest number in $\mathcal{F}$ which is smaller than x. The next number in $\mathcal{F}$ is given by

$$x_+ = \left[(\beta_0.\beta_1\beta_2\beta_3,\dots,\beta_k)_b + b^{-k}\right] \times b^e = x_- + b^{e-k}.$$

By construction,

$$x_- \le x \le x_+, \quad x_+ - x_- = b^{e-k}.$$

Let $\hat{x} \in \mathcal{F}$ denote the number which is closest to x, i.e., either $\hat{x} = x_-$ or $\hat{x} = x_+$. Then the error $e = x - \hat{x}$ satisfies

$$|e| \le \frac{1}{2} b^{e-k}$$

Since $x \ge b^e$ we can bound the relative error $r = (x-\hat{x})/x$ as follows

$$|r| \le \frac{1}{2} b^{-k}.$$

■

Remark 4.1 The number $\hat{x}$ produced by the Theorem 4.1 is called the floating point representation of x with respect to $\mathcal{F}$ and is written $\hat{x} = \mathrm{fl}(x)$. The number $u = \frac{1}{2}b^{-k}$ is called the unit round off error.

Corollary 4.1 *Let $x \in \mathbb{R}$ denote a number in the representable range of $\mathcal{F}$ and let u denote the unit round off error. Then the floating point representation $fl(x)$ of x can be written as*

$$fl(x) = x(1+\epsilon), \quad \textit{where } |\delta| \leq u. \tag{4.3}$$

Moreover, $fl(x)$ can also be written as

$$(1+\delta) fl(x) = x, \quad \textit{where } |\delta| \leq u. \tag{4.4}$$

Proof. We will only prove equation (4.3) and leave equation (4.4) to the reader as Exercise 7. In both cases it is vital to understand the proof of Theorem 4.1. Let $\hat{x} = \mathrm{fl}(x)$ denote the floating point representation of x. We now rewrite $\hat{x}$ in an attempt to identify the appropriate value of δ. We have

$$\hat{x} = x + (\hat{x} - x) = x + \left(\frac{\hat{x} - x}{x}\right) x = x\left(1 + \frac{\hat{x} - x}{x}\right) = x(1+\delta),$$

where $\delta = -\frac{x-\hat{x}}{x}$ satisfies $|\delta| \leq u$, by Theorem. This completes the proof of equation (4.3). ■

Corollary 4.1 is the mathematical statement that a number x in the representable range can be approximated by $\hat{x} \in \mathcal{F}$ with a relative error which is bounded by the unit round off error u.

Below follows the standard model for floating point computations.

Model of floating point computations

Let $\circ$ denote one of the basic arithmetic operations, i.e., addition, subtraction, multiplication or division. Let $x, y \in \mathcal{F} = \mathcal{F}(b, k, m, M)$. Let $r = x \circ y$ denote the exact result. If r is in the representable range, then the computed value $\hat{r}$ satisfies

$$\hat{r} = \mathrm{fl}(r) \tag{4.5}$$

Most computers conforms to the `IEEE` standard for floating point arithmetic (IEEE 754) and offers the user the choice between binary single precision numbers and binary double precision number. Binary single precision contains the set $\mathcal{F}(2, 23, -126, 127)$. Binary double precision contains $\mathcal{F}(2, 52, -1022, 1023)$. The IEEE standard includes 0, -0, ∞, $-\infty$ as well as NaN (not a number). The IEEE standard also includes tiny subnormal numbers which allows for gradual underflow to zero at the expense of precision.

Exercises

1. Let $\mathcal{F} = \mathcal{F}(b, k, m, M)$ denote a system of floating point numbers. Show that $\mathcal{F}$ has precisely
$$\#\mathcal{F} = 2(M - m + 1)(b-1)b^k$$
elements.

2. Find the value of the smallest and the largest element of $\mathcal{F} = \mathcal{F}(b, k, m, M)$, i.e., prove Lemma 4.1.

3. What is the smallest (largest) normalized (binary) IEEE single precision number?

4. What is the smallest (largest) normalized (binary) IEEE double precision number?

5. Find the smallest positive number $a \in F = \mathcal{F}(2, 23, -126, 127)$ such that
$$[a, a + 2\pi] \cap \mathcal{F} = \{a\}.$$

6. Find the smallest positive number $a \in F = \mathcal{F}(2, 52, -1022, 1023)$ such that
$$[a, a + 2\pi] \cap \mathcal{F} = \{a\}.$$

7. Prove that the floating point representation of a real number x in the representable range satisfies equation (4.4).

8. Is it possible to compute $\cos(x)$ for all x in the representable range?

9. Complete the proof of Corollary 4.1. Hint: the key is to find the only value of ϵ such that $x = \hat{x}(1 + \epsilon)$.

4.2 A priori error analysis

Consider the problem of computing, say, the sum s of n floating point numbers $\{x_i\}_{i=1}^n$, i.e. the number
$$s = \sum_{i=1}^{n} x_j. \tag{4.6}$$
In all likelihood, the computed sum $\hat{s}$ will differ from the exact sum s. The difference is caused by the rounding errors committed during the individual additions. An a priori error bound is an error bound of the form
$$|s - \hat{s}| \leq c_n(u) \tag{4.7}$$
where u is the unit round off error and c_n is a function which is *independent* of x_j, but satisfies
$$c_n(u) \rightarrow 0, \quad u \rightarrow 0_+ \tag{4.8}$$
A priori error bounds are often a bit pessimistic. The purpose of a priori analysis is to determine if an algorithm is reliable or not. Running error bounds can be much more accurate. They depend on the actual input and the intermediate results, but require extra arithmetic operations to compute. Running error analysis will be discussed in Section 4.3.

In this section we demonstrate how to bound the error associated with several standard computations: sums, inner products, polynomials and triangular linear systems. Experience has show that it is practical to use rather specialized notation to simplify the analysis.

Definition 4.3 *Let n be a positive integer such that $nu < 1$. The number γ_n is defined by*
$$\gamma_n = \frac{nu}{1 - nu}. \tag{4.9}$$

The key properties of these numbers are contained in the following lemma.

Lemma 4.2 *Let m and n be positive integers, such that $(m + n)u < 1$. Then*
$$\gamma_m + \gamma_n + \gamma_m\gamma_n \leq \gamma_{m+n}. \tag{4.10}$$

Proof. The proof is left to the reader as Problem 3. ■

The following notation is extremely useful, but it is necessary to be very careful

Definition 4.4 *We write $x = \langle n \rangle$ if and only if $x = 1 + \Theta$ where Θ satisfies $|\Theta| \leq \gamma_n$.*

Corollary 4.2 *Let m and n be nonnegative integers, such that $(m+n)u < 1$. Let x and y be real number such that $x = \langle m \rangle$ and $y = \langle n \rangle$, then $xy = \langle m+n \rangle$.*

Proof. We have to show that $xy = 1 + \Theta$ with $|\Theta| \leq \gamma_{m+n}$. By assumption, we have

$$x = 1 + \Theta_x, \quad y = 1 + \Theta_y,$$

where $|\Theta_x| \leq \gamma_m$ and Θ_y. It follows,

$$xy = (1+\Theta_x)(1+\Theta_y) = 1 + \Theta_x + \Theta_y + \Theta_x\Theta_y.$$

It is clear that the choice of

$$\Theta = \Theta_x + \Theta_y + \Theta_x\Theta_y.$$

allows us to write $xy = 1 + \Theta$. Moreover, by Lemma 4.2 we have

$$|\Theta| \leq |\Theta_x| + |\Theta_y| + |\Theta_x||\Theta_y| \leq \gamma_m + \gamma_n + \gamma_m\gamma_n \leq \gamma_{m+n}$$

This completes the proof. ■

The fundamental model of floating point computation can be stated in terms of the bracket-notation.

Model of floating point computations

Let $\circ$ denote one of the basic arithmetic operations, i.e., addition, subtraction, multiplication or division. Let $x, y \in \mathcal{F} = \mathcal{F}(b,k,m,M)$. Let $r = x \circ y$ denote the exact result. If r is in the representable range, then the computed value $\hat{r}$ can be written as

$$\hat{r} = r\langle 1 \rangle. \tag{4.11}$$

Moreover, r can also be written as

$$\langle 1 \rangle \hat{r} = r. \tag{4.12}$$

Exercises

1. In IEEE single (double) precision, what is the largest value of n for which γ_n is defined?

2. A hypothetical machine can complete 2^{30} floating point operations per second. How long will it take to add n numbers, where n is the smallest value such that γ_n is not defined in single (double) precision.

3. Prove the fundamental Lemma 4.2. Hint: Write the term $\gamma_m + \gamma_n + \gamma_n\gamma_m$ as a single fraction with a common denominator and simplify as much as possible.

4. Let a, b, and c be floating point numbers and compute the sum as follows

$$\begin{aligned} d &\leftarrow (b+c) \\ s &\leftarrow a+d \end{aligned}$$

   Use the fundamental model of floating point arithmetic to show that the computed sum $\hat{s}$ can be written as

$$\hat{s} = a\langle 1 \rangle + (b+c)\langle 2 \rangle.$$

   Moreover, show that the computed sum satisfies an equation of the form

$$\langle 1 \rangle \hat{s} = a + (b+c)\langle 2 \rangle.$$

5. Let a, b, c, and $d \neq 0$ be floating point numbers and compute the solution x of $dx = a - bc$ as follows,

$$\begin{aligned} r &\leftarrow bc \\ s &\leftarrow a - r \\ x &\leftarrow \frac{s}{d} \end{aligned}$$

Use the fundamental model of floating point arithmetic to show that the computed result $\hat{x}$ satisfies an equation of the form

$$d\langle 2\rangle \hat{x} = a - bc\langle 1\rangle.$$

4.2.1 Sums

Let n be a positive integer and let $\{x_i\}_{i=1}^n$ floating point numbers. Let $s = \sum_{i=1}^n x_i$ denote the sum. Let $\hat{s}$ denote the computed sum obtained using Algorithm 2.

Algorithm 2 Summation

```
Require: $\{x_i\}_{i=1}^n \in \mathbb{R}$
Ensure: $s = \sum_{i=1}^n x_i \in \mathbb{R}$
$s_1 \leftarrow x_1$
for $i = 2, \dots, n$ do
    $s_i \leftarrow s_{i-1} + x_i$
end for
$s = s_n$
return $s$
```

Our goal is to bound the error $s - \hat{s}$. To that end we require a formula for the computed sum $\hat{s}$. We begin with a small example.

Example 4.1 Find an expression for the value returned by Algorithm 2, when summing $n = 4$ values.

Solution Let $\hat{s}_j$ denote the computed value of s_j. By applying the fundamental model of floating point computation we find

$$\begin{aligned} \hat{s}_1 &= x_1, \\ \hat{s}_2 &= (\hat{s}_1 + x_2)\langle 1\rangle = (x_1 + x_2)\langle 1\rangle, \\ \hat{s}_3 &= (\hat{s}_2 + x_3)\langle 1\rangle = (x_1 + x_2)\langle 2\rangle + x_3\langle 1\rangle, \\ \hat{s}_4 &= (\hat{s}_3 + x_4)\langle 1\rangle = (x_1 + x_2)\langle 3\rangle + x_3\langle 2\rangle + x_4\langle 1\rangle. \end{aligned}$$

It follows that the computed sum $\hat{s}$ can be written as

$$\hat{s} = x_1\langle 3\rangle + x_2\langle 3\rangle + x_3\langle 2\rangle + x_4\langle 1\rangle.$$

■

It is not difficult to see how this pattern generalize to any value of n. We have the following theorem.

Theorem 4.2 *Assume $nu < 1$. If Algorithm 2 does not experience overflow/underflow, then the computed sun $\hat{s}$ satisfies*

$$\hat{s} = \sum_{j=1}^n x_j \langle k_j\rangle, \tag{4.13}$$

where the integers k_j are given by

$$k_j = \begin{cases} n-1 & j = 1,2 \\ n-j+1 & j > 2. \end{cases} \tag{4.14}$$

Proof. Left to the reader as Exercise 1. ■

An error bound flows directly from our representation of the computed sum. We begin with a small example.

Example 4.2 Derive an error bound when summing $n = 4$ floating point numbers.

Solution From Example 4.1 we know that the computed sum $\hat{s}$ can be written as

$$\hat{s} = x_1(1+\theta_1^{(3)}) + x_2(1+\theta_2^{(3)}) + x_3(1+\theta_3^{(2)}) + x_4(1+\theta_4^{(1)}),$$

where $|\theta_j^{(k)}| \leq \gamma_k$. The error can therefore be expressed as

$$s - \hat{s} = x_1\theta_1^{(3)} + x_2\theta_2^{(3)} + x_3\theta_3^{(2)} + x_4\theta_4^{(3)}.$$

It follows that

$$|s-\hat{s}| \leq |x_1|\gamma_3 + |x_2|\gamma_3 + |x_3|\gamma_2 + |x_4|\gamma_1 \leq \gamma_3(|x_1| + |x_2| + |x_3| + |x_4|).$$

or equivalently

$$|s-\hat{s}| \leq \gamma_3 \sum_{i=1}^{4} |x_i| \tag{4.15}$$

■

In general we have the following theorem.

Theorem 4.3 *Assume $nu < 1$. If Algorithm 2 does not experience overflow/underflow, then the computed sum satisfies the error bound*

$$|s-\hat{s}| \leq \gamma_{n-1} \sum_{j=1}^{n} |x_j|. \tag{4.16}$$

Proof. By Theorem 4.2 the computed sum written as

$$\hat{s} = \sum_{j=1}^{n} x_j(1+\theta_j^{(k_j)}), \quad |\theta_j^{(k_j)}| \leq \gamma_{k_j}.$$

The error can therefore be expressed as

$$s - \hat{s} = \sum_{j=1}^{n} x_j\theta_j^{(k_j)}).$$

This representation allows us to conclude

$$|s-\hat{s}| \leq \sum_{j=1}^{n} |x_j|\gamma_{k_j} \leq \gamma_{n-1} \sum_{j=1}^{n} |x_j|.$$

This completes the proof. ■

Once the error is bounded it is straightforward to bound the relative error. We have the following theorem.

Theorem 4.4 *Assume $nu < 1$ and $s \neq 0$. If Algorithm 1 does not overflow/underflow, then the computed sum satisfies the relative error bound*

$$\frac{|s - \hat{s}|}{|s|} \leq \gamma_{n-1} \frac{\sum_{j=1}^{n} |x_j|}{\left|\sum_{j=1}^{n} x_j\right|}. \tag{4.17}$$

If the numbers x_j have the *same* sign, then the relative error is small, because

$$\frac{|s - \hat{s}|}{|s|} \leq \gamma_{n-1}$$

However, if the terms have different signs, then s can small, while $\sum_{j=1}^{n} |x_j|$ is large. In this case, we can not be certain that the relative error is small. In practice, the relative error will be not be small.

Exercises

1. Prove Theorem 1.

2. Consider the problem of computing the sum of 4 floating point numbers using the formula

$$s = (x_1 + x_2) + (x_3 + x_4)$$

Show that the computed sum satisfies

$$\hat{s} = (x_1 + x_2)\langle 2\rangle + (x_3 + x_4)\langle 2\rangle$$

and the error $s - \hat{s}$ is bounded by

$$|s - \hat{s}| \leq \gamma_2(|x_1| + |x_2| + |x_3| + |x_4|)$$

Compare with Theorem 4.3.

3. Consider the problem of computing the sum of $8 = 2^3$ floating point numbers using the formula

$$s = ((x_1 + x_2) + (x_3 + x_4)) + ((x_5 + x_6) + (x_7 + x_8))$$

Show that the compute sum $\hat{s}$ satisfies

$$\hat{s} = (x_1 + x_2)\langle 3\rangle + (x_3 + x_4)\langle 3\rangle + (x_1 + x_2)\langle 3\rangle + (x_3 + x_4)\langle 3\rangle$$

and

$$|s - \hat{s}| \leq \gamma_3 \sum_{j=1}^{8} |x_j|$$

Compare with Theorem 4.3.

4. Generalize Problem 2 and Problem 3 to an arbitrary power of 2, i.e. the addition of $n = 2^k$ terms. Show that the computed sum can be written as

$$\hat{s} = \sum_{j=1}^{n/2} (x_{2j-1} + x_{2j})\langle k\rangle_j$$

and satisfies the error bound

$$|s - \hat{s}| \leq \gamma_k \sum_{j=1}^{n} |x_j|$$

4.2.2 Inner products

Let n be an integer and let $\mathbf{x} \in \mathbb{R}^{\mathbf{n}}$ and $\mathbf{y} \in \mathbb{R}^{\mathbf{n}}$ be vectors of floating point numbers. Let s denote the inner product between $\mathbf{x}$ and $\mathbf{y}$, i.e.,

$$s = \mathbf{x}^T \mathbf{y} = \sum_{i=1}^{n} x_i y_i.$$

The inner product is also known as the scalar product. Let $\hat{s}$ denote the computed value of s obtained using Algorithm 3.

Algorithm 3 Inner product

```
Ensure: $\mathbf{x}, \mathbf{y} \in \mathbb{R}^{\mathbf{n}}$
Require: $t = \mathbf{x}^{\mathbf{T}}\mathbf{y} \in \mathbb{R}$
$s_0 \leftarrow 0$
for $j = 1, \dots, n$ do
    $s_j \leftarrow s_{j-1} + x_j y_j$
end for
$s \leftarrow s_n$
return $s$
```

Our goal is to bound the error $s - \hat{s}$. To that end, we require a formula for the computed inner product $\hat{s}$.

Theorem 4.5 *Assume $nu < 1$. If Algorithm 3 does experience overflow/underflow, then the computed value of the inner product $s = \mathbf{x}^{\mathbf{T}}\mathbf{y}$ satisfies*

$$\hat{s} = \sum_{j=1}^{n} x_i y_i \langle m_j \rangle,$$

where the integers m_j are given by

$$m_j = \begin{cases} n & j = 1, 2, \\ n - j + 2 & j > 2. \end{cases}$$

Proof. Let $z_i = x_i y_i$ and let $\hat{z}_i$ denote the compute value of z_i. By the fundamental model of floating point arithmetic we have $\hat{z}_i = z_i\langle 1 \rangle$. By Theorem 4.2 we have

$$\hat{s} = \sum_{i=1}^{n} \hat{z}_i \langle k_j \rangle, \quad k_j = m_j - 1 \tag{4.18}$$

It follows, that

$$\hat{s} = \sum_{i=1}^{n} z_i \langle 1 \rangle \langle k_j \rangle = \sum_{i=1}^{n} z_i \langle m_j \rangle \tag{4.19}$$

This completes the proof. ■

An error bound flows directly from our expression for the computed value of the inner product. We have the following theorem.

Theorem 4.6 *Assume $nu < 1$. If Algorithm 3 does not experience overflow/underflow, then the computed inner product $\hat{s}$ satisfies*

$$|s - \hat{s}| \leq \gamma_n \sum_{i=1}^{n} |x_i||y_i|.$$

Proof. Left to the reader as Exercise 1. ■

We can now provide an upper bound for the relative error.

Theorem 4.7 *Assume $nu < 1$ and $s = \mathbf{x}^T\mathbf{y} \neq \mathbf{0}$. If Algorithm 3 does not experience overflow/underflow, then the computed inner product $\hat{s}$ satisfies the relative error bound*

$$\frac{|s-\hat{s}|}{|s|} \leq \gamma_n \frac{\sum_{i=1}^n |x_i||y_i|}{|\sum_{i=1}^n x_i y_i|}. \tag{4.20}$$

Proof. Left to the reader as Exercise 2. ■

If $\mathbf{x}$ and $\mathbf{y}$ are nearly orthogonal, then we can not be certain that Algorithm 3 computes the inner product with a small relative error. In practice, the relative error will be large, if $\mathbf{x}$ and $\mathbf{y}$ are nearly orthogonal.

Exercises

1. Prove the error bound given by Theorem 4.6.

2. Prove the relative error bound given by Theorem 4.7.

3. Let $A \in \mathbb{R}^{n\times n}$ be the tridiagonal matrix given by

$$A = n^2 \begin{bmatrix} 2 & -1 & & \\ -1 & \ddots & \ddots & \\ & \ddots & \ddots & -1 \\ & & -1 & 2 \end{bmatrix} \tag{4.21}$$

   (a) Show that the eigenvectors of A are given by the columns of the orthonormal matrix $V = [v_{ij}]$, where

$$v_{ij} = \sqrt{\frac{2}{n+1}} \sin\left(\frac{ij\pi}{n+1}\right) \tag{4.22}$$

4. Let $V \in \mathbb{R}^{n\times n}$ be the orthogonal matrix given by equation (4.22). Verify Theorem 4.6 by computing V in double precision and computing $V^TV - I$ in single precision

4.2.3 Polynomials

A real polynomial p of degree n can be written as

$$p(x) = \sum_{j=0}^n a_j x^j. \tag{4.23}$$

Polynomials can be evaluated using Horner's method, Algorithm 4.

Example 4.3 Show that Horner's method returns the correct result in exact arithmetic for all polynomials of degree $n = 2$.

Solution Let p denote a polynomial of degree 2, i.e.,

$$p(x) = a_0 + a_1 x + a_2 x^2.$$

Algorithm 4 Horner's method

```
A real polynomial $p$ of degree $n$ and $t \in \mathbb{R}$
**Ensure:** The value $y = p(t)$.
$p_0 \leftarrow a_n$
**for** $j = 1, \dots, n$ **do**
    $p_j \leftarrow p_{j-1}x + a_{n-j}$
**end for**
$y = p_n$
**return** $y$
```

By stepping through the algorithm we see that Horner's method computes

$$\begin{aligned} p_0 &= a_2, \\ p_1 &= p_0 x + a_1 = a_2 x + a_1, \\ p_2 &= p_1 x + a_0 = (a_2 x + a_1)x + a_0 = a_2 x^2 + a_1 x + a_0, \end{aligned}$$

and returns $y = p_2 = p(x)$ as expected. ■

It is no surprise that we have the following result.

Theorem 4.8 *In exact arithmetic, Algorithm 4 returns $y = p(x)$*

Proof. Left to the reader as Exercise 1. ■

In practice, the computed result will be affected by rounding errors. In order to bound the error we must first derive a representation of the computed values.

We start with two small example, i.e., $n = 2$ and $n = 3$.

Example 4.4 Find a representation of the computed value returned by Horner's method when $n = 2$.

Solution Let $\hat{p}_j$ denote the compute value of p_j. By applying the fundamental model of floating point arithmetic, we find

$$\begin{aligned} \hat{p}_0 &= a_2, \\ \hat{p}_1 &= \hat{p}_0 x\langle 2\rangle + a_1\langle 1\rangle = a_2 x\langle 2\rangle + a_1\langle 1\rangle, \\ \hat{p}_2 &= \hat{p}_1 x\langle 2\rangle + a_0\langle 1\rangle = a_2 x^2\langle 4\rangle + a_1 x\langle 3\rangle + a_0\langle 1\rangle. \end{aligned}$$

It follows that the computed value $\hat{y}$ can be written as

$$\hat{y} = a_2 x^2\langle 4\rangle + a_1 x\langle 3\rangle + a_0\langle 1\rangle.$$

■

Example 4.5 Analyse the error of Horner's method for $n = 3$.

Solution We proceed as in the previous example. We find that

$$\begin{aligned} \hat{p}_0 &= a_3 \\ \hat{p}_1 &= \hat{p}_0 x\langle 2\rangle + a_2\langle 1\rangle = a_3 x\langle 2\rangle + a_2\langle 1\rangle \\ \hat{p}_2 &= \hat{p}_1 x\langle 2\rangle + a_1\langle 1\rangle = a_3 x^2\langle 4\rangle + a_2 x\langle 3\rangle + a_1\langle 1\rangle \\ \hat{p}_3 &= \hat{p}_2 x\langle 2\rangle + a_0\langle 1\rangle = a_3 x^3\langle 6\rangle + a_2 x^2\langle 5\rangle + a_1\langle 3\rangle + a_0\langle 1\rangle. \end{aligned}$$

It follows that the computed value can be written as

$$\hat{y} = a_3 x^3 \langle 6 \rangle + a_2 x^2 \langle 5 \rangle + a_1 x \langle 3 \rangle + a_0 \langle 1 \rangle.$$

■

These two examples reveals the general pattern. The pattern can be maintained as long as the model of floating point arithmetic holds true and as long as the assumptions needed for Corollary 4.2 are satisfied. We have the following theorem.

Theorem 4.9 *Assume $2nu < 1$. If Algorithm 4 does not experience overflow/underflow, then the computed value $\hat{y}$ satisfies*

$$\hat{y} = a_n \langle 2n \rangle x^n + \sum_{j=0}^{n-1} a_j \langle 2j+1 \rangle x^j \tag{4.24}$$

This representation of the computed value allows us to bound the error and the relative error. It is useful to introduce the polynomial $\tilde{p}$ given by

$$\tilde{p}(t) = \sum_{j=0}^{n} |a_j| t^j. \tag{4.25}$$

Theorem 4.10 *Assume $2nu < 1$. If Algorithm 4 does not experience overflow/underflow, then the error satifies*

$$|y - \hat{y}| \leq \gamma_{2n} \tilde{p}(|x|) \tag{4.26}$$

Proof. By Theorem 4.10 we know that the computed value returned by Horner's method can be written as

$$\hat{y} = \sum_{j=0}^{n} a_j x^j (1 + \theta_j^{(k_j)}), \tag{4.27}$$

where $|\theta_j^{(k)}| \leq \gamma_k$ and k_j satisfies $k_j \leq 2n$. It is clear that

$$\hat{y} = y + \sum_{j=0}^{n} a_j x^j \theta_j^{(k_j)}). \tag{4.28}$$

The triangle inequality now implies

$$|y - \hat{y}| \leq \sum_{j=0}^{n} |a_j||x|^j \left|\theta_j^{(k_j)}\right| \leq \gamma_{2n} \sum_{j=0}^{n} |a_j||x|^j = \gamma_{2n} \tilde{p}(|x|) \tag{4.29}$$

This completes the proof. ■

Theorem 4.11 *Assume $2nu < 1$. If Algorithm 4 does not experience overflow/underflow, then the relative error satisfies*

$$\frac{|y - \hat{y}|}{|y|} \leq \gamma_{2n} \frac{\tilde{p}(|x|)}{|p(x)|} \tag{4.30}$$

It is important to notice that the relative error is not necessarily small. If all a_j and x are positive, then $\tilde{p}(|x|) = p(x)$ and the relative error is bounded by γ_{2n}. In general, $|p(x)| \leq \tilde{p}(|x|)$ and if x is close to a root of p, then $|p(x)| \ll \tilde{p}(|x|)$. This means that it is difficult to compute $p(x)$ accurately, when x is close to a root.

Exercises

1. Prove that Horner's method returns the correct value in exact arithmetic.

2. Implement and test Horner's method in MATLAB a function `XHorner`. The call sequence should be `[y]=XHorner(a,x)` where

   - `a` is an array containing the coefficients of the polynomial
   - `x` is an array of real numbers
   - `y` is an array containing $y = p(x)$

   (a) Your source code must contain a short description, the complete call sequence, a description of the input and output arguments as well as a minimal working example.

   (b) Write a minimal working example `XHornerMWE1` which computes the polynomial $p : [-1, 1] \to \mathbb{R}$ given by
   $$p(x) = 4x^3 - 3x$$
   using this definition as well as Horner's method.

   (c) The minimal working example should verify that your implementation is (probably) correct by producing a plot similar to Figure 4.1.

3. Extend your implementation of Horner's method `XHorner` to also compute the polynomial $x \to \tilde{p}(x)$ needed for the error bound given by Theorem 4.10.

   (a) The new call sequence should be `[y, pt]=XHorner(a,x)` where `y, a, x` are defined in Problem 2.

   (b) Write a minimal working example `XHornerMWE2` which computes $p : [0, 2] \to \mathbb{R}$ given by
   $$p_1(x) = (x - 1)^3 \tag{4.31}$$
   in double precision and applies Horner's method in single precision to the equivalent polynomial
   $$p_2(x) = x^3 - 3x^2 + 3x - 1 \tag{4.32}$$

   (c) Adjust `XHornerMWE2` until it produces a figure similar to Fig. 4.2.

4. Let $p(x) = \sum_{j=0}^{n} a_j x^j$ denote a polynomial. Let $p_j(x)$ denote the polynomials used internally by Horner's method when evaluating p.

   (a) Show that that the derivatives $q_j = p'_j(x)$ satisfy the recurrence relation
   $$q_j(x) = p_j(x) + xq_{j-1}(x). \tag{4.33}$$
   together with the initial condition $q_0 = 0$.

   (b) Extend the function `XHorner` to also compute the derivative of p.

4.2.4 Forward substitution

Let $\mathbf{L} = [l_{ij}] \in \mathbb{R}^{n\times n}$ be lower triangular matrix and consider the linear system

$$\mathbf{Lx} = \mathbf{f}.$$

Such triangular linear systems occur naturally when computing the Newton form of the interpolating polynomial of a real function. If L is nonsingular, then the system has a unique solution which can be computed using Algorithm 5

In order to analyse this algorithm we require the following lemma.

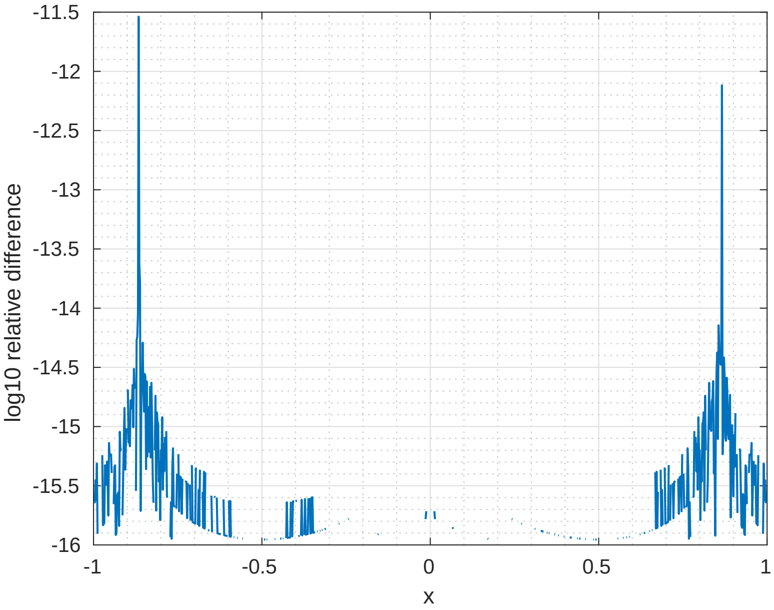

Figure 4.1: The relative difference between Horner's method and the direct computation of $p(x) = 4x^3 - 3x$, using $n = 1001$ equidistant points in the interval $[-1, 1]$.

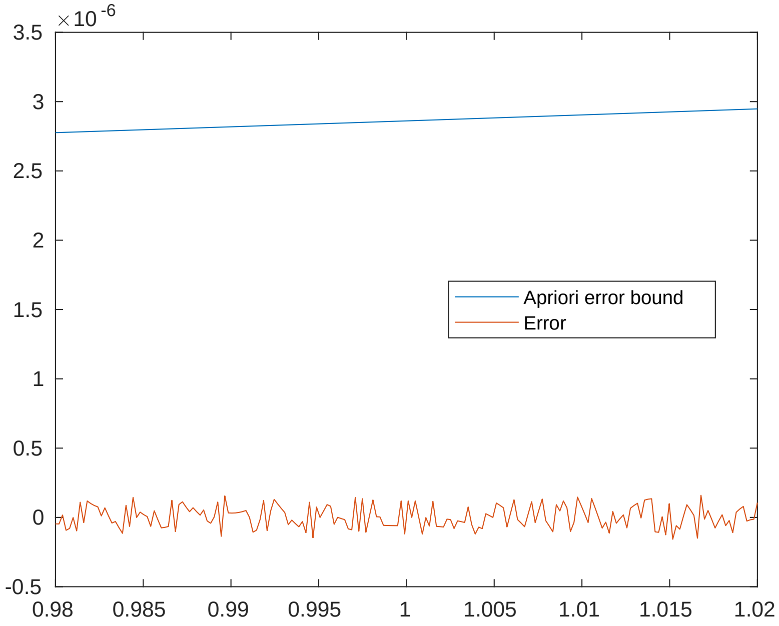

Figure 4.2: The error and the a priori error bound for Horner's method applied to the polynomial $p(x) = (x-1)^3$. Plot is generated using 200 equidistant points in the interval $[0.98, 1.02]$.

Algorithm 5 Forward substitution

```
Require: A nonsingular lower triangular linear system $\mathbf{Lx} = \mathbf{f}$
Ensure: The solution $\mathbf{x}$.
  $\mathbf{x} \leftarrow \mathbf{f}$
  for $i = 1, \dots, n$ do
    for $j = 1 : i - 1$ do
      $x_i = x_i - l_{ij} x_j$
    end for
    $x_i \leftarrow \frac{x_i}{l_{ii}}$
  end for
  return $\mathbf{x}$
```

Lemma 4.3 *Let $\mathbf{a}, \mathbf{b} \in \mathbb{R}^{\mathbf{k}}$, let $f \in \mathbb{R}$ and let $d \neq 0$. Assume that*

$$x = \frac{f - \mathbf{a}^{\mathbf{T}}\mathbf{b}}{d}$$

is computed using Algorithm 6. Then the computed value of $\hat{x}$ satisfies an equation of the form

$$d\langle k \rangle x = f - \sum_{j=1}^{k} \langle j \rangle a_j b_j$$

Algorithm 6 General update and division

```
$s \leftarrow f$
for $j = 1, 2, \dots, k$ do
  $s \leftarrow s - a_j b_j$
end for
$x \leftarrow \frac{s}{d}$
return $x$
```

The following theorem shows that the computed result $\hat{\mathbf{x}}$ solves a linear system which is close to the original linear system.

Theorem 4.12 *Assume $nu < 1$. If Algorithm 5 runs to completion, then the computed result $\hat{\mathbf{x}}$ satisfies a lower triangular linear system of the from*

$$\begin{bmatrix} l_{11}\langle 1 \rangle & & & & \\ l_{21}\langle 1 \rangle & l_{22}\langle 2 \rangle & & & \\ l_{31}\langle 1 \rangle & l_{32}\langle 2 \rangle & l_{33}\langle 3 \rangle & & \\ \vdots & \vdots & \vdots & \ddots & \\ l_{n1}\langle 1 \rangle & l_{n2}\langle 2 \rangle & \dots & \dots & l_{nn}\langle n \rangle \end{bmatrix} \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \\ \hat{x}_3 \\ \vdots \\ \hat{x}_n \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ \vdots \\ f_n \end{bmatrix}$$

Exercises

1. Analyse Algorithm 6. Show that the computed value $\hat{s}$ of s satisfies an equation of the form

$$\langle k \rangle \hat{s} = f - \sum_{j=1}^{k} \langle j \rangle a_j b_j$$

and use this result to prove Lemma 4.3.

2. Prove Theorem 4.12.
3. Write a function `lts` which solves a lower triangular linear system $Tx = f$ using forward substitution.
4. Consider the solution of $Tx = f$ where $T \in \mathbb{R}^{n\times n}$ is lower triangular and $f \in Rn$. Show forward substitution requires exactly n^2 arithmetic operations.

4.3 Running error analysis

In this section we show how to accurately estimate the error associated with each of the four elementary arithmetic operations: addition, subtraction, multiplication and division. In principle, this will allow us to estimate the error associated with *any* computation, such as the solution of linear systems of equations. We will later demonstrate this technique for sums, inner products, polynomials and triangular linear systems. In general, running error analysis is supplemented with more advanced techniques.

Now let $\circ$ denote one of the four elementary arithmetic operations. Let a and b be real numbers and let $r = a \circ b$ denote the result. In the case of division, we naturally assume that $b \neq 0$. In each case we are given approximations $\hat{a}$ and $\hat{b}$ as well as numbers α, and β for which

$$|a - \hat{a}| \leq \alpha u, \quad |b - \hat{b}| \leq \beta u, \tag{4.34}$$

where u denotes the unit round off error. Our task is to establish an error bound of the form

$$|r - \hat{r}| \leq \mu u \tag{4.35}$$

for a suitable value of μ.

Example 4.6 Find a suitable value of μ in the case of addition.

Solution The computed value $\hat{r}$ of the sum satisfies

$$(1+\delta)\hat{r} = (\hat{a} + \hat{b}) \quad \Rightarrow \quad \hat{r} = \hat{a} + \hat{b} - \delta\hat{r},$$

where δ satisfies $|\delta| \leq u$. It follows that

$$\begin{aligned} r - \hat{r} &= a + b - (\hat{a} + \hat{b} - \delta\hat{r}) \\ &= (a - \hat{a}) + (b - \hat{b}) + \delta\hat{r} \end{aligned}$$

which implies

$$|r - \hat{r}| \leq \alpha u + \beta u + |\hat{r}|u = (\alpha + \beta + |\hat{r}|)u.$$

We conclude that

$$\mu = \alpha + \beta + |\hat{r}| \tag{4.36}$$

ensures that the error bound given by (4.35) is satisfied. ■

Example 4.7 Find a suitable value of μ in the case of subtraction.

Solution The computed value $\hat{r}$ of the difference $r = a - b$ satisfies

$$(1+\delta)\hat{r} = (\hat{a} - \hat{b}) \quad \Rightarrow \quad \hat{r} = \hat{a} - \hat{b} - \delta\hat{r},$$

where δ satisfies $|\delta| \leq u$. It follows that

$$\begin{aligned} r - \hat{r} &= (a - b) - (\hat{a} - \hat{b} - \delta\hat{r}) \\ &= (a - \hat{a}) - (b - \hat{b}) + \delta\hat{r} \end{aligned}$$

which implies

$$|r - \hat{r}| \le \alpha u + \beta u + |\hat{r}|u = (\alpha + \beta + |\hat{r}|)u.$$

We conclude that

$$\mu = \alpha + \beta + |\hat{r}| \tag{4.37}$$

ensures that the error bound given by (4.35) is satisfied. ■

Remark 4.2 It is clear that analysis of subtraction is identical to the analysis of addition and we could have saved space by combining the two. However, there is no harm in stating matters clearly.

Example 4.8 Find a suitable value of μ in the case of multiplication.

Solution The computed value $\hat{r}$ of the product $r = ab$ satisfies

$$(1+\delta)\hat{r} = \hat{a}\hat{b} \quad \Rightarrow \quad \hat{r} = \hat{a}\hat{b} - \delta\hat{r},$$

where δ satisfies $|\delta| \le u$. It follows that

$$\begin{aligned} r - \hat{r} &= ab - (\hat{a}\hat{b} - \delta\hat{r}) \\ &= ab - \hat{a}\hat{b} + \delta\hat{r} \end{aligned}$$

We must now rewrite the expression to involve our error bounds. It is straightforward to verify that

$$ab - \hat{a}\hat{b} = (a - \hat{a})b + \hat{a}(b - \hat{b}) = (a - \hat{a})(b - \hat{b}) + (a - \hat{a})\hat{b} + \hat{a}(b - \hat{b}).$$

It follows

$$|r - \hat{r}| \le \left(\alpha\beta u^2 + |\hat{b}|\alpha + |\hat{a}|\beta + |\hat{r}|\right) u.$$

We conclude that the value

$$\mu = \alpha\beta u^2 + |\hat{b}|\alpha + |\hat{a}|\beta + |\hat{r}|. \tag{4.38}$$

ensures that the error bound given by (4.35) is satisfied. However, in practice, we omit the term $\alpha\beta u^2$ and use the simpler expression

$$\mu = |\hat{b}|\alpha + |\hat{a}|\beta + |\hat{r}|. \tag{4.39}$$

Problem 1 asks you to justify this simplification. ■

Example 4.9 Find a suitable value of μ in the case of division.

Solution The computed value $\hat{r}$ of the division $r = \frac{a}{b}$ satisfies

$$(1+\delta)\hat{r} = \frac{\hat{a}}{\hat{b}} \quad \Rightarrow \quad \hat{r} = \frac{\hat{a}}{\hat{b}} - \delta\hat{r},$$

where δ satisfies $|\delta| \le u$. It follows, that

$$r - \hat{r} = \frac{a}{b} - \frac{\hat{a}}{\hat{b}} + \delta\hat{r}$$

We must now rewrite this expression to involve our initial error bounds (4.34). It is straightforward to verify that

$$\frac{a}{b} - \frac{\hat{a}}{\hat{b}} = \frac{a\hat{b} - \hat{a}b}{b\hat{b}} = \frac{(a - \hat{a})\hat{b} - \hat{a}(b - \hat{b})}{b\hat{b}}.$$

It follows that

$$|r - \hat{r}| \le \left(\frac{\alpha|\hat{b}| + \beta|\hat{a}|}{|b||\hat{b}|} + |\hat{r}|\right) u.$$

We can not compute the right hand side of this expression, because b is not available. However, if $\hat{b}^{-1}$ is a good approximation of b^{-1}, then we can write

$$|r-\hat{r}| \lesssim \mu u, \quad \mu = \left(\frac{\alpha|\hat{b}| + \beta|\hat{a}|}{|\hat{b}|^2} + |\hat{r}| \right). \tag{4.40}$$

When is $\hat{b}^{-1}$ is a good approximation of b^{-1}? The relative error is given by

$$\frac{\frac{1}{b} - \frac{1}{\hat{b}}}{\frac{1}{b}} = \frac{\frac{\hat{b}-b}{b\hat{b}}}{\frac{1}{b}} = \frac{\hat{b}-b}{\hat{b}}.$$

It follows, that $\hat{b}^{-1}$ is a good approximation of b^{-1} if

$$\frac{\beta u}{|\hat{b}|} \ll 1, \tag{4.41}$$

and this condition can be verified at runtime. ■

The results of our investigation of the elementary arithmetic operations are summarized here.

operation	μ	conditions for use
addition	$\alpha + \beta + \lvert\hat{r}\rvert$	none
subtraction	$\alpha + \beta + \lvert\hat{r}\rvert$	none
multiplication	$\lvert\hat{\beta}\rvert\alpha + \lvert\hat{\alpha}\rvert\beta + \lvert\hat{r}\rvert$	$\beta u \ll \lvert\hat{b}\rvert$ and $\alpha u \ll \lvert\hat{a}\rvert$.
division	$\frac{\alpha\lvert\hat{b}\rvert+\beta\lvert\hat{a}\rvert}{\lvert\hat{b}\rvert^2} + \lvert\hat{r}\rvert$	$\beta u \ll \lvert\hat{b}\rvert$

Exercises

1. Determine when it is safe to omit the term $\alpha\beta u^2$ in equation (4.38) and instead use equation (4.39).

2. Find a running error bound ffor the formula $y = a + bx$. Consider the general case, where you are given approximation $\hat{a}$, $\hat{b}$, and $\hat{x}$ as well as error bounds of the form

$$|a-\hat{a}| \leq \mu_a u, \quad |b-\hat{b}| \leq \mu_b u, \quad |x-\hat{x}| \leq \mu_x u \tag{4.42}$$

4.3.1 Sums

Let $\mathbf{x} = (\mathbf{x_1}, \mathbf{x_2}, \dots, \mathbf{x_n})^{\mathbf{T}} \in \mathbb{R}^{\mathbf{n}}$. Let $s = \sum_{i=1}^n x_i$ denote the sum of the components of $\mathbf{x}$. Let $\hat{s}$ denote the computed sum obtained using Algorithm 2. We seek an error bound of the form

$$|s-\hat{s}| \leq \mu u, \tag{4.43}$$

where u is the unit roundoff error. To that end, let $\hat{s}_i$ denote the computed value of $s_i = \sum_{j=1}^i x_j$. We will obtain bounds of the form

$$|s_i - \hat{s}_i| \leq \mu_i u, \quad i = 1, 2, \dots, n. \tag{4.44}$$

It is clear that $\mu_1 = 0$, because x_1 is a floating point number. Now suppose that

$$|s_{i-1} - \hat{s}_{i-1}| \leq \mu_{i-1} u. \tag{4.45}$$

Since $s_i = s_{i-1} + x_i$, our analysis of addition shows that

$$|s_i - \hat{s}_i| \leq \mu_i u, \quad \mu_i = \mu_{i-1} + |\hat{s}_i|. \tag{4.46}$$

Algorithm 7 shows how to compute the sum together with this running error bound.

Algorithm 7 Summation with running error bound

Require: $\{x_i\}_{i=1}^n \in \mathbb{R}$

Ensure: $s = \sum_{i=1}^n x_i \in \mathbb{R}$, and $\mu \in \mathbb{R}$ such that

$$|s - \hat{s}| \leq \mu u \tag{4.47}$$

```
$s_1 \leftarrow x_1$
$\mu_1 \leftarrow 0$
for $i = 2, \dots, n$ do
    $s_i \leftarrow s_{i-1} + x_i$
    $\mu_i \leftarrow \mu_{i-1} + |s_i|$
end for
$s = s_n$
$\mu = \mu_n$
return $[s, \mu]$
```

Exercises

1. Show that the simple summation of n terms requires $n-1$ floating point operations. Show that including the running error bound increase the flop count to $2(n-1)$ floating point operations.

2. Show that simple summation requires at least n read operations. Does the calculation of the running error bound increase the number of read operations?

3. Implement and test the simple summation algorithm with a running error bound. Apply it to partial sums of the harmonic series. Sum the terms both in descending and in ascending order. Which method is the most accurate? Are the error bounds accurate?

4.3.2 Inner products

Let $\mathbf{x} \in \mathbb{R}^{\mathbf{n}}$ and $\mathbf{y} \in \mathbb{R}^{\mathbf{n}}$ and let s denote the inner product

$$s = \mathbf{x}^T \mathbf{y} = \sum_{i=1}^n x_i y_i.$$

Let $\hat{s}$ denote the computed value of s obtained using Algorithm 3. We seek an error bound of the form

$$|s - \hat{s}| \leq \mu u, \tag{4.48}$$

where u is the unit roundoff error. Let $\hat{s}_i$ denote the compute value of s_0. We will now derive error bounds of the form

$$|s_i - \hat{s}_i| \leq \mu_i u.$$

For the sake of simplicity we will ignore any errors on $\mathbf{x}$ and $\mathbf{y}$. It is clear that $\mu_0 = 0$ is a valid choice. In general,

$$s_i = s_{i-1} + t_i, \quad t_i = x_i y_i.$$

Let $\hat{t}_i$ denote the computed value of the product $t_i = x_i y_i$. Our analysis of multiplication shows that

$$|t_i - \hat{t}_i| \leq |\hat{t}_i| u, \tag{4.49}$$

because x_i and y_i are considered exact. Our analysis of addition shows that

$$|s_i - \hat{s}_i| \leq \mu_i u,$$

where

$$\mu_i = \mu_{i-1} + |\hat{t}_i| + |\hat{s}_i|. \tag{4.50}$$

Algorithm 8 shows how to compute the inner product together with this running error bound.

Algorithm 8 Inner product with running error bound

```
Require: $\mathbf{x}, \mathbf{y} \in \mathbb{R}^{\mathbf{n}}$
Ensure: $s = \mathbf{x}^{\mathbf{T}}\mathbf{y} \in \mathbb{R}$, and $\mu \in \mathbb{R}$ such that
                    $|s - \hat{s}| \leq \mu u$
$s_0 \leftarrow 0$
$\mu_0 \leftarrow 0$
for $i = 1, \dots, n$ do
    $t_i \leftarrow x_i y_i$
    $s_i \leftarrow s_{i-1} + t_i$
    $\mu_i \leftarrow \mu_{i-1} + |t_i| + |s_i|$
end for
$s \leftarrow s_n$
$\mu \leftarrow \mu_n$
return $[s, \mu]$
```

Exercises

4.3.3 Polynomials

Algorithm 9 extends Horner's method to include the computation of a running error bound. The algorithm ignores any errors in the input arguments, i.e., the coefficients a_j and the value t. Deriving the algorithm is an application of our analysis of addition and multiplication.

Exercises

1. Use the analysis of the basic arithmetic operations to derive Algorithm 9.

2. Extend your implementation of `XHorner` to also include the calculation of number μ which controls the running error bound given by Algorithm 9.

   (a) The new call sequence should be `[y, pt, mu]=XHorner(a,x)`, where `y, pt, a, x` are given by Problem 3 of Section 4.2.3.

   (b) Program a new minimal working example `XHornerMWE3` which computes the polynomial p given by $p(x) = (x-2)^3$ in double precision and applies Horner's method in single precision to the equivalent form.

   (c) Adjust `XHornerMWE3` until it can plot the target function, the computed approxmation, the error and the error bounds in a manner very similar to Fig. 4.3.

3. Show that Horner's method requires $2n$ floating point operations. Show that including the error bound increases the flop counts to $5n$ operations.

Algorithm 9 Horner's method with running error bound

Require: A real polynomial

$$p(x) = \sum_{j=0}^{n} a_j x^j \tag{4.51}$$

and $t \in \mathbb{R}$.

Ensure: The $y = p(t)$ and μ such that

$$|y - \hat{y}| \leq \mu u \tag{4.52}$$

```
$p_0 \leftarrow a_n$
$\mu_0 \leftarrow 0$
for $j = 1, \dots, n$ do
    $z_j \leftarrow p_{j-1} x$
    $p_j \leftarrow z_j + a_{n-j}$
    $\mu_j \leftarrow \mu_{j-1}|x| + |z_j| + |p_j|$
end for
$y \leftarrow p_n$
$\mu \leftarrow \mu_n$
return $y, \mu$
```

4. Show that Horner's method requires at least $n + 1$ read operations. Does the calculation of the running error bound increase the number of read operations?

4.3.4 Forward substitution

Let $\mathbf{L} \in \mathbb{R}^{\mathbf{n}\times\mathbf{n}}$ be a nonsingular lower triangular matrix and let $f \in \mathbb{R}^n$. The linear system

$$\mathbf{L}\mathbf{x} = \mathbf{f} \tag{4.53}$$

can be solved using forward substitution, i.e., Algorithm 5.

Let $\hat{\mathbf{x}}$ denote the computed valued of x. Algorithm 10 computes an error bound of the form

$$|x_i - \hat{x}_i| \leq \mu_i u \tag{4.54}$$

where u is the unit roundoff error. For the sake of simplicity, any initial error on t_{ij} or f_j is ignored.

Exercises

1. Apply the analysis of the elementary arithmetic operations to derive Algorithm 10.

Exercises

1. Show that forward substitution requires n^2 floating point operation and that the inclusion of the running error bound increases the flop count to $3n^2$.
2. Implement and test a routine for backward substitution which computes a running error bound. Good test examples include linear systems where you choose the solution $\mathbf{x}$ and compute the solution $\mathbf{f} = \mathbf{T}\mathbf{x}$, which is then used as input to your solver.

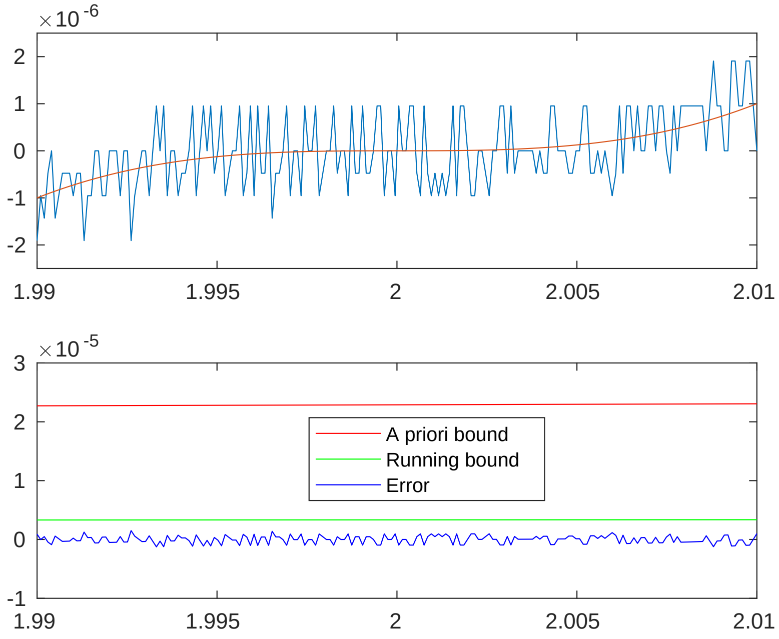

Figure 4.3: Computed polynomial values (top) and running and a priori bounds (bottom) for Horner's method using 200 equidistant points in the interval $[1.99, 2.01]$.

Algorithm 10 Forward substitution with running error bound

Require: A nonsingular lower triangular matrix $\mathbf{L} \in \mathbb{R}^{\mathbf{n}\times\mathbf{n}}$ and $\mathbf{f} \in \mathbb{R}^{\mathbf{n}}$.
Ensure: The solution $\mathbf{x}$ of $\mathbf{Lx} = \mathbf{f}$ and a vector $\mu \in \mathbb{R}^{\mathbf{n}}$, such that

$$|x_j - \hat{x}_j| \leq \mu_j u. \tag{4.55}$$

```
$\mathbf{x} \leftarrow \mathbf{f}$
$\mu \leftarrow \mathbf{0}$
for $i = 1, 2, \ldots, n$ do
    for $j = 1, 2, \ldots, i-1$ do
        $r_j \leftarrow t_{ij} x_j$
        $x_i \leftarrow x_i - r_i$
        $\mu_i \leftarrow \mu_i + (\mu_j |t_{ij}| + |r_i|) + |x_i|$
    end for
    $x_i \leftarrow \frac{x_i}{t_{ii}}$
    $\mu_i \leftarrow \frac{\mu_i}{|t_{ii}|} + |x_i|$
end for
```

4.4 Stability theory

Most computational problems can be reduced to the question of computing a function

$$f : A \to B.$$

The domain A is the set of valid input and the codomain B is the set of valid output.

In this section we seek to quantify how the output is affected by errors of the input. If the output is insensitive to errors in the input, then the problem is said to be well-conditioned. Otherwise, the problem is ill-conditioned.

We are equally concerned with the quality of our algorithms. If the output is insensitive to errors in the input, then the algorithm is said to be stable. Otherwise, the algorithm is unstable.

4.4.1 The conditioning of a problem

Before making any formal definitions, it is useful to investigate a few specific examples.

Example 4.10 Investigate how sensitive a sum $s(\mathbf{x}) = \sum_{i=1}^n x_i$ is to small perturbations of the input vector

$$\mathbf{x} = (x_1, x_2, \ldots, x_n)^T \in \mathbb{R}^n.$$

Solution Let $\mathbf{x} \in \mathbb{R}^n$ and let $\Delta\mathbf{x} \in \mathbb{R}^n$ denote the perturbation

$$\Delta x = (\Delta x_1, \Delta x_2, \cdots, \Delta x_n)^T.$$

In general, we have

$$s(\mathbf{x} + \Delta\mathbf{x}) = \sum_{i=1}^n (x_i + \Delta x_i) = \sum_{i=1}^n x_i + \sum_{i=1}^n \Delta x_i = s(\mathbf{x}) + \sum_{i=1}^n \Delta x_i.$$

Let $\epsilon > 0$ and consider perturbations for which the componentwise relative error satisfies

$$\max_{j=1,2\ldots,n} \left| \frac{\Delta x_j}{x_j} \right| = \epsilon. \tag{4.56}$$

Obviously, we have to assume $x_j \neq 0$ for all j, but this is not a severe restriction as we are interested in the sum. In this case, we have

$$|s(\mathbf{x} + \Delta\mathbf{x}) - s(\mathbf{x})| \leq \sum_{i=1}^n |\Delta x_i| = \sum_{i=1}^n \left| \frac{\Delta x_i}{x_i} \right| |x_i| \leq \epsilon \sum_{i=1}^n |x_i|.$$

Moreover, the special perturbation $\Delta\mathbf{x}'$ given by

$$\Delta x_i' = \epsilon |x_i|.$$

satisfies equation (4.56) and in this special case we have

$$s(\mathbf{x} + \Delta\mathbf{x}') - s(\mathbf{x}) = \sum_{i=1}^n \Delta x_i' = \epsilon \sum_{i=1}^n |x_i|.$$

We conclude that if $\Delta\mathbf{x}$ is a perturbation satisfying (4.56), then

$$\frac{|s(\mathbf{x} + \Delta\mathbf{x}) - s(\mathbf{x})|}{|s(\mathbf{x})|} \leq \epsilon \frac{\sum_{i=1}^n |x_i|}{|\sum_{i=1}^n x_i|}$$

and equality is possible. ■

Example 4.11 Investigate how sensitive an inner product $s(\mathbf{x}, \mathbf{y}) = \mathbf{x^T y}$ is to small perturbations of the input.

Solution Let $\Delta\mathbf{x}$ and $\Delta\mathbf{y}$ denote perturbations of $\mathbf{x}$ and $\Delta\mathbf{y}$. In general, we have

$$s(\mathbf{x} + \Delta\mathbf{x}, \mathbf{y} + \Delta\mathbf{y}) - s(\mathbf{x}, \mathbf{y}) = \mathbf{x}^T \Delta\mathbf{y} + \Delta\mathbf{x}^T \mathbf{y} + \Delta\mathbf{x}^T \Delta\mathbf{y}.$$

Let $\epsilon > 0$ and assume that

$$\max \left\{ \max_{j=1,2\ldots,n} \left| \frac{\Delta x_j}{x_j} \right|, \max_{j=1,2\ldots,n} \left| \frac{\Delta y_j}{y_j} \right| \right\} = \epsilon. \tag{4.57}$$

This is a straightforward generalization of (4.56). Obviously, we have to assume that $x_j \neq 0$ and $y_j \neq 0$ for all j, but this is not a severe restriction as we are interested in the inner product. In this case, we have

$$|s(\mathbf{x}+\Delta\mathbf{x},\mathbf{y}+\Delta\mathbf{y}) - s(\mathbf{x},\mathbf{y})| \leq (2\epsilon+\epsilon^2)\sum_{i=1}^n |x_i||y_i|.$$

This is a very good upper bound for the absolute value of the error. In particular, the special perturbations $\Delta\mathbf{x}'$ and $\Delta\mathbf{y}'$ given by

$$\Delta x_i' = \epsilon \cdot \operatorname{sign}(x_i y_i) x_i, \quad \Delta y_i' = \epsilon \cdot \operatorname{sign}(x_i y_i) y_i \tag{4.58}$$

satisfy (4.57) and

$$s(\mathbf{x}+\Delta\mathbf{x}',\mathbf{y}+\Delta\mathbf{y}') - s(\mathbf{x},\mathbf{y}) = 2\epsilon\sum_{i=1}^n |x_i||y_i| + \epsilon^2 \sum_{i=1}^n x_i y_i \geq (2\epsilon-\epsilon^2)\sum_{i=1}^n |x_i||y_i|.$$

We conclude that if the perturbations $\Delta\mathbf{x}$ and $\Delta\mathbf{y}$ satisfy (4.57), then

$$\frac{|s(\mathbf{x}+\Delta\mathbf{x},\mathbf{y}+\Delta\mathbf{y}) - s(\mathbf{x},\mathbf{y})|}{|s(\mathbf{x},\mathbf{y})|} \leq (2\epsilon+\epsilon^2)\frac{\sum_{i=1}^n |x_i||y_i|}{|\sum_{i=1}^n x_i y_i|}.$$

Moreover, the very special perturbations given by (4.58) *also* satisfy

$$\frac{|s(\mathbf{x}+\Delta\mathbf{x}',\mathbf{y}+\Delta\mathbf{y}') - s(\mathbf{x},\mathbf{y})|}{|s(\mathbf{x},\mathbf{y})|} \geq (2\epsilon-\epsilon^2)\frac{\sum_{i=1}^n |x_i||y_i|}{|\sum_{i=1}^n x_i y_i|}.$$

■

Example 4.12 Let $x \in \mathbb{R}$ be fixed and consider the problem of computing $p(\mathbf{a}) = \sum_{j=0}^n a_j x^j$. Investigate how sensitive $y = p(\mathbf{a})$ is to small perturbations of the coefficients a_j.

Solution Let $\mathbf{a} \in \mathbb{R}^{n+1}$ denote the vector of coefficients

$$\mathbf{a} = (a_0, a_1, a_2, \ldots, a_n)^T.$$

and let $\Delta\mathbf{a}$ denote the perturbation

$$\Delta\mathbf{a} = (\Delta a_0, \Delta a_1, \Delta a_2, \ldots, \Delta a_n)^T.$$

In general, we have

$$p(\mathbf{a}+\Delta\mathbf{a}) = \sum_{j=0}^n (a_j + \Delta a_j)x^j = \sum_{j=0}^n a_j x^j + \sum_{j=0}^n \Delta a_j x^j = p(\mathbf{a}) + \sum_{j=0}^n |\Delta a_j||x|^j.$$

Let $\epsilon > 0$ and consider perturbations $\Delta\mathbf{a}$ for which

$$\max_{j=0,1,2,\ldots,n} \left|\frac{\Delta a_j}{a_j}\right| = \epsilon. \tag{4.59}$$

In this case we have

$$|p(\mathbf{a}+\Delta\mathbf{a}) - p(\mathbf{a})| \leq \epsilon \sum_{j=0}^n |\Delta a_j||x|^j.$$

Moreover, the special perturbation $\Delta\mathbf{a}'$ given by

$$\Delta a_j' = \epsilon \cdot \operatorname{sign}(a_j x^j) a_j$$

satisfies equation (4.59) and

$$p(\mathbf{a}+\Delta\mathbf{a}') - p(\mathbf{a}) = \sum_{j=0}^{n} \Delta a_j' x^j = \epsilon \sum_{j=0}^{n} |a_j||x|^j.$$

We conclude that if $\Delta\mathbf{a}$ is a perturbation which satisfies equation (4.59), then

$$\frac{|p(\mathbf{a}+\Delta\mathbf{a}) - p(\mathbf{a})|}{|p(\mathbf{a})|} \leq \epsilon \frac{\sum_{j=0}^{n} |a_j||x|^j}{\left|\sum_{j=0}^{n} a_j x^j\right|}$$

and equality is possible. ■

Our goal is to generalize these examples and find a sensible way to define the conditioning of a problem. Consider a real valued function $f : A \to \mathbb{R}$ defined on a subset $A \subseteq \mathbb{R}^m$. Let $\mathbf{x} \in A$. We seek to quantify how sensitive the output is to small changes of the input. Let $\mathbf{y} \in A$ and view $f(\mathbf{y})$ as an approximation of $f(\mathbf{x})$. Then the absolute value of the relative error is

$$\sigma(f(\mathbf{x}), f(\mathbf{y})) = \frac{|f(\mathbf{y}) - f(\mathbf{x})|}{|f(\mathbf{x})|}.$$

There is more than one way to define the relative error between $\mathbf{x}$ and $\mathbf{y}$. One choice is the componentwise relative error, i.e.

$$\sigma(x, y) = \max_{j=1,2,\dots,n} \left| \frac{x_j - y_j}{x_j} \right| \tag{4.60}$$

but the normwise relative error, i.e.,

$$\sigma(x, y) = \frac{\|x - y\|_2}{\|x\|_2} \tag{4.61}$$

is another possibility.

In general, we wish to bound the ratio between the relative error of the output and the relative error of the input. If $\sigma(x, x + \Delta x) \leq \epsilon$, then the worst case behavior is characterized by the number

$$\kappa_f(\mathbf{x}, \nu) = \sup \left\{ \frac{\sigma(f(\mathbf{x}), f(\mathbf{x} + \Delta x))}{\sigma(\mathbf{x}, \mathbf{x} + \Delta \mathbf{x})} \mid \Delta\mathbf{x} \,:\, \sigma(\mathbf{x}, \mathbf{x} + \Delta\mathbf{x}) \leq \nu \right\}.$$

We observe that $\kappa_f(\mathbf{x}, \nu)$ is a monotone increasing function of ν, simply because as ϵ increases, the supremum is over an increasing set of perturbations $\Delta\mathbf{x}$. It follows that the limit

$$\lim_{\nu \to 0_+} \kappa_f(\mathbf{x}, \nu)$$

exists. Since we are primarily interested in small perturbations, the following definition is sensible.

Definition 4.5 *Let $A \subseteq \mathbb{R}^m$ and let $f : A \to \mathbb{R}$ be a function and let $x \in A$. If $f(x) \neq 0$, then the condition number of f at the point x is given by*

$$\kappa_f(\mathbf{x}) = \lim_{\nu \to 0+} \sup \left\{ \frac{\sigma(f(x), f(x + \Delta x))}{\sigma(x, x + \Delta x)} \,:\, \sigma(x, x + \Delta x) \leq \nu \right\}.$$

It remains to be seen if this definition is useful for practical calculations.

Example 4.13 Find the componentwise condition number of a sum $s = \sum_{i=1}^{n} x_i$.

Solution Let $\nu > 0$ and let

$$\sigma(\mathbf{x}, \mathbf{x}+\Delta\mathbf{x}) = \max_j \left|\frac{\Delta x_j}{x_j}\right| = \epsilon \leq \nu$$

Then by Example 4.10 we know that

$$\frac{1}{\epsilon}\frac{|s(\mathbf{x}+\Delta\mathbf{x}) - s(\mathbf{x})|}{|s(\mathbf{x})|} \leq \frac{\sum_{i=1}^n |x_i|}{|\sum_{i=1}^n x_i|}$$

and equality *is* possible. It follows immediately, that

$$\kappa_f(x,\nu) = \frac{\sum_{i=1}^n |x_i|}{|\sum_{i=1}^n x_i|}$$

and

$$\kappa_f(x) = \frac{\sum_{i=1}^n |x_i|}{|\sum_{i=1}^n x_i|}.$$

■

We conclude that if the terms x_i have the *same* sign, then the sum $s = \sum_{i=1}^n x_i$ is well conditioned. However, if the terms have different signs, then the sum can be very ill conditioned. This happens when $|\sum_{i=1} x_i|$ is tiny, but $\sum_{i=1}^n |x_i|$ is large.

Example 4.14 Find the componentwise condition number of an inner product $s(\mathbf{x},\mathbf{y}) = \mathbf{x}^T\mathbf{y}$.

Solution Let $\nu > 0$ and let

$$\sigma((\mathbf{x},\mathbf{y}), (\mathbf{x}+\Delta\mathbf{x}, \mathbf{y}+\Delta\mathbf{y}) = \max\left\{\max_j \left|\frac{\Delta x_j}{x_j}\right|, \max_j \left|\frac{\Delta y_j}{y_j}\right|\right\} = \epsilon$$

Then by Example 4.11 we know that

$$\frac{1}{\epsilon}\frac{|s(\mathbf{x}+\Delta\mathbf{x}, \mathbf{y}+\Delta\mathbf{y}) - s(\mathbf{x},\mathbf{y})|}{|s(\mathbf{x},\mathbf{y})|} \leq (2+\epsilon)\frac{\sum_{i=1}^n |x_i||y_i|}{|\sum_{i=1}^n x_i y_i|}.$$

Moreover, there are special perturbations which satisfy equation (4.57) and

$$\frac{1}{\epsilon}\frac{|s(\mathbf{x}+\Delta\mathbf{x}, \mathbf{y}+\Delta\mathbf{y}) - s(\mathbf{x},\mathbf{y})|}{\epsilon|s(\mathbf{x},\mathbf{y})|} \geq (2-\epsilon)\frac{\sum_{i=1}^n |x_i||y_i|}{|\sum_{i=1}^n x_i y_i|}.$$

It follows that

$$(2-\epsilon)\frac{\sum_{i=1}^n |x_i||y_i|}{|\sum_{i=1}^n x_i y_i|} \leq \kappa_f(\mathbf{x},\epsilon) \leq (2+\epsilon)\frac{\sum_{i=1}^n |x_i||y_i|}{|\sum_{i=1}^n x_i y_i|}.$$

In the limit, when ϵ tends to zero, we find

$$\kappa_f(\mathbf{x}) = 2 \cdot \frac{\sum_{i=1}^n |x_i||y_i|}{|\sum_{i=1}^n x_i y_i|}.$$

■

We conclude that if x_j and y_j have the same sign for all j, then the inner product $s(\mathbf{x},\mathbf{y}) = \mathbf{x}^T\mathbf{y}$ is well conditioned. However, if the vectors are nearly orthogonal, then the inner product can be ill conditioned. This happens when $|\mathbf{x}^T\mathbf{y}|$ is small while $|\mathbf{x}|^T|\mathbf{y}|$ is large.

Example 4.15 Let $x \in \mathbb{R}$. Find the condition number of $p(\mathbf{a}) = \sum_{j=0}^n a_j x^j$.

Solution Let $\nu > 0$ and let

$$\sigma(\mathbf{a}, \mathbf{a} + \Delta a) = \max_j \left| \frac{\Delta a_j}{a_j} \right| = \epsilon \leq nu$$

Then by Example 4.12 we know that

$$\frac{1}{\epsilon} \frac{|p(\mathbf{a} + \Delta \mathbf{a}) - p(\mathbf{a})|}{p(\mathbf{a})} \leq \frac{\sum_{j=0}^{n} |a_j||x|^j}{\left|\sum_{j=0}^{n} a_j x^j\right|}.$$

and equality is possible for special perturbations. It follows immediately that

$$\kappa_p(\mathbf{a}, \nu) = \frac{\sum_{j=0}^{n} |a_j||x|^j}{\left|\sum_{j=0}^{n} a_j x^j\right|}$$

and

$$\kappa_p(\mathbf{a}) = \frac{\sum_{j=0}^{n} |a_j||x|^j}{\left|\sum_{j=0}^{n} a_j x^j\right|}.$$

■

We conclude that if $a_j x^j$ has the same sign for all j, then the problem of computing $p(\mathbf{a})$ is well conditioned. However, if x is such that $p(\mathbf{a})$ is nearly zero, then the problem can be ill conditioned. This happens when $\left|\sum_{j=0}^{n} a_j x^j\right|$ is small and $\sum_{j=0}^{n} |a_j||x|^j$ is large.

Theorem 4.13 *Let $f \in C^1(\mathbb{R}, \mathbb{R})$. If $x \neq 0$ and if $f(x) \neq 0$, then*

$$\kappa_f(x) = \left| \frac{x f'(x)}{f(x)} \right|$$

Proof. Let Δx denote any perturbation. Then by the mean value theorem

$$f(x + \Delta x) - f(x) = f'(\xi)\Delta x = f'(x)\Delta x + (f'(\xi) - f'(x))\Delta x$$

for at least one ξ between x and $x + \Delta x$. Now let $\nu > 0$ and let $|\Delta x| = \epsilon |x|$ where $\epsilon \leq \nu$. Then by the continuity of f' we have

$$f'(\xi) = f'(x) + O(x - \xi).$$

This means there exists a constant $C > 0$ such that

$$|f'(\xi) - f'(x)| \leq C|x|\epsilon$$

We can now conclude that

$$\frac{|f'(x + \Delta x) - f(x)|}{|f(x)|} = \frac{|x||f'(x)|}{|f(x)|}\epsilon + O(\epsilon^2)$$

Therefore

$$\kappa_f(x, \nu) = \frac{|x||f'(x)|}{|f(x)|} + O(\nu)$$

and

$$\kappa_f(x) = \frac{|x||f'(x)|}{|f(x)|}.$$

This completes the proof. ■

Exercises

1. Let $a, b \in \mathbb{R}$. Show that the problem of computing the product $r = ab$ is well conditioned.
2. Let $a, b \in (0, \infty)$. Consider the problem of computing the sum $r = a + b$. Show that the condition number is 1 by direct application of the defintion.
3. Let $a, b \in \mathbb{R}$ with $a \neq 0$. Show that problem of computing of solving the single linear equation $ax = b$ with respect to x is well conditioned
4. Let $f : \mathbb{R} \rightarrow \mathbb{R}$ be given by $f(x) = e^x$. Showt that the condition number of f at the point $x \neq 0$ is given by $\kappa_f(x) = |x|$.
5. Let $f : \mathbb{R} \rightarrow \mathbb{R}$ be given by $f(x) = \cos(x)$. Show that the condition number of f at the point $x \neq 0$ is given by $\kappa_f(x) = |x \tan(x)|$
6. Let $f : \mathbb{R} \rightarrow \mathbb{R}$ be given by $f(x) = \sin(x)$. Show that f is well conditioned when $x \neq 0$, but $x \approx 0$.
7. Let $f, g : \mathbb{R} \rightarrow \mathbb{R}$ be differentiable functions and let $h = f \circ g$ denote the composition of f and g. Let $x \not0$ be such that $y = g(x) \neq 0$ and $f(y) \neq 0$. Show that the condition number of h at x is given by
$$\kappa_h(x) = \kappa_f(y)\kappa_g(x) \tag{4.62}$$
8. Explain why it is possible for a team of programmers to develop a large modular program where each module works perfectly, but the complete program produces nothing but garbage! Hint: Generalize Exercise 7 to the case of $m > 2$ functions.

4.4.2 The stability of an algorithm

Let $f : A \rightarrow \mathbb{R}$ and consider the problem of computing

$$y = f(x)$$

using a particular algorithm.

Definition 4.6 *If for each $x \in A$, the computed value $\hat{y}$ of $y = f(x)$ satisfies*

$$\hat{y} = f(\bar{x})$$

for at least one $\bar{x} \in A$ and if

$$\sigma(x, \bar{x}) = O(u)$$

where u is the unit roundoff, then the algorithm is said to be backwards stable.

Example 4.16 Analyse the stability of Algorithm 2 for computing sums.

Solution We assume $nu < 1$. By Theorem 4.2, the computed sum can be written as

$$\hat{s} = \sum_{j=1}^{n} x_j \langle k_j \rangle$$

where $k_j = \min\{n - 1, n - j + 1\} \leq n - 1$. We see that the computed sum is the exact sum corresponding to a vector $\bar{x} = x + \Delta x$ where $|\Delta x| \leq \gamma_{n-1}|x|$. This implies that

$$\sigma(x, \bar{x}) \leq \gamma_{n-1}$$

We conclude, that Algorithm 2 is (componentwise) backward stable. ■

By Theorem 4.4 the relative error satisfies

$$\frac{|s-\hat{s}|}{|s|} \leq \gamma_{n-1} \frac{\sum_{j=1}^n |x_j|}{\left|\sum_{j=1}^n x_j\right|}. \tag{4.63}$$

We recognize the term

$$\kappa_s(\mathbf{x}) = \frac{\sum_{j=1}^n |x_j|}{\left|\sum_{j=1}^n x_j\right|}.$$

as the condition number of the sum. We conclude that the relative forward error is small, when the sum s is well conditioned.

Example 4.17 Analyse the stability of Algorithm 3 for computing inner products.

Solution We assume $nu < 1$. By Theorem 4.5 the computed inner product can be written as

$$\hat{s} = \sum_{i=1}^n x_i y_i \langle m_i \rangle,$$

for $m_i = \min\{n, n-i+2\}$. We see that the computed inner product is in fact the exact inner product of, say, $\mathbf{x}$ with $\mathbf{z}$, where

$$z_i = y_i \langle m_i \rangle.$$

Since z_i is only a relative small perturbation of y_i, we conclude that Algorithm 3 is (componentwise) backward stable. -+ If $s \neq 0$, then by Theorem 4.7 the computed inner product satisfies

$$\frac{|s-\hat{s}|}{|s|} \leq \gamma_n \frac{\sum_{i=1}^n |x_i||y_i|}{|\sum_{i=1}^n x_i y_i|}. \tag{4.64}$$

We recognize the condition number

$$\kappa_s(\mathbf{x}, \mathbf{y}) = \frac{\sum_{i=1}^n |x_i||y_i|}{|\sum_{i=1}^n x_i y_i|}.$$

We conclude that the forward relative error is small, when the inner product is well conditioned.

■

Exercises

1. Let $a, b \in \mathbb{R}$ with $a \neq 0$ and consider the problem of solving $ax = b$. Devise an algorithm which is backward stable.

4.5 Subtractive cancellation

Consider the problem of computing the difference $d = a - b$. If a and b are floating point numbers, then the difference is computed with a small relative error. In fact, the computed value $\hat{d}$ of d satisfies

$$\hat{d} = (a-b)(1+\delta), \quad |\delta| \leq u$$

where u is the unit round off error. However, in general a and b are real numbers and we only have approximations $\hat{a} \approx a$ and $\hat{b} \approx b$. In this case, the difference $d = a - b$ is not necessarily computed with a small relative error. This phenomenon is known as subtractive cancellation.

Example 4.18 Plot the function $f : \mathbb{R} \to \mathbb{R}$ given by

$$f(x) = \begin{cases} \frac{x-\sin(x)}{x^3}, & x \neq 0, \\ \frac{1}{6}, & x = 0. \end{cases} \tag{4.65}$$

for small values of x and comment on the quality of the results.

Solution Figure 4.4 shows the MATLAB commands used to generate Figure 4.5. The real function f is continuous for all x, even $x = 0$ and $f(x) \to \frac{1}{6}$ for $x \to 0$. This special case of $x = 0$ can be investigated by repeated application of l'Hospital's rule, see Exercise 1. However, the *computed* graph indicates that f tends to 0, rather than $\frac{1}{6}$, as x tends 0. Moreover, the very rapid oscillations f is also inconsistent with our expectation. It is clear, that we must find another way to compute f, at least for small values of x. ■

```
f=@(x)(x-sin(x))./x.^3;
x=linspace(-1,1,1025)*2^-22
plot(x,f(x)); grid; grid MINOR; axis([min(x),max(x),-0.05,0.35]);
```

Figure 4.4: The MATLAB script used to generate the Figure 4.5

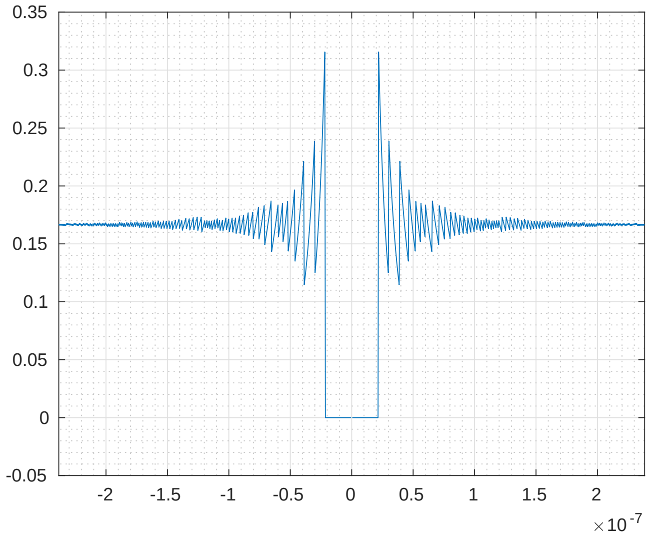

Figure 4.5: The graph generated by the script given in Figure 4.4.

We have the following theorem.

Theorem 4.14 *Suppose that $a \neq b$ are real numbers which are approximated with floating point numbers $\hat{a}$ and $\hat{b}$ for which*

$$\hat{a} = a(1+\Theta_a), \qquad |\Theta_a| \leq \gamma_m,$$
$$\hat{b} = b(1+\Theta_a), \qquad |\Theta_b| \leq \gamma_n.$$

Then the compute difference $\hat{d}$ satisfies the relative error bound

$$\left| \frac{d-\hat{d}}{d} \right| \leq \gamma_{k+1} \frac{|a|+|b|}{|a-b|}, \quad k = \max\{m,n\} \tag{4.66}$$

Proof. In order to bound the relative error we require an expression for the computed result. By the fundamental model of floating point calculations we have

$$\hat{d} = (\hat{a} - \hat{b})(1 + \delta), \quad |\delta| \leq u.$$

Inserting the expression for $\hat{a}$ and $\hat{b}$ yields

$$\begin{aligned}\hat{d} &= [a(1 + \Theta_a) - b(1 + \Theta_b)]\,(1 + \delta) \\ &= a(1 + \Theta_a)(1 + \delta) - b(1 + \Theta_b)(1 + \delta) \\ &= a(1 + \Theta_a') - b(1 + \Theta_b')\end{aligned}$$

where

$$|\Theta_a'| \leq \gamma_{k+1}, \quad |\Theta_b'| \leq \gamma_{k+1}.$$

It follows that the error $d - \hat{d}$ can be written as

$$d - \hat{d} = b\Theta_b' - a\Theta_a'.$$

This expression of the error allows us to estimate

$$\left|\frac{d - \hat{d}}{d}\right| \leq \frac{|b||\Theta_b'| + |a||\Theta_a'|}{|a - b|} \leq \gamma_{k+1}\frac{|a| + |b|}{|a - b|}.$$

This completes the proof. ■

If $a \approx b$, then we can not be certain that the difference d is computed with a small relative error. In practice, the relative error will be large if $a \approx b$. In order to avoid this problem it is necessary to replace the offending expression with another equivalent expression which does not suffer from subtractive cancellation.

Example 4.19 Find a way to approximate the function f given in Example 4.18 which does not suffer from subtractive cancellation when $x \approx 0$.

Solution By Taylor's theorem we

$$\sin(x) = x - \frac{1}{6}x^3 + \frac{1}{120}x^5 + O(x^7). \tag{4.67}$$

It follows that

$$f(x) = \frac{x - \sin(x)}{x^3} = \frac{1}{6} - \frac{1}{120}x^2 + O(x^4), \quad x \neq 0 \tag{4.68}$$

The new expression contains a subtraction, but it can not cancel catastrophically, because the to terms $a(x) = \frac{1}{6}$ and $b = b(x) = \frac{1}{120}x^2$ are very different for small values of x. ■

Example 4.20 Let $f : \mathbb{R} \to \mathbb{R}$ be given by

$$f(x) = \sqrt{x^2 + 4} - 2.$$

Show that the naive computation of f suffers from subtractive cancellation when $x \approx 0$ and find a way to avoid the problem.

Solution Let $a = a(x) = \sqrt{x^2 + 4}$ and let $b = b(x) = 2$. Since a is a continuous function and $a(0) = 2 = b(0)$ is clear that the computation $f(x) = a(x) - b(x)$ can suffer from subtractive cancellation for small values oof x. However,

$$f(x) = (\sqrt{x^2 + 4} - 2)\frac{\sqrt{x^2 + 4} + 2}{\sqrt{x^2 + 4} + 2} = \frac{(\sqrt{x^2 + 4} - 2)(\sqrt{x^2 + 4} + 2)}{\sqrt{x^2 + 4} + 2} = \frac{x^2}{\sqrt{x^2 + 4} + 2} = g(x)$$

Figure 4.6 shows a plot of the graph of f using the original definition as well as the (mathematically) equivalent representation g. It is clear that g reproduces the expected behavior of f, a non-negative function with a single zero at $x = 0$, strictly decreasing for $x < 0$ and strictly increasing for $x > 0$. ■

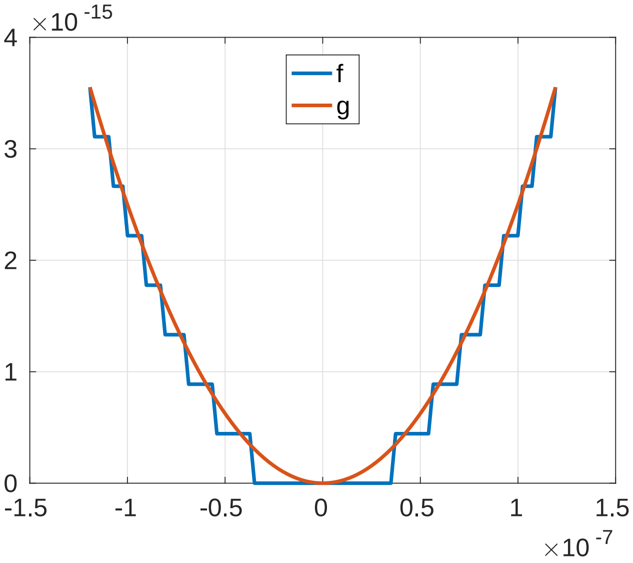

Figure 4.6: Reproducing the graph of the function $f(x) = \sqrt{4+x^2} - 2$ using both the original expression as well as $g(x) = x^2/(\sqrt{4+x^2}+2)$.

.

Subtractive cancellation can occur when a and b are "close". It is useful to know when subtractive cancellation cannot occur. The following corollary gives a clear answer to this question.

Corollary 4.3 *Let a, b, $\hat{a}$, $\hat{b}$, and $d = a - b$ be as in Theorem 4.14. If $|a| \geq 2|b|$ or $|b| \geq 2|a|$, then the computed difference $\hat{d}$ satisfies*

$$\left|\frac{d-\hat{d}}{d}\right| \leq 3\gamma_{k+1}$$

Proof. Left to the reader as Exercise 2. ■

In short, if one of the two arguments a and b is at least twice as large as the other in absolute value, then the difference $d = a - b$ can be computed with only a modest increase in the relative error.

Example 4.21 Determine when subtractive cancellation is not an issue in the subtraction $x - \sin(x)$.

Solution Let $a = a(x) = x$ and $b = b(x) = \sin(x)$. We want to ensure that $|a| \geq 2|b|$ or vise versa. However, since $|b| = |\sin(x)| \leq 1$, we have $|a| \geq 2|b|$ provided $|a| \geq 2$. We conclude that subtractive cancellation is a non-issue for $|x| \geq 2$. However, inside the interval $|x| \geq 2$ the impact of subtractive cancellation increases as we approach 0. ■

Exercises

1. Let $T(x) = x - \sin(x)$ and let $N(x) = x^3$. Show that $T(x)/N(x) \to \frac{1}{6}$ by repeated application of l'Hospital's rule.

2. Let a and b be real numbers such that $|a| \geq 2|b|$. Prove that

$$|a| + |b| \leq 3(|a| - |b|).$$

Hint: Exploit that $|a| - |b| \geq |b|$ and $|a| - |b| \geq \frac{1}{2}|a|$.

3. Consider the function $f : \mathbb{R} \to \mathbb{R}$ given by

$$f(x) = \begin{cases} \frac{e^x - 1}{x} & x \neq 0 \\ 1 & x = 0. \end{cases} \tag{4.69}$$

   (a) Show that f is continuous for all x.

   (b) Show that f is differentiable for $x = 0$.

   (c) Find a way to avoid subtractive cancellation for $x \approx 0$.

4. Consider the function $f : \mathbb{R} \to \mathbb{R}$ given by

$$f(x) = \begin{cases} \frac{1-\cos(x)}{x^2} & x \neq 0 \\ \frac{1}{2} & x = \frac{1}{2} \end{cases} \tag{4.70}$$

   (a) Show that f is continuous at $x = 0$.

   (b) Plot f using the definition for $x \approx 0$. What physical evidence do you find to indicated this approach is unreliable?

   (c) Find a way to avoid subtractive cancellation for $x \approx 0$.

4.6 General advice

As developer you are responsible for writing code which is reliable and produces accurate results. Speed is irrelevant if the program can fail or produces inaccurate results. This section contains some general advice for writing numerical programs which are reliable and accurate. Many of the entries on list are due to Ole Østerby from DAIMI at Aarhus University, Denmark and are copied almost ad verbatim from his notes.

1. **A good approximation plus a small correction term**

   In many calculations, typically iterations, we shall compute better and (hopefully) better approximations to the solution. If the calculations can be arranged as above, this will always be advantageous. For example, when computing an average of two real numbers

$$a + \frac{b-a}{2} \quad \text{is better than} \quad \frac{a+b}{2},$$

   and when executing Newton's method for computing the square root of a real number s, then

$$x - \frac{x^2 - s}{2x} \quad \text{is better than} \quad \frac{1}{2}\left(x + \frac{s}{x}\right).$$

2. **Add the small terms first**

   When adding a large number of positive terms you must sum the smallest terms first, otherwise their contribution, which can be significant, is never felt. If the number of terms is extremely large, then you should consider tree-wise addition. If you have both negative and positive terms, then you need to be extra careful, as you might experience catastrophic cancellation, see item 3.

3. **Be very careful when subtracting two *real* numbers which are almost equal**

   This point is frequently misunderstood. By design, the computer can subtract two *floating point* numbers x and y with a small relative error and

$$\text{fl}(x - y) = (x - y)(1 + \delta), \quad |\delta| \leq u.$$

   where $fl(x - y)$ is the computed (floating point representation) of $x - y$. So where is the problem? Given a pair of *real* numbers, the machine must first create a floating point representation of x and y and only then can it subtract the two representations. It is the *combination* of these two operations which can result in disaster. Specifically, we have

$$\frac{|(x-y) - \mathrm{fl}(\mathrm{fl}(x) - \mathrm{fl}(x))|}{|x-y|} \leq \frac{2u}{1-2u}\frac{|x|+|y|}{|x-y|}$$

In short, if x and y are almost equal, your relative error bound can be very large and you can not necessarily trust the computed value for $x - y$.

4. **Avoid large intermediate results on the road to a small final result**

   A good example is the naive computation of $x \to e^x$. The Taylor series expansion of $\exp(x)$ at $x_0 = 0$, works fine for positive values of x, but we get negative results for many negative values of x.

5. **Use mathematical reformulations to avoid 3. and 4.**

   For small values of x
   $$2\sin^2\frac{x}{2} \quad \text{is better than} \quad 1-\cos(x)$$
   and
   $$\frac{x^4}{\sqrt{x^4+2}+2} \quad \text{is better than} \quad \sqrt{x^4+4}-2.$$
   For negative values of x,
   $$1\Big/\sum_{j=0}^{n}\frac{(-x)^j}{j!} \quad \text{is a better approximation of } e^x = 1/e^{-x} \text{ than} \quad \sum_{j=0}^{n}\frac{x^j}{j!},$$
   and it completely eliminates the specific problem mentioned in item 4.

6. **Use series expansions to supplement 5**

   For small values of x
   $$\frac{1}{2} - \frac{x^2}{24} + \frac{x^4}{720} - \frac{x^6}{40320} + \ldots \quad \text{is better than} \quad \frac{1-\cos(x)}{x^2}$$

7. **Use integer calculations whenever possible**

   The following MATLAB command will never terminate

   ```
   >>b=0; a=1/1000; while (b~=1) b=b+a; end;
   ```

   although we would expect it to terminate after 1000 iterations! It is far safer to use an integer to control the loop, as in

   ```
   >>b=0; a=1/1000: for i=1:1000 b=b+a; end;
   ```

   Similarly, it you should pass integers to subroutines whenever possible.

8. **Be wary of very small number or very large numbers**

   In exact arithmetic
   $$\sqrt{ab} = \sqrt{a}\sqrt{b},$$
   when a and b are positive real numbers. However, if a and b are too small (too large) then $\sqrt{ab}$ underflows (overflows), while $\sqrt{a}\sqrt{b}$ will succeed. This is due to the fact that there is a smallest (largest) positive floating point number.

9. **Look at the numbers once in a while**

   It is a good idea to (write and) check intermediate results, e.g. while testing the program, and judge whether they make sense. Sound judgment and common sense are invaluable helpers. The experienced programmer will always develop a set of test cases where she knows the results in advance. Such test cases will catch many errors, but you must work even harder to rigourously prove that your program works.

Chapter 5

Functions

Let $I \subset \mathbb{R}$ be an interval and let $f : I \to \mathbb{R}$ be a function. In this chapter we consider the problem of approximationg $f(x)$ for $x \in I$.

5.1 Fundamental considerations

In general, there are three fundamental problems when computing $f(x)$ using finite precision arithmetic.

1. We cannot input $x \in I$ into our program!

   (a) The number $x \in I$ is not necessarily a machine number.

   (b) The number $x \in I$ is output of another function/subroutine.

   In both cases, we are forced to replace x with an approximation $\hat{x}$.

2. It is not necessarily practical or necessary to compute $f(\hat{x})$.

   (a) We do not have the time to evaluate $f(\hat{x})$

   (b) We do not need the exact value of $f(\hat{x})$.

   Instead we must compute an approximation $f(\hat{x}) \approx p(\hat{x})$.

3. We can not compute the exact value of $p(\hat{x})$.

   (a) Rounding errors are inevitable even when p is a polynomial

   Instead we compute an approximation $\hat{y} \approx p(\hat{x})$.

The error is th
The (absolute) error can be written as a sum of three terms

$$f(x) - \hat{y} = (f(x) - f(\hat{x})) + (f(\hat{x}) - p(\hat{x})) + (p(\hat{x}) - \hat{y})$$

Discuss the three terms separately

1. Condition number

2. Approximation

3. Implementation

Characterize a good implementation of a function

5.2 Polynomials

Let $p : \mathbb{R} \to \mathbb{R}$ be a real polynomial of degree n, i.e.

$$p(x) = a_0 + a_1 x + a_2 x^2 + \ldots + a_n x^n,$$

where the coefficients $a_j \in \mathbb{R}$ and $a_n \neq 0$. We can compute $p(x)$ using Horner's method which is given as Algorithm 11.

Algorithm 11 Horner's method

```
$p_0 \leftarrow a_n$
for $j = 1, \ldots, n$ do
    $p_j \leftarrow p_{j-1} x + a_{n-j}$
end for
$y = p_n$
return $y$
```

Example 5.1 Show that Horner's method returns the correct result in exact arithmetic for all polynomials of degree $n = 2$.

Solution Let p denote a polynomial of degree 2, i.e.,

$$p(x) = a_0 + a_1 x + a_2 x^2.$$

By stepping through the algorithm we see that Horner's method computes

$$\begin{aligned} p_0 &= a_2, \\ p_1 &= p_0 x + a_1 = a_2 x + a_1, \\ p_2 &= p_1 x + a_0 = (a_2 x + a_1)x + a_0 = a_2 x^2 + a_1 x + a_0, \end{aligned}$$

and returns $y = p_2 = p(x)$ as expected. ■

It is no surprise that we have the following result.

Theorem 5.1 *In exact arithmetic, Algorithm 11 returns $y = p(x)$*

Proof. ■

In practice, the computed result will be affected by rounding errors. In order to bound the error we must first derive a representation of the computed values.

We start with two small example, i.e., $n = 2$ and $n = 3$.

Example 5.2 Find a representation of the computed value returned by Horner's method when $n = 2$.

Solution Let $\hat{p}_j$ denote the compute value of p_j. By applying the fundamental model of floating point arithmetic, we find

$$\begin{aligned} \hat{p}_0 &= a_2, \\ \hat{p}_1 &= \hat{p}_0 x\langle 2\rangle + a_1\langle 1\rangle = a_2 x\langle 2\rangle + a_1\langle 1\rangle, \\ \hat{p}_2 &= \hat{p}_1 x\langle 2\rangle + a_0\langle 1\rangle = a_2 x^2\langle 4\rangle + a_1 x\langle 3\rangle + a_0\langle 1\rangle. \end{aligned}$$

It follows that the computed value $\hat{y}$ can be written as

$$\hat{y} = a_2 x^2\langle 4\rangle + a_1 x\langle 3\rangle + a_0\langle 1\rangle.$$

■

Example 5.3 Analyse the error of Horner's method for $n = 3$.

Solution We proceed as in the previous example. We find that

$$\begin{aligned}
\hat{p}_0 &= a_3 \\
\hat{p}_1 &= \hat{p}_0 x\langle 2\rangle + a_2\langle 1\rangle = a_3 x\langle 2\rangle + a_2\langle 1\rangle \\
\hat{p}_2 &= \hat{p}_1 x\langle 2\rangle + a_1\langle 1\rangle = a_3 x^2\langle 4\rangle + a_2 x\langle 3\rangle + a_1\langle 1\rangle \\
\hat{p}_3 &= \hat{p}_2 x\langle 2\rangle + a_0\langle 1\rangle = a_3 x^3\langle 6\rangle + a_2 x^2\langle 5\rangle + a_1\langle 3\rangle + a_0\langle 1\rangle.
\end{aligned}$$

It follows that the computed value can be written as

$$\hat{y} = a_3 x^3\langle 6\rangle + a_2 x^2\langle 5\rangle + a_1 x\langle 3\rangle + a_0\langle 1\rangle.$$

■

These two examples reveals the general pattern. The pattern can be maintained as long as the model of floating point arithmetic holds true and as long as the assumptions needed for Corollary 4.2 are satisfied. We have the following theorem.

Theorem 5.2 *Assume $2nu < 1$. If Algorithm 11 does not experience overflow/underflow, then the computed value $\hat{y}$ satisfies*

$$\hat{y} = a_n\langle 2n\rangle x^n + \sum_{j=0}^{n-1} a_j\langle 2j+1\rangle x^j \tag{5.1}$$

This representation of the computed value allows us to bound first the error and then the relative error. It is useful to introduce the polynomial $\tilde{p}$ given by

$$\tilde{p}(t) = \sum_{j=0}^{n} |a_j| t^j. \tag{5.2}$$

Theorem 5.3 *Assume $2nu < 1$. If Algorithm 11 does not experience overflow/underflow, then the error satifies*

$$|y - \hat{y}| \leq \gamma_{2n}\tilde{p}(|x|) \tag{5.3}$$

Proof. By Theorem 5.2 we know that the computed value returned by Horner's method can be written as

$$\hat{y} = \sum_{j=0}^{n} a_j x^j (1 + \theta_j^{(k_j)}), \tag{5.4}$$

where $|\theta_j^{(k)}| \leq \gamma_k$ and k_j satisfies $k_j \leq 2n$. It is clear that

$$\hat{y} = y + \sum_{j=0}^{n} a_j x^j \theta_j^{(k_j)}). \tag{5.5}$$

The triangle inequality now implies

$$|y - \hat{y}| \leq \sum_{j=0}^{n} |a_j||x|^j \left|\theta_j^{(k_j)}\right| \leq \gamma_{2n} \sum_{j=0}^{n} |a_j||x|^j = \gamma_{2n}\tilde{p}(|x|) \tag{5.6}$$

This completes the proof. ■

Theorem 5.4 *Assume $2nu < 1$. If Algorithm 11 does not experience overflow/underflow, then the relative error satisfies*

$$\frac{|y-\hat{y}|}{|y|} \leq \gamma_{2n} \frac{\tilde{p}(|x|)}{|p(x)|}. \tag{5.7}$$

It is important to recognize that $p(x)$ is not necessarily computed with a small relative error. The fundamental problem is that $\tilde{p}(|x|)$ and $p(|x|)$ can be quite different! However, if a_j and x all have the same sign, then $\tilde{p}(|x|) = p(|x|)$ and now Theorem 5.4 ensures that $p(x)$ is computed with a small relative error.

Exercises

1. Horner's method
2. Horner's method for the derivative
3. Horner's method with running error bound

5.3 Polynomial interpolation

Frequently, we are given a list of nodes x_i and function values $f(x_i)$ in the form of a table, i.e.,

x_0	x_1	x_2	$\ldots$	x_n
$f(x_0)$	$f(x_1)$	$f(x_2)$	$\ldots$	$f(x_n)$

and are we tasked with approximating $f(x)$ for a suitable range of x values.

In this section we will show how to compute a polynomial p of degree at most n which satisfies

$$p(x_i) = f(x_i), \quad i = 0, 1, 2, \ldots, n.$$

We say that p interpolates f at the nodes x_i. We will show that this polynomial is unique and we will derive a formula for the error given by

$$e(x) = f(x) - p(x).$$

We begin by investigating the problem in the case of $n = 1$.

Example 5.4 Let $x_0 \neq x_1$. Show that there exists a polynomial p of degree at most 1 such that

$$\begin{aligned} p(x_0) &= f(x_0) \\ p(x_1) &= f(x_1) \end{aligned}$$

Solution We begin by examining two very special polynomials l_0 and l_1 given by

$$\begin{aligned} l_0(x) &= \frac{x - x_1}{x_0 - x_1} \\ l_1(x) &= \frac{x - x_0}{x_1 - x_0} \end{aligned}$$

It is straigtforward to verify that

$$\begin{aligned} l_0(x_0) &= 1 \qquad l_0(x_1) = 0 \\ l_1(x_0) &= 0 \qquad l_1(x_1) = 1 \end{aligned}$$

It is clear that l_0 and l_1 are polynomials of degree 1. It follows that

$$p(x) = f(x_0)l_0(x) + f(x_1)l_1(x) \tag{5.8}$$

is a polynomial of degree (at most) 1. Moreover, we have

$$\begin{aligned} p(x_0) &= f(x_0)l_0(x_0) + f(x_1)l_1(x_0) = f(x_0) \cdot 1 + f(x_1) \cdot 0 = f(x_0) \\ p(x_1) &= f(x_0)l_0(x_1) + f(x_1)l_1(x_1) = f(x_0) \cdot 0 + f(x_1) \cdot 1 = f(x_1) \end{aligned}$$

■

Example 5.5 Show that the polynomial p given by Example 5.4 is unique.

Solution Suppose that q is a polynomial of degree at most 1 such that

$$\begin{aligned} q(x_0) &= f(x_0), \\ q(x_1) &= f(x_1). \end{aligned}$$

We claim that $p(x) = q(x)$ for all x where p is the polynomial given by Example 5.4. Let r denote the function given by $r(x) = p(x) - q(x)$. We claim that $r(x) = 0$ for all x. It is clear that r is a polynomial of degree at most 1. However, r has at least two zeros! Specifically, we have $r(x_i) = 0$ for $i = 0, 1$. Now, if r is a polynomial of degree 1, then r has exacly one zero. We conclude that r must have degree 0, i.e., r is a constant. It follows that $r(x) = 0$ for all x. ■

We now seek to extend Example 5.4 and Example 5.5 to the case of $n > 1$.

Definition 5.1 *The Lagrange basis polynomial $l_j : \mathbb{R} \to \mathbb{R}$ corresponding to the node x_j is given by*

$$l_j(x) = \prod_{i \neq j} \frac{x - x_i}{x_j - x_i}. \tag{5.9}$$

Example 5.6 Construct the Lagrange basis polynomials corresponding to each of the three nodes $x_0 = -1$, $x_1 = 0$, and $x_2 = 1$ and compute $l_j(x_i)$ explicitly for $i, j \in 0, 1, 2$.

Solution By definition, we have

$$\begin{aligned} l_0(x) &= \frac{(x - x_1)(x - x_2)}{(x_0 - x_1)(x_0 - x_2)} = \frac{x(x-1)}{2} \\ l_1(x) &= \frac{(x - x_0)(x - x_2)}{(x_1 - x_0)(x_1 - x_2)} = (x+1)(1-x) \\ l_2(x) &= \frac{(x - x_0)(x - x_1)}{(x_2 - x_0)(x_2 - x_1)} = \frac{(x+1)x}{2} \end{aligned}$$

It is easy to verify that

$$\begin{aligned} l_0(-1) &= \frac{(-1)(-1-1)}{2} = 1, & l_0(0) &= \frac{0 \cdot (0-1)}{2} = 0, & l_0(1) &= \frac{1(1-1)}{2} = 0, \\ l_1(-1) &= (-1+1)(1-(-1)) = 0, & l_1(0) &= (0+1)(1-0) = 1, & l_1(1) &= (1+1)(1-1) = 0, \\ l_2(-1) &= \frac{(-1+1) \cdot (-1)}{2} = 0, & l_2(0) &= \frac{(0+1) \cdot 0}{2} = 0, & l_2(1) &= \frac{(1+1) \cdot 1}{2} = 1. \end{aligned}$$

■

Lemma 5.1 *The Lagrange basis polynomial l_j corresponding to the node x_j satisfies*

$$l_j(x_i) = \delta_{ij}$$

where δ_{ij} is the Kronecker delta given by

$$\delta_{ij} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}.$$

Proof. Left to the reader as Exercise 1. ■

The Lagrange basis polynomials allows us to solve the interpolation problem. We have the following theorem.

Theorem 5.5 *Let $f : [a,b] \to \mathbb{R}$ be a continuous function, let $\{x_i\}_{i=0}^n \subset [a,b]$ denote a set of distinct nodes, and let l_j denote the Lagrange basis polynomial corresponding to the node x_j. The polynomial $p : \mathbb{R} \to \mathbb{R}$ given by*

$$p(x) = \sum_{j=0}^{n} f(x_j) l_j(x) \tag{5.10}$$

satisfies $p(x_i) = f(x_i)$ a for $i = 0, 1, 2, \ldots n$ and has degree at most n.

Proof. Every polynomial l_j has degree n. It follows that p has degree at most n. By Lemma 5.1, $l_j(x_i) = \delta_{ij}$. It follows that

$$p(x_i) = \sum_{j=0}^{n} f(x_j) l_j(x_i) = \sum_{j=0}^{n} f(x_j)\delta_{ij} = f(x_i).$$

This completes the proof. ■

Theorem 5.6 *The polynomial p given by Theorem 5.5 is unique. Specifically, if q is a polynomial which satisfies $q(x_i) = f(x_i)$ for $i = 0, 1, 2, \ldots, n$, then $p(x) = q(x)$ for all x*

Proof. We claim that $p(t) = q(t)$ for all $t \in \mathbb{R}$. To that end we study the properties of the function $r : \mathbb{R} \to \mathbb{R}$ given by $r(t) = p(t) - q(t)$. It is clear that r is a polynomial of degree at most n. Let $k \in \{0, 1, 2, \ldots, n\}$ denote the degree of r. If $k > 1$, then r has exactly k roots by the fundamental theorem of algebra. However, by construction $r(x_i) = 0$ for $i = 0, 1, 2 \ldots, n$. We conclude that r has at least $n + 1$ roots. It follows that $k = 0$ is the only option. Therefore r is a constant and $r(x) = 0$ is the only option. This completes the proof. ■

Theorem 5.6 and Theorem 5.6 is the theoretical foundation for the following definition.

Definition 5.2 *The polynomial p given by equation* (5.10) *is called the interpolating polynomial for f corresponding to the nodes x_i.*

The definition would be meaningless if p did not exist. The use of the definite article "the" as in "*the* interpolating polynomial" would be meaningless if p was not unique. The specific representation given by equation (5.10) is called the Lagrange form of the interpolating polynomial.

Example 5.7 Find the interpolating polynomial for $f(x) = \sin(x)$ corresponding to the nodes $\{x_j\}_{j=0}^4$ given by

$$x_j = -\pi + j\frac{\pi}{2}, \qquad j = 0, 1, 2, 3, 4.$$

Solution We observe that $f(x_j) = 0$ for $j = 0, 2, 4$. Therefore, the is no reason to compute l_0, l_2, l_4. We have

$$l_1(x) = \frac{(x - x_0)(x - x_2)(x - x_3)(x - x_4)}{(x_1 - x_0)(x_1 - x_2)(x_1 - x_3)(x_1 - x_4)} = -\frac{8(x + \pi)x(x - \frac{\pi}{2})(x - \pi)}{3\pi^4}$$

and

$$l_3(x) = \frac{(x-x_0)(x-x_1)(x-x_2)(x-x_4)}{(x_3-x_0)(x_3-x_1)(x_3-x_2)(x_3-x_4)} = -\frac{8(x+\pi)(x+\frac{\pi}{2})x(x-\pi)}{3\pi^4}.$$

Since $f(x_1) = -1$ and $f(x_3) = 1$ the interpolation polynomial p is given by

$$p(x) = l_3(x) - l_1(x) = \frac{8x(\pi^2 - x^2)}{3\pi^3}.$$

It is easy to verify that $p(-\pi) = p(0) = p(\pi) = 0$ as required. Moreover,

$$p(-\tfrac{\pi}{2}) = \frac{8(-\frac{\pi}{2})(\frac{3\pi^2}{4})}{3\pi^3} = -1$$

and

$$p(\tfrac{\pi}{2}) = \frac{8(\frac{\pi}{2})(\frac{3\pi^2}{4})}{3\pi^3} = 1.$$

We conclude, that p is *the* interpolating polynomial of f corresponding to the nodes x_j. We emphasize that p has degree 3 rather than 4. This is not a violation of Theorem 5.5. ■

By definition, the interpolating polynomial p satisfies $p(x_i) = f(x_i)$. We will now consider a very important question: Is $p(x)$ a good approximation of $f(x)$ when x is not a node?

Theorem 5.7 *Let $f \in C^{n+1}([a,b],\mathbb{R})$ and let $p : [a,b] \to \mathbb{R}$ be the interpolating polynomial of f corresponding to the $n+1$ distinct nodes $\{x_i\}_{i=0}^n \subset [a,b]$. Let $x \in [a,b]$. Then there exists ξ between $\min\{\min_i\{x_i\}, x\}$ and $\min\{\max_i\{x_i\}, x\}$ such that*

$$f(x) - p(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!}\omega(x) \tag{5.11}$$

where $\omega : [a,b] \to \mathbb{R}$ is the polynomial given by

$$\omega(x) = (x-x_0)(x-x_1)\dots(x-x_n) = \prod_{i=0}^{n}(x-x_i) \tag{5.12}$$

Proof. If $x \in \{x_i\}_{i=0}^n$, then there is nothing to show, because $f(x_i) - p(x_i) = 0$ by the definition of p and $\omega(x_i) = 0$ by the definition of ω. We therefore assume that $x \notin \{x_i\}_{i=0}^n$. We now introduce the auxilliary function $g : [a,b] \to \mathbb{R}$ given by

$$g(t) = f(t) - p(t) - \left[\frac{f(x) - p(x)}{\omega(x)}\right]\omega(t).$$

Notice, that we do not divide by zero, because $\omega(x) \neq 0$. We will now collect relevant information about g. It is clear that $g \in C^{n+1}$ because $f \in C^{n+1}$ and

$$g^{(n+1)}(t) = f^{(n+1)}(t) - \left[\frac{f(x) - p(x)}{\omega(x)}\right](n+1)!$$

because $p^{(n+1)}(t) = 0$ (p has degree at most n) and $\omega^{(n+1)}(t) = (n+1)!$. Moreover, $g(x_i) = 0$ and $g(x) = 0$. We conclude that g has at least $n+2$ zeros. By Rolle's theorem, g' has at least $n+1$ zeros $\min\{x_i, x\}$ and $\max\{x_i, x\}$. In fact, the repeated application of Rolle's theorem shows that $g^{(n+1)}$ has at least one zero ξ between $\min\{x_i, x\}$ and $\max\{x_i, x\}$. It follows, that

$$f^{(n+1)}(\xi) - \left[\frac{f(x) - p(x)}{\omega(x)}\right](n+1)! = 0$$

or equivalently

$$f(x) - p(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!}\omega(x).$$

This completes the proof. ■

In general, we want a small (relative) error. We can only guarantee that the error is small if the nodes are close and if do not venture too far away. In particular, if x is far from any node, then $\omega(x)$ will be large and we have no guarantee that the error will be small. Typically, the error will be large and our attempt of to extrapolate f using p will fail miserably. Extrapolation is possible, but requires an altogether different approach.

Theorem 5.7 is particularly useful when we have extra information available.

Corollary 5.1 *Let f, $\{x_i\}$ and p be as in Theorem 5.7. If*

$$|f^{(n+1)}(x)| \leq M_n$$

for all $x \in [a, b]$, then

$$|f(x) - p(x)| \leq \frac{M_n}{(n+1)!} h^{n+1}$$

where $h = b - a$ denotes the length of the interval.

Proof. The proof is left to reader as Exercise 2. ■

The following example considers the case where it is easy to bound the derivatives of f.

Example 5.8 Let $f : [-\pi, \pi] :\rightarrow \mathbb{R}$ be given by $f(x) = \sin(x)$ and let $p : [-\pi, \pi] :\rightarrow \mathbb{R}$ be the polynomial which interpolates f at the 5 nodes $x_j = -\pi + \frac{j}{2}\pi$ for $j = 0, 1, 2, 3, 4$. Show that the error $e(x) = f(x) - p(x)$ satisfies

$$|e(x)| \lesssim 0.2894.$$

Solution We have $f \in C^\infty$, so $f \in C^5$ and Theorem 5.7 applies with $n = 4$. It is clear that $|f^{(5)}(x)| \leq 1$. Therefore

$$|e(x)| \leq \frac{1}{5!}|\omega(x)|$$

where

$$\omega(x) = \prod_{j=0}^{4}(x - x_j) = (x - \pi)(x - \tfrac{\pi}{2})x(x + \tfrac{\pi}{2})(x + \pi) = x(x^2 - \pi^2)(x^2 - \tfrac{\pi^2}{4})$$

We now seek to establish an upper bound of $|\omega(x)|$. To that end, we determine the range of ω using standard techniques. We first examine the endpoint of the interval. We have

$$\omega(-\pi) = \omega(\pi) = 0.$$

We then find the stationary points of ω by solving the equation $\omega'(x) = 0$. Using the product rule of differentiation we obtain

$$\omega'(x) = (x^2 - \pi^2)(x^2 - \tfrac{\pi^2}{4}) + 2x^2(x^2 - \tfrac{\pi^2}{4}) + 2x^2(x^2 - \pi^2)$$

We observe that ω' can be written as

$$\omega'(x) = q(x^2)$$

where

$$q(t) = 5t^2 - \tfrac{15}{4}\pi^2 t + \tfrac{\pi^4}{4}$$

This simplifies the problem of solving the equation $\omega'(x) = 0$. It is straight forward to verify that

$$q(t) = 0$$

if and only if

$$t = \frac{15 \pm \sqrt{145}}{40}\pi^2$$

The four zeros ξ_j of ω' are therefore given by

$$\xi_1 = -\sqrt{\frac{15+\sqrt{145}}{40}}\pi \qquad \xi_2 = -\sqrt{\frac{15-\sqrt{145}}{40}}\pi \tag{5.13}$$

$$\xi_3 = -\xi_2 \qquad \xi_4 = -\xi_1 \tag{5.14}$$

A simple calculation establishes that

$$\begin{array}{ll} \omega(\xi_1) \approx 34.7278 & \omega(\xi_2) \approx -13.5672 \\ \omega(\xi_3) = -\omega(\xi_2) & \omega(\xi_4) = -\omega(\xi_1). \end{array}$$

We can now conclude that

$$-\omega(\xi_1) - \leq \omega(x) \leq \omega(\xi_1).$$

for all $x \in [-\pi, \pi]$. It follows that

$$|\omega(x)| \leq |\omega(\xi_1)| \approx 34.7278$$

We can finally estimate

$$|e(x)| \leq \frac{|\omega(\xi_1)|}{120} \approx 0.2894.$$

■

Exercises

1. Prove Lemma 5.1.
2. Prove Corollary 5.1.

Computer problems

5.4 Spline interpolation

Motivation: differentiable interpolation in tables Cubic spline interpolation Algorithm Examples

5.5 Uniform approximation theory

Theorem 5.8 **Characterization of best uniform approximation**

Example 5.9 Find the best uniform approximation of the square root on $[1, 2]$.

Example 5.10 Find the best uniform approximation of the reciprocal function on $[1, 2]$

Definition 5.3 **Bernstein polynomials**

Theorem 5.9 **Stone-Weierstrass**

Chapter 6

Non-linear equations

Contents

6.1 Introduction

In this chapter we consider the problem of finding roots of equations (or zeros of functions). More formally, let $I \subseteq \mathbb{R}$ and let $f : I \to \mathbb{R}$ and consider the equation

$$f(x) = 0.$$

A root $\alpha \in I$ is a solution, i.e., $f(r) = 0$. In general, it is impossible to find a simple formula for the roots of a non-linear equation. Instead we use an iterative method to produce a sequence $\{x_n\}_{n=0}^{\infty}$ of approximations such that

$$x_n \to \alpha, \quad n \to \infty,$$

where α is some root of the equation.

Example 6.1 Solving the non-linear equation

$$f(x) = x^2 - \alpha = 0 \tag{6.1}$$

with respect to $x > 0$.

Solution Equation 6.1 has a unique positive solution namely the square root $\sqrt{\alpha}$. In general, square roots can be computed using the Babylonian method. This is the iterative method

$$x_{n+1} = \frac{1}{2}\left(x_n + \frac{\alpha}{x_n}\right), \quad n = 0, 1, 2, \ldots,$$

where $x_0 > 0$ is an initial approximation of $\sqrt{\alpha}$. Figure 6.1 shows the result of applying the Babylonian method to the case of $\alpha = 2$ with $x_0 = 1.5$. We see that x_n converges rapidly

n	x_n
1	1.500000000000000
2	1.416666666666667
3	1.414215686274510
4	1.414213562374690
5	1.414213562373095

Figure 6.1: The results of applying the Babylonian method to the problem of computing $\sqrt{2}$ using $x_0 = 1.5$ as the initial approximation.

towards $\sqrt{2}$. In fact, convergence is so rapid that the number of correct significant figures is doubling from one iteration to the next, until we run out of figures.

■

This simple experiment raises several interesting questions which will be addressed in the following sections.

1. How do we construct an iteration which converges to a root α of a specific nonlinear iteration?
2. How do we construct the initial guess $x_0 \approx \alpha$
3. When should we terminate the iteration?
4. How do we measure the convergence and efficiency of our method?
5. What are the consequences of using finite precision arithmetic?

6.2 Bisection

The bisection algorithm hinges on the following fundamental theorem.

Theorem 6.1 **Intermediate-Value Theorem for Continuous functions** *Let $f : [a, b] \to \mathbb{R}$ be continuous. If $f(a) < m < f(b)$, then there exists c such that $a < c < b$ and $m = f(c)$.*

In particular, if $f(a)$ and $f(b)$ have different sign, say, $f(a) < 0$ and $f(b) > 0$, then f has a zero r in the interval (a, b). The best estimate of r is the midpoint of the interval, i.e. $c = (a+b)/2$. If $f(c) \neq 0$, then we consider the sign of $f(c)$. If $f(a)$ and $f(c)$ have different sign, then there is a root in the subinterval (a, c). If $f(c)$ and $f(b)$ have different sign, then there is a root in the subinterval (b, c). In either case, the length of the subinterval is one half the length of the length of (a, b). The entire proceduce is formalized as Algorithm 12.

Theorem 6.2 *Let $[a_0, b_0], [a_1, b_1], \dots, [a_n, b_n]$, denote the brackets generated by the bisection algorithm when applied to the equation $f(x) = 0$. Then the limits $\lim\limits_{n\to\infty} a_n$ and $\lim\limits_{n\to\infty} b_n$ exists, are equal and represent a zero r of f. Moreover,*

$$|r - c_n| \leq 2^{-n+1}(b_0 - a_0)$$

Proof. The sequence $\{a_n\}_{n=0}$ is increasing and bounded from above by b_0. It follows that

$$a = \lim_{n\to\infty} a_n$$

Algorithm 12 Basic bisection algorithm

```
for $j = 0, 1, 2, \dots, n-1$ do
    $c_j \leftarrow (a_j + b_j)/2$
    $f_{c_j} \leftarrow f(c_j)$
    if $y_j \neq 0$ then
        if $f(a_j)f(c_j) < 0$ then
            $a_{j+1} \leftarrow a_j$
            $b_{j+1} \leftarrow c_j$
        else
            $a_{j+1} \leftarrow c_j$
            $b_{j+1} \leftarrow b_j$
        end if
    else
        return
    end if
end for
```

exists. The sequence $\{b_n\}_{n=0}^{\infty}$ is decreasing and bounded from below by a_0. It follows that

$$b = \lim_{n \to \infty} b_n$$

exists. Since $a_n \leq b_n$ for all n, we have $a \leq b$. By construction,

$$b_{n+1} - a_{n+1} = \frac{1}{2}(b_n - a_n)$$

It follows by induction, that

$$b_n - a_n = 2^{-n}(b_0 - a_0).$$

This implies that $b_n - a_n \to 0$ as $n \to \infty$. We conclude that $a = b$. Now, let r denote the common value. By design, $f(a_n)$ and $f(b_n)$ have opposite sign, so $f(a_n)f(b_n) \leq 0$. This implies that

$$f(r)^2 = f(r)f(r) = \left(\lim_{n\to\infty} f(a_n)\right)\left(\lim_{n\to\infty} f(b_n)\right) = \lim_{n\to\infty} f(a)f(b_n) \leq 0$$

We conclude that $f(r) = 0$. Finally, we observe that $r \in [a_n, b_n]$. This implies that

$$|r - c_n| \leq \frac{1}{2}(b_n - a_n) \leq 2^{-(n+1)}(b_0 - a_0).$$

■

Algorithm 12 and the error analysis contained in Theorem 6.2 assumes that the calculations are done in exact arithmetic. In particular, a computer implementation of Algorithm 12 will not work as intented:

1. It is entirely possible that the compute value $\hat{c}$ of $c = (a+b)/2$ lies outside the interval (a, b). Therefore c should be computed using the formula

   $$c = a + \frac{b-a}{2}.$$

2. Rounding errors will typically prevents us from computing $f_c = f(c)$ exactly. It is therefore meaningless to compare with zero the computed value $\hat{f}_c$ of f_c. Instead we should terminate the search if

   $$|\hat{f}_c| < \epsilon,$$

   where $\epsilon > 0$ is a tolerance specified by the user.

3. We should also terminate the search if the length of the current search interval is sufficiently small, i.e., if
$$|b - a| < \delta,$$
where $\delta > 0$ is a tolearance specified by the user.

Algorithm 13 includes these modifications. A computer implementation will function as intended, provided the sign of f is computed correctly.

Algorithm 13 Practical bisection algorithm

```
$a \leftarrow \min\{a_0, b_0\}$
$b \leftarrow \max\{a_0, b_0\}$
$f_a \leftarrow f(a)$
$f_b \leftarrow f(b)$
for $j = 0, 1, 2, \dots, n-1$ do
  $c \leftarrow a + \frac{b-a}{2}$
  $f_c \leftarrow f(c)$
  if $|b - a| < \delta$ then
    return
  else
    if $|f_c| < \epsilon$ then
      return
    else
      if $\text{sign}(f_a) \cdot \text{sign}(f_c) < 0$ then
        $b \leftarrow c$
        $f_b \leftarrow f_c$
      else
        $a \leftarrow c$
        $f_a \leftarrow f_c$
      end if
    end if
  end if
end for
```

6.3 Newton's method

Newton's method is given by
$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \tag{6.2}$$

Example 6.2 The number $\sqrt{\alpha}$ is the unique positive solution of the nonlinear equation
$$f(x) = x^2 - \alpha = 0$$
Analyse the convergence of Newton's method for this equation.

Solution Newton's method takes the form
$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} = x_n - \frac{x_n^2 - \alpha}{2x_n} = \frac{x_n^2 + \alpha}{2x_n} = g(x_n)$$
where $g : (0, \infty) \to \mathbb{R}$ is given by
$$g(x) = \frac{1}{2}\left(x + \frac{\alpha}{x}\right).$$

It is clear that the natural choice of $x_0 > 0$ implies that $x_n > 0$ for all n. We have

$$e_{n+1} = \sqrt{\alpha} - x_{n+1} = -\frac{x_n^2 - 2\sqrt{\alpha}x_n + \alpha}{2x_n} = -\frac{(x_n - \sqrt{\alpha})^2}{2x_n} = -\frac{e_n^2}{2x_n}.$$

An elementary analysis reveals that

$$g(x) \geq \sqrt{\alpha}, \quad x > 0.$$

It follows that

$$x_n \geq \sqrt{\alpha}, \quad n \geq 1,$$

regardless of the choice of $x_0 > 0$. We can therefore assume that $x_n \geq \sqrt{\alpha}$. If necessary, we simply discard x_0 and renumber, i.e., $x_n' = x_{n+1}$ for $n \geq 0$. Now let

$$r_n = \frac{e_n}{\sqrt{\alpha}}$$

denote the relative error. Then

$$r_{n+1} = -\frac{\sqrt{\alpha}}{2x_n}\frac{r_n^2}{2}$$

and

$$|r_{n+1}| \leq \frac{1}{2}r_n^2.$$

Let $y_n = \frac{1}{2}|r_n|$, then

$$y_{n+1} \leq y_n^2$$

By induction on n we discover, that

$$y_n \leq (y_0)^{2^n}$$

or equivalently

$$|r_n| \leq 2\left(\frac{|r_0|}{2}\right)^{2^n}.$$

We conclude that Newton's method converges if that the initial relative error is less than 2. ■

The following example discusses the possibility of constructing a good guess for the square root of a real number $\alpha \in [1, 4]$.

Example 6.3 Let $I = [1, 4]$ and let $x_0 : I \to \mathbb{R}$ be given by

$$x_0(s) = \frac{1}{3}s + \frac{17}{24}$$

Let $a \in I$ be given and define

$$r_0 = \frac{\sqrt{a} - x_0(a)}{\sqrt{a}}.$$

Show that

$$|r_0| \leq \frac{1}{24}.$$

Solution Let $h : [1, 4] \to \mathbb{R}$ be given by

$$h(s) = \sqrt{s} - \left(\frac{1}{3}s + \frac{17}{24}\right)$$

An elementary analysis reveals that

$$\forall\, s \in [1, 4] \,:\, s - \frac{1}{24} \leq h(s) \leq \frac{1}{24}.$$

Since $1 \leq \sqrt{a}$, the result

$$|r_0| \leq \frac{1}{24}$$

is immediate. ■

The following example shows that is possible to use Newton's method to compute the reciprocal of a number $\alpha \neq 0$, i.e., the number α^{-1}.

Example 6.4 Let $\alpha > 0$. The number $\frac{1}{\alpha}$ is the unique solution of the equation

$$f(x) = \frac{1}{x} - \alpha = 0$$

Show that Newton's method does not require any divisions and analyze the convergence.

Solution Newton's method can be written as

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

This representation requires no divisions. Now let r_n be given by

$$r_n = 1 - \alpha x_n.$$

It is clear that $r_n \to 0$ as $n \to \infty$ if and only if $x_n \to \alpha^{-1}$ as $n \to \infty$. We have

$$r_{n+1} = 1 - \alpha x_{n+1} = 1 - \alpha(x_n + x_n(1 - \alpha x_n)) = (1 - \alpha x_n)^2$$

It follows by induction that

$$|r_n| \leq |r_0|^{2^n}.$$

We conclude that Newton's method converges rapidly if $|r_0| < 1$. ■

Lemma 6.1 **The error formula for Newton's method** *Let $f \in C^2(\mathbb{R}, \mathbb{R})$. There exists a ξ_n between r and x_n such that*

$$e_{n+1} = -\frac{1}{2}\frac{f''(\xi_n)}{f'(x_n)} e_n^2$$

Proof. By Taylor's formula there exist a ξ_n between r and x_n such that

$$f(r) = f(x_n) + f'(x_n)(r - x_n) + \frac{1}{2}f''(\xi_n)(r - x_n)^2 = f(x_n) + f'(x_n)e_n + \frac{1}{2}f''(\xi_n)e_n^2$$

Since $f(r) = 0$ we can write

$$f(x_n) = -\left(f'(x_n)e_n + \frac{1}{2}f''(\xi_n)e_n^2\right)$$

It follows

$$e_{n+1} = r - x_{n+1} = r - x_n + \frac{f(x_n)}{f'(x_n)} = e_n - \frac{f'(x_n)e_n + \frac{1}{2}f''(\xi_n)e_n^2}{f'(x_n)} = \frac{1}{2}\frac{f''(\xi_n)}{f'(x_n)} e_n^2$$

■

The error formula can be used to prove global convergence of Newton's method when certain specific conditions are satified. We have the following theorem.

Theorem 6.3 **Global convergence of Newton's method** *Let $f \in C^2(\mathbb{R}, \mathbb{R})$ have a zero $r \in \mathbb{R}$. Assume that $f'(x) > 0$ and $f''(x) > 0$, then the zero r is unique and Newton's method converges to r for all $x_0 \in \mathbb{R}$.*

Proof. Since f is strictly increasing it is clear that f has at most one zero. Now let $x_0 \in \mathbb{R}$ be given. Then by Lemma 6.1 there exists at least one ξ between x_0 and r such that

$$e_1 = -\frac{1}{2}\frac{f''(\xi_0)}{f'(x_0)}e_0^2$$

We conclude that $e_0 \neq 0$, then $e_1 < 0$. It follows inductively, that $e_n \leq 0$ for all $n \geq 1$. By disregarding the initial guess and renumbering, i.e., $x_n' = x_{n+1}$ we can assume that $e_n = r - x_n \leq 0$ for all n. Equivalently, we have

$$r \leq x_n$$

Since f is increasing we deduce that

$$0 \leq f(x_n)$$

and

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

In summary, the sequence $\{x_n\}_{n=0}^{\infty}$ is decreasing and bounded from below. It follows that there exists a $t \in \mathbb{R}$ such that $x_n \to t$ as $n \to \mathbb{N}$. It follows by the continuity of f and f', that

$$x_{n+1} \to t - \frac{f(t)}{f'(t)}$$

From this we deduce $f(t) = 0$. Since f has only one zero, we have $t = r$. This completes the proof. ■

Example 6.5 Let $\alpha > 0$. The number $\log(\alpha)$ is the unique solution of the non-linear equation

$$f(x) = \exp(x) - \alpha = 0.$$

Show that Newton's method for this equation converges to $r = \log(\alpha)$ for all $x_0 \in \mathbb{R}$

Solution It is clear that $f \in C^{(2)}(\mathbb{R},\mathbb{R})$. Moreover, $f'(x) = f''(x) = \exp(x) > 0$. By Theorem 6.3 it follows that Newton's method converges to $r = \log(\alpha)$ for all $x_0 \in \mathbb{R}$. However if x_0 is far from α, then there is no guarantee that the convergence will be rapid. Figure 6.2 considers the case of $\alpha = 2$ and $x_0 = 20$. ■

In general, we can not expect Newton's method to converge to a root. However, the following theorem shows that Newton's method converges provided that the initial guess x_0 lies sufficiently close to a root r.

Theorem 6.4 **Local convergence of Newton's method** *Let I be an open interval and let $f \in C^{(2)}(I, R)$ and let $r \in I$ be a zero of f. Then there exists $\delta > 0$ such that $J = [r-\delta, r+\delta] \subset I$ and Newton's method is defined for all $x_0 \in J$ and $x_n \to r$ for $n \to \infty$.*

Proof. This is a direct application of Theorem 6.7 and Theorem 6.5. ■

6.4 Secant method

Iteration

$$x_{n+1} = x_n - f(x_n)\left(\frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}\right)^{-1}. \tag{6.3}$$

Using divided differences, we can express this as

$$x_{n+1} = x_n - \frac{f(x_n)}{f[x_{n-1}, x_n]} \tag{6.4}$$

```
         n                       x(n)                  f(x(n))
--------------------------------------------------------------
         0      2.000000000000000e+01    4.851651934097903e+08
         1      1.900000000412231e+01    1.784822996989459e+08
         2      1.800000001532790e+01    6.565996814375989e+07
         3      1.700000004578786e+01    2.415495185957885e+07
         4      1.600000012858661e+01    8.886109663142741e+06
         5      1.500000035365693e+01    3.269016528582950e+06
         6      1.400000096546135e+01    1.202603445233293e+06
         7      1.300000262851718e+01    4.424125549016519e+05
         8      1.200000714916411e+01    1.627539549838774e+05
         9      1.100001943750097e+01    5.987330553019620e+04
        10      1.000005284025328e+01    2.202562970958861e+04
        11      9.000143635315041e+00    8.102247900179910e+03
        12      8.000390419473748e+00    2.980122038309648e+03
        13      7.001061082838394e+00    1.095797394619168e+03
        14      6.002882912631216e+00    4.025935215552242e+02
        15      5.007826145514215e+00    1.475792190075677e+02
        16      4.021196986820044e+00    5.376781923590243e+01
        17      3.057059963486566e+00    1.926494545068728e+01
        18      2.151111462684527e+00    6.594405451552822e+00
        19      1.383820986487438e+00    1.990118726553189e+00
        20      8.850592044608854e-01    4.231278472167492e-01
        21      7.104386834253313e-01    3.488373264164979e-02
        22      6.932958206258119e-01    2.973022266967718e-04
        23      6.931471916063326e-01    2.209277472076110e-08
        24      6.931471805599454e-01                        0
```

Figure 6.2: The result of applying Newton's method to the equation $f(x) = \exp(x) - 2$ using $x_0 = 20$.

Lemma 6.2 *The error $e_n = r - x_n$ of the secant method satisfies the following expression*

$$e_{n+1} = -e_n e_{n-1} \frac{f[x_{n-1}, x_n, r]}{f[x_{n-1}, x_n]} \tag{6.5}$$

Proof. The proof consists of short sequence of elementary transformations. Since $f(r) = 0$ we have

$$e_{n+1} = r - x_{n+1} = r - \left(x_n - \frac{f(x_n)}{f[x_{n-1}, x_n]} \right) = r - x_n - \frac{f(r) - f(x_n)}{f[x_{n-1}, x_n]}$$

We now write

$$f(r) - f(x_n) = \frac{f(r) - f(x_n)}{r - x_n}(r - x_n) = f[x_n, r](r - x_n)$$

and deduce

$$e_{n+1} = -(r - x_n)\left(\frac{f[x_n, r]}{f[x_{n-1}, x_n]} - 1 \right) = -(r - x_n)\frac{f[x_n, r] - f[x_{n-1}, x_n]}{f[x_{n-1}, x_n]}$$

We now write

$$f[x_n, r] - f[x_{n-1}, x_n] = \frac{f[x_n, r] - f[x_{n-1}, x_n]}{r - x_{n-1}}(r - x_{n-1}$$

and deduce

$$e_{n+1} = -(r - x_n)(r - x_{n-1})\frac{f[x_{n-1}, x_n, r]}{f[x_{n-1}, x_n]} = -e_n e_{n-1}\frac{f[x_{n-1}, x_n, r]}{f[x_{n-1}, x_n]}.$$

This completes the proof. ■

In the case of convergence and $f'(r) \neq 0$ we have

$$\frac{f[x_{n-1}, x_n, r]}{f[x_{n-1}, x_n]} \to \frac{1}{2}\frac{f''(r)}{f'(r)}, \quad n \to \infty.$$

It follows that

$$e_{n+1} \approx -e_{n-1}e_n \frac{1}{2}\frac{f''(r)}{f'(r)}.$$

6.5 Fixed points and functional iteration

Many equations can solved using a functional iteration, i.e., a sequence $\{x_n\}_{n=1}^{\infty}$ of the form

$$x_{n+1} = g(x_n), \quad n \geq 0.$$

Newton's method for the equation

$$f(x) = 0,$$

is a special example of a functional iteration. Here the function g is given by

$$g(x) = x - \frac{f(x)}{f'(x)}.$$

We are primarily interested in sequences $\{x_n\}_{n=1}^{\infty}$ which are convergent. If

$$x_n \to x, \quad n \to \infty, \quad n \in \mathbb{N},$$

and if g is continuous, then

$$g(x_n) \to g(x), \quad n \to \infty, \quad n \in \mathbb{N}.$$

We conclude that

$$g(x) = x.$$

This motivates the following definition.

Definition 6.1 *Let $I \subseteq \mathbb{R}$ and let $g : I \to I$. If $x \in I$ satisfies*

$$x = g(x)$$

then x is called a fixed point of g.

For this reason functional iterations are also known as fixed point iterations. Our main theorems results concern *contractive mappings* which we now define.

Definition 6.2 **Contractive mapping** *Let $I \subseteq \mathbb{R}$ be an interval. A function $g : I \to \mathbb{R}$ is a contractive mapping, if there exists $L < 1$, such that*

$$\forall x, y \; : \; |g(x) - g(y)| \leq L|x - y|.$$

Example 6.6 Let $I = [0, 1]$ and let $g : I \to \mathbb{R}$ be given by

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

Show that g is a contractive mapping.

Solution Let x_1 and x_2 be any pair of numbers in I. By the mean value theorem there exists at least one ξ between x_1 and x_2 such that

$$g(x_1) - g(x_2) = g'(\xi)(x_1 - x_2)$$

We now see to bound the absolute value of $g'(\xi)$ independently of ξ. We have $g'(t) = \sin(t)$ and since $t \to \sin(t)$ is nonnegative and monotone increasing on $[0, 1]$ we have

$$\forall t \in [0, 1] \; : \; |g'(t)| \leq \sin(1)$$

It follows that

$$|g(x_1) - g(x_2)| \leq L|x_1 - x_2|,$$

where

$$L = \sin(1) \approx 0.8415 < 1.$$

We conclude that $g : I \to \mathbb{R}$ is a contractive mapping of I into $\mathbb{R}$. ■

Theorem 6.5 **Contractive mapping theorem** *Let $I \subseteq \mathbb{R}$ be a closed interval and let $g : I \to I$ be a contractive mapping of I into I. Then g has exactly one fixed point a. Moreover the functional iteration*

$$x_{n+1} = g(x_n), \quad n = 0, 1, 2, \ldots$$

converges to a for any initial value $x_0 \in I$.

Proof. We begin by showing that the functional iteration converges to a point $a \in I$. To that end, we consider the infinite sequence

$$s = \sum_{j=0}^{\infty} (x_{j+1} - x_j).$$

The partial sums are given by

$$s_n = \sum_{j=0}^{n-1} (x_{j+1} - x_j) = x_n - x_0, \quad n \geq 1.$$

It follows, that the series s is convergent if and only if the sequence $\{x_n\}$ is convergent. We will now show that the series s is absolutely convergent. Let $L < 1$ be such that

$$\forall x, y \in I \; : \; |g(x) - g(y)| \leq L|x - y|$$

Then we have the estimate

$$|s_n| \leq \sum_{j=0}^{n-1} L^j |x_1 - x_0| = \frac{1 - L^n}{1 - L} |x_1 - x_0| \leq \frac{1}{1 - L} |x_1 - x_0|$$

It follows that s is absolutely convengent, hence convergent. This shows that the functional iteration converges to a point $t \in \mathbb{R}$. Since $x_n \in I$ and I is a closed interval we have $t \in I$. The function g is continuous, because it is a contractive mapping. This implies that t is a fixed point of g.

We will now show that g can not have *two* distinct fixed points. Suppose, that x and y are both fixed points of g. We will show that $x = y$. By assumption $x = g(x)$ and $y = g(y)$. It follows, that

$$|x - y| = |g(x) - g(y)| \leq L|x - y|$$

or equivalently

$$(1 - L)|x - y| \leq 0$$

Since $L < 1$, we conclude

$$|x - y| \leq 0.$$

Since $|x - y|$ is always non-negative, then only possibility is $x = y$. This completes the proof. ■

Example 6.7 Consider the functional iteration $x_{n+1} = g(x_n)$, where $g : \mathbb{R} \to \mathbb{R}$ is given by

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

Show that g has exactly one fixed point $t \in \mathbb{R}$ and $t \in [0, 1]$. Show that the functional iteration converges to t for all $x_0 \in \mathbb{R}$.

Solution We begin by observing that the function g maps $\mathbb{R}$ into $[-1, 1]$ and $[-1, 1]$ into $[0, 1]$. Therefore, if $t \in \mathbb{R}$ is any fixed point of g, then $t = g(t) = g(g(t))$ and t must lie in the interval $I = [0, 1]$. Now consider the restriction $g\big|_I : I \to I$ of g to the interval I. Example 6.6 shows that $g\big|_I$ is a contractive mapping of I into I. Theorem 6.5 can now be applied to $g\big|_I : I \to I$. We conclude that the functional iteration converges to the unique solution of the equation

$$t = \cos(t).$$

regardless of the choice of $x_0 \in \mathbb{R}$. ■

Theorem 6.6 *If $L < 1$ is such that*

$$\forall x, y \in I \ : \ |x - y| \leq L|x - y|$$

then

$$|a - x_n| \leq L^{n-1}|x_1 - x_0|, \quad n \in \mathbb{N}.$$

Proof. Let $V \subseteq \mathbb{N}$ be given by

$$V = \{n \in \mathbb{N} \ : \ |a - x_n| \leq L^{n-1}|x_1 - x_0|\}$$

We will now prove that $V = \mathbb{N}$. It is clear that $1 \in V$. Now suppose that $k \in V$, i.e. $|a - x_k| \leq L^{k-1}|x_1 - x_0$. We will now prove that $k + 1 \in V$. We have

$$|a - x_{k+1}| = |g(a) - g(x_k)| \leq L|a - x_k| \leq L^k|a - x_0|.$$

It follows that $k + 1 \in V$. By the principle of mathematical induction, $V = \mathbb{N}$. ■

Theorem 6.7 *Let $I \in \mathbb{R}$ be an open interval and let $g \in C^{(1)}(I, \mathbb{R})$. Let a denote a fixed point of g and assume that $|g'(a)| < 1$. Then there exists $\rho > 0$ such that $J = [a - \delta, a + \delta] \subset I$ and g is a contractive mapping of J into J.*

Proof. Since I is an open interval and $a \in I$, there exist $\delta_1 > 0$ such that $(a - \delta_1, a + \delta_1) \subseteq I$. Let $\epsilon = 1 - |g'(a)| > 0$. By the continuity of g', there exists a $\delta_2 > 0$, such that

$$\forall x \in I \ : \ |x - a| < \delta_2 \ \Rightarrow \ |g'(x) - g'(a)| < \epsilon.$$

Set

$$\delta = \frac{1}{2}\min\{\delta_1, \delta_2\}$$

and let

$$J = [a - \delta, a + \delta].$$

Then $J \subset I$. We claim that g is a contractive mapping of J into $\mathbb{R}$. Let x and y be elements of J, then there exists at least one ξ between x and y, such that

$$|g(x) - g(y)| = |g'(\xi)||x - y|.$$

If t is *any* element of J, then $|t - a| < \delta_2$ and

$$|g'(t)| \leq |g'(t) - g'(a)| + |g'(a)| < 1.$$

Since J is a closed and bounded interval and g' is continuous, there exists $L < 1$, such that

$$\forall t \in J \;:\; |g'(t)| \leq L.$$

This shows that $g : J \to \mathbb{R}$ is a contractive mapping. We will now show that g maps J into J. Let $x \in J$. Since $a \in J$ and $g(a) = a$ we have

$$|g(x) - a| = |g(x) - g(a)| \leq L|x - a| < |x - a| \leq \delta.$$

This shows that $g(x) \in J$. We conclude that g maps J into J. ■

Theorem 6.8 *Let $I \subseteq \mathbb{R}$ be a closed interval and let $g \in C^{(p)}(I, I)$ be a contractive mapping of I into I. Let $a \in I$ denote the unique fixed point of g. If*

$$g^{(k)}(a) = 0, \quad k = 1, 2, \ldots, p-1, \quad g^{(p)}(a) \neq 0,$$

then the functional iteration

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

with $x_0 \in I$ and $x_0 \neq a$ satisfies

$$\frac{x_{n+1} - a}{(x_n - a)^p} \to \frac{1}{p!} g^{(p)}(a), \quad n \to \infty, \quad n \in \mathbb{N}.$$

Proof. Let $x \in I$. Then by Taylor's formula, there exists at least one ξ between x and a such that

$$g(x) - g(a) = \frac{1}{p!} g^{(p)}(\xi)(x - a)^p.$$

In particular, there exists a sequence $\{\xi_n\}_{n=0}^{\infty}$ such that

$$g(x_n) - g(a) = x_{n+1} - a = \frac{1}{p!} g^{(p)}(\xi_n)(x_n - a)^p.$$

and ξ_n is between x_n and a. We have $x_n \to a$ as $n \to 0$ which ensure that $\xi_n \to a$ as $n \to \infty$. Moreover, the fact that $x_0 \neq a$ ensures that $x_n \neq a$ for all n. It follows

$$\frac{x_{n+1} - a}{(x_n - a)^p} = \frac{1}{p!} g^{(p)}(\xi_n) \to \frac{1}{p!} g^{(p)}(a), \quad n \to \infty.$$

■

Example 6.8 Let $g : \mathbb{R} \to \mathbb{R}$ be given by

$$g(x) = x + \sin(x).$$

Analyse the convergence of the functional iteration

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

where $x_0 \approx \pi$.

Solution We see that π is a fixed point of g, i.e. $\pi = g(\pi)$. We have

$$g'(x) = 1 + \cos(x), \quad g'(\pi) = 0.$$

By Theorem 6.7 there exists a closed interval J containing π such that $g\big|_J$ is a contractive mapping of J into J. By Theorem 6.5 the functional iteration converges to π.

It is clear that g is infinitely often differentiable. We have

$$g''(x) = -\sin(x), \quad g^{(3)}(x) = -\cos(x).$$

n	x_n
0	3.000000000000000
1	3.141120008059867
2	3.141592653572196
3	3.141592653589793

Figure 6.3: The functional iteration $x_{n+1} = x_n + \sin(x_n)$ converges rapidly to π even with $x_0 = 3$.

Therefore

$$g''(\pi) = 0, \quad g^{(3)}(\pi) = -\cos(\pi) = 1 \neq 0.$$

Then by Theorem 6.8

$$\frac{\pi - x_{n+1}}{(\pi - x_n)^3} \to \frac{1}{6}, \quad n \to \infty, \quad n \in \mathbb{N}.$$

Figure 6.3 ilustrates the rapid convergence this functional iteration. ■

In practice, it is impossible to compute g exactly. We now explore consequences of rounding error during the computation of g. We will assume that g is always computed with forward relative error which can be bounded uniformly. Specifically, we assume that computed value $\hat{y}$ of $y = g(x)$ can be written as

$$\hat{y} = g(x)(1 + \theta_x), \quad |\theta_x| \leq \Theta.$$

This is not an unreasonable assumption, because the functional relation does not stray far from the initial value and we only have to compute g accurately on a small interval, rather than all machine numbers. Moreover, we will assume that g is uniformly bounded, i.e., there exists a constant M such that

$$|g(x)| \leq M$$

for all relevant x.

We have the following theorem which relates the exact values x_n to the computed values $\hat{x}_n$.

Theorem 6.9 *Let g be a constraction and let $L \in (0, 1)$ satisfy*

$$\forall x, y \ : |g(x) - g(y)| \leq L|x - y|.$$

Let M be a number such that g is bounded by M,

$$\forall x : |g(x)| \leq M \tag{6.6}$$

and let Θ be a number such that the forward relative error is bounded by Θ, i.e.,

$$\hat{y} = g(x)(1 + \theta_x), \quad |\theta_x| \leq \Theta, \tag{6.7}$$

then the computed value $\hat{x}_n$ of x_n satisfies

$$|x_n - \hat{x}_n| \leq L^n |x_n - \hat{x}_0| + \frac{1 - L^n}{1 - L} M\Theta$$

Proof. The exact value x_{j+1} is given by

$$x_{j+1} = g(x_j).$$

By assumption, the computed value $\hat{x}_{j+1}$ can be written as

$$\hat{x}_{j+1} = g(\hat{x}_j)(1+\theta_j), \quad |\theta_j| \leq \Theta.$$

It follows that the difference $d_{j+1} = x_{j+1} - \hat{x}_{j+1}$ can be expressed as

$$d_{j+1} = g(x_j) - g(\hat{x}_j)(1+\theta_j) = g(x_j) - g(\hat{x}_j) - g(\hat{x}_j)\theta_j.$$

Since g is Lipschitz continuous with Lipschitz constant $L < 1$ it follows, that

$$|d_{j+1}| \leq L|x_j - \hat{x}_j| + M|\theta_j| \leq L|d_j| + M\Theta.$$

This is the fundamental inquality governed by Theorem 3.3. It follows immediately that

$$|x_n - \hat{x}_n| \leq L^n|x_0 - \hat{x}_0| + \frac{1-L^n}{1-L}M\Theta$$

This completes the proof. ■

When n is sufficiently large, the term L^n is neglible, and we can write

$$|r - \hat{x}_n| \lesssim \frac{1}{1-L}M\Theta.$$

This explains why there is a hard limitation for how accurately r can be computed.

Exercises

1. Let $I = [a,b] \subset I$ and let $f : I \to I$ be a continuous function. Show that f has at least one fixed point. Hint: Consider the auxiliary function $h(x) = x - f(x)$. Why can we assume that $h(a) < 0$ and $0 < h(b)$?

2. Let $\alpha > 0$ and let $\nu = \alpha^{\frac{1}{3}}$. Consider the functional iteration

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

where

$$g(x) = \frac{1}{3}\left(2x + \frac{\alpha}{x^2}\right)$$

   (a) Show that

$$x_{n+1} - \nu = \frac{(x_n-\nu)^2}{x_n} + \frac{(x_n-\nu)^3}{3x_n^2}$$

   (b) Show that

$$\forall n \geq 1 \, : \, x_n \geq \nu$$

   regardless of the choice of $x_0 \in (0,\infty)$.

   (c) Determine an interval I containing ν such that the largest possible interval I around ν such that the functional iteration converges to ν for all $x_0 \in I$.

3. Let $\alpha > 0$ and let p be a positive integer. Consider the functional iteration

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

where

$$g(x) = \frac{1}{p}\left((p-1)x + \frac{\alpha}{x^{p-1}}\right)$$

   (a) Show that $\alpha^{\frac{1}{p}}$ is a fixed point for g.

(b) Show that the functional iteration converges to $\alpha^{\frac{1}{p}}$ for all x_0 sufficiently close to $\alpha^{\frac{1}{p}}$.

4. Consider the function $g : \mathbb{R} \to \mathbb{R}$ given by

$$g(x) = x + \sin(x)$$

(a) Show that π is a fixed point for g and that there exists a closed interval I containing π such that $g\big|_I$ is a contractive mapping of I into I.

(b) Show that the functional iteration

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

converges to π for all $x_0 \in I$ and the order of convergence is 3.

5. Consider the function $g : \mathbb{R} \to \mathbb{R}$ given by

$$g(x) = x + \sin(x) + \frac{1}{6}\sin(x)^3.$$

(a) Show that π is a fixed point for g and that there exists a closed interval I containing π such that $g\big|_I$ is a contractive mapping of I into I.

(b) Show that the functional iteration

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

converges to π for all $x_0 \in I$ and the order of convergence is 5.

6. Consider the function $g\mathbb{R} \to \mathbb{R}$ given by

$$g(x) = x + \sin(x) + \frac{1}{6}\sin(x)^3 + \frac{3}{40}\sin(x)^5.$$

(a) Show that π is a fixed point for g and that there exists a closed interval I containing π such that $g\big|_I$ is a contractive mapping of I into I.

(b) Show that the functional iteration

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

converges to π for all $x_0 \in I$ and the order of convergence is 7.

Let $\alpha > 0$ and consider the problem of computing α^{-1}.

7. Construct a functional iteration $x_{n+1} = g(x_n)$ such that

$$1 - \alpha x_{n+1} = (1 - \alpha x_n)^2.$$

8. Construct a functional iteration $x_{n+1} = g(x_n)$ such that

$$1 - \alpha x_{n+1} = (1 - \alpha x_n)^3.$$

9. Construct a functional iteration $x_{n+1} = g(x_n)$ such that

$$1 - \alpha x_{n+1} = (1 - \alpha x_n)^4.$$

6.6 Convergence and efficiency of iterative methods

In this section how to classify the convergence and effiency of iterative methods. We begin by studying a very special sequence.

Theorem 6.10 *Let $p \geq 1$, $c > 0$, $\epsilon_0 > 0$ and define*

$$\epsilon_{n+1} = c \cdot \epsilon_n^p, \tag{6.8}$$

Then

$$\epsilon_n = c^{\psi(n)} \cdot \epsilon_0^{p^n},$$

where

$$\psi(n) = \sum_{j=0}^{n-1} p^j = \begin{cases} n & p = 1 \\ \frac{p^n-1}{p-1} & p > 1 \end{cases}$$

Moreover,

$$\epsilon_n \to 0, \quad n \to \infty,$$

if and only if

$$c \cdot \epsilon_0^{p-1} < 1$$

In this case, we say that ϵ_n converges to zero with order p.

Proof. ■

By definition, our special sequence has the property

$$\frac{\epsilon_{n+1}}{\epsilon_n^p} = c.$$

In particular, we have the (trivial) property

$$\frac{\epsilon_{n+1}}{\epsilon_n^p} \to c, \quad n \to \infty.$$

This particular feature is a central part of the following definition.

Definition 6.3 **Convergence of order** p *We say that x_n converges to α with order at least p if there exists a sequence ϵ_n, such that*

$$|\alpha - x_n| \leq \epsilon_n$$

and there exists $c > 0$ such that

$$\frac{\epsilon_{n+1}}{\epsilon_n^p} \to c, \quad n \to \infty. \tag{6.9}$$

In the case of $p = 1$ we require $c \in (0, 1)$.

The case of $p = 1$ is called linear convergence. The case of $p = 2$ is called quadratic convergence. The case of $p = 3$ is called cubic convergence.

Example 6.9 Show that the sequence $x_n = \sin(n)2^{-n}$ converges at least linearly to zero.

Solution We have

$$|x_n| \leq 2^{-n}$$

which suggests that we consider

$$\epsilon_n = 2^{-n}.$$

We have

$$\frac{\epsilon_{n+1}}{\epsilon_n} = \frac{2^{-(n+1)}}{2^{-n}} = 2^{n-(n+1)} = \frac{1}{2} \to \frac{1}{2}, \quad n \to \infty.$$

This shows that x_n converges as least linearly to zero. ■

Example 6.10 Show that the sequence $x_n = \sin(n^5)3^{-2^n}$ converges at least quadratically to zero.

Solution We have

$$|x_n| \leq 3^{-2^n}$$

which suggests that we consider

$$\epsilon_n = 3^{-2^n}.$$

We have

$$\frac{\epsilon_{n+1}}{\epsilon_n^2} = \frac{3^{-2^{n+1}}}{\left(3^{-2^n}\right)^2} = \frac{3^{-2^{n+1}}}{3^{-2^n} \cdot 3^{-2^n}} = \frac{3^{-2^{n+1}}}{3^{-(2^n+2^n)}} = 1.$$

This shows that x_n converges at least quadratically to zero. ■

There are two special cases which are not include Definition 6.3.

Definition 6.4 **Sublinear convergence** *We say that x_n converges sublinearly to α if there exists ϵ_n such that*

$$|\alpha - x_n| \leq \epsilon_n$$

and

$$\frac{\epsilon_{n+1}}{\epsilon_n} \to 1, \quad n \to \infty. \tag{6.10}$$

Example 6.11 Show that the sequence $x_n = \frac{\sin(n)}{n}$ converges sublinearly to zero.

Solution We have

$$|x_n| \leq \frac{1}{n}$$

which suggests that we should consider $\epsilon_n = \frac{1}{n}$. Then

$$\frac{\epsilon_{n+1}}{\epsilon_n} = \frac{\frac{1}{n+1}}{\frac{1}{n}} = \frac{n}{n+1} \to 1, \quad n \to \infty.$$

This shows that x_n converges sublinearly to 0. ■

Definition 6.5 **Superlinear convergence** *We say that x_n converges superlinearly to α if there exists ϵ_n such that*

$$|\alpha - x_n| \leq \epsilon_n$$

and

$$\frac{\epsilon_{n+1}}{\epsilon_n} \to 0, \quad n \to \infty. \tag{6.11}$$

Example 6.12 Let $x_n = \sin(n) \cdot 10^{-n^2}$. Show that x_n converges superlinearly to zero.

Solution We have

$$|x_n| \leq 10^{-n^2}$$

which suggest that we consider

$$\epsilon_n = 10^{-n^2}$$

We have

$$\frac{\epsilon_{n+1}}{\epsilon_n} = \frac{10^{-(n+1)^2}}{10^{-n^2}} = 10^{n^2-(n+1)^2} = 10^{-2n-1} \to 0,$$

This shows that x_n converges superlinearly to zero. ■

Now consider an iterative method which converges to a root with order of convergence $p > 1$. We wish to evaluate the efficiency of the method and determine how rapidly the error decays as a function of the work which has been done.

By definition, there exists a sequence ϵ_n such that

$$|\alpha - x_n| \leq \epsilon_n$$

where

$$\frac{\epsilon_{n+1}}{\epsilon_n^p} \to c, \quad n \in \mathbb{N}.$$

Now suppose that each iteration requires m units of work. A unit of work is typically one function evaluation. Then

$$|\alpha - x_n| \lesssim c^{\frac{1}{1-p}} (c^{\frac{1}{p-1}} \epsilon_0)^{p^n} = c^{\frac{1}{1-p}} (c^{\frac{1}{p-1}} \epsilon_0)^{(p^{1/m})^{nm}} \tag{6.12}$$

The product nm is the work required to complete n iterations. Equation (6.12) is the motivation behind the following definition

Definition 6.6 **Efficiency index** *An iterative method which has order of congence $p > 1$ and requires m units of work per iteration has efficiency index $p^{1/m}$. If $p = 1$, then the efficiency index is 1.*

Example 6.13 Show that the sequence $x_n = n\sin(n)2^{-n}$ converges at least linearly to $\alpha = 0$.

Solution Since $|\sin(t)| \leq 1$ for all $t \in \mathbb{R}$ we have

$$|\alpha - x_n| = |x_n| \leq n2^{-n}.$$

We therefore consider $\epsilon_n = n2^{-n}$. We find that

$$\frac{\epsilon_{n+1}}{\epsilon_n} = \frac{(n+1)2^{-n+1}}{n2^{-n}} = \frac{1}{2}\frac{(n+1)}{n} \to \frac{1}{2}.$$

This shows that x_n converges to 0 at least linearly. ■

Example 6.14 Let $|q| < 1$ and consider the geometric series

$$\alpha = \sum_{j=0}^{\infty} q^j = \frac{1}{1-q}$$

Let x_n be given by

$$x_n = \sum_{j=0}^{n} q^j$$

Show that x_n converges at least linearly to α.

Solution We begin by estimating the difference between α and x_n. By the triangle inequality we have

$$|\alpha - x_n| = \left| \sum_{j=n+1}^{\infty} q^j \right| \leq \sum_{j=n+1}^{\infty} |q|^j = \frac{|q|^{n+1}}{1-|q|}.$$

This suggests that we define

$$\epsilon_n = \frac{|q|^{n+1}}{1-|q|}.$$

It follows immediately, that

$$\frac{\epsilon_{n+1}}{\epsilon_n} = |q| < 1.$$

This shows that x_n converges to α at least linearly. ■

Definition 6.7 **Convergence of order** p *Let $p > 1$. We say that x_n converges to α with order at least p, if there exists a $\epsilon_n > 0$ and a $c > 0$ such that*

$$\frac{\epsilon_{n+1}}{\epsilon_n^p} \to c, \quad n \to \infty.$$

If $p = 1$, we require $c \in (0, 1)$.

Example 6.15 Consider the sequence $x_n = \sin(n)n \cdot 10^{-2^n}$. Show that x_n converges at least quadratically to 0.

Solution We have the upper bound

$$|x_n| \leq n10^{-2^n}$$

which suggests that we define $\epsilon_n = n \cdot 10^{-n^2}$. Then we have

$$\frac{\epsilon_{n+1}}{\epsilon_n^2} = \frac{(n+1) \cdot 10^{-2^{n+1}}}{(n \cdot 10^{-2^n})^2} = \frac{n+1}{n} \to 1, \quad n \to \infty$$

This shows that x_n converges at least quadratically to 0. ■

Exercises

1. Let $a \in (-1, 1)$ and $b \in \mathbb{R}$ and consider the sequence given by

   $$x_0 = b, \quad x_n = ax_{n-1} + b$$

   Show that x_n converges at least linearly to $\alpha = \frac{b}{1-a}$.

2. Suppose that x_n converges to α with order of convergence $p > 1$. Show that the convergence is super-linear.

6.7 The design of reliable iterative methods

A good routine for solving the non-linear equation

$$f(x) = 0$$

should satisfy the following two criteria.

1. It should maintain and shrink a bracket around the root.
2. It should be fast.

Robust and fast methods are needed for all serious applications where failure is unacceptable and speed is of the essence.

1. The bisection method is robust, but it is not fast.
2. Newton's method is not robust. It is fast, provided
   (a) the derivative is simple to compute, and
   (b) we start sufficiently close to a root
3. The secant method it is not robust. It is fast, provided we start sufficiently close to a root.

A robust and fast method can be constructed by combined the best features of bisection and the secant method.

6.8 Newton's method for systems of equations

Newton's method generalizes readily to the case of $f : \Omega \to \mathbb{R}$ where Ω is an open subset of $\mathbb{R}^n$. The derivative f' is replaced with the Jacobian, the matrix of first order derivatives. Specifically, if

$$f(x) = (f_1(x), f_2(x), \quad f_n(x))^T$$

then the Jacobian of f at the point x is given by

$$Df(x) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \cdots & \cdots & & \cdots \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}$$

If the Jacobian is non-singular, then Newton's method is given by

$$x_{n+1} = x_n - Df(x)^{-1} f(x_n).$$

Chapter 7

Numerical differentiation

7.1 Basic techniques

7.2 Practical error estimation

Exercises

Computer problems

Chapter 8

Richardson extrapolation

In many computations, the accuracy is controlled by a small set of real parameters. As an example, we consider the problem of computing the target value

$$T = f'(x)$$

using the approximation A_h

$$A_h = \frac{f(x+h) - f(x-h)}{2h}$$

Here the accuracy is controlled by a single real parameter, i.e., the stepsize h. By Taylor's formula, we have

$$T - A_h = f'(x) - \frac{f(x+h) - f(x-h)}{2h} = O(h^2), \quad h \rightarrow 0, \quad h > 0.$$

In exact arithmetic, we can use A_h to evaluate T as accurately as we like. However, in practice rounding errors will impose a limit on the accuracy which can be achieved. In particular, as h tends to zero, subtractive cancellation in the expression $f(x+h) - f(x-h)$ will introduce a large relative error and the computed value $\hat{A}_h$ will not be an accurate approximation of T.

In this chapter we consider the general problem of approximating a target value T using approximations $A = A_h$ which depend on a single real parameter h. In general, the computed approximation $\hat{A}_H$ will differ from the the exact approximation A_h. We will show how to detect when the difference is irrelevant and how to estimate the error $E_h = T - A_h$ accurately.

8.1 Asymptotic error expansions

Throughout this chapter we will assume that the error $E_h = T - A(h)$ obeys an *asymptotic error expansion*, i.e., there exists a positive integer m and real numbers p_j and s with

$$0 < p_1 < p_2 < \ldots < p_m < s. \tag{8.1}$$

and real numbers $\alpha_j \neq 0$, such that

$$E_h = \sum_{j=1}^{m} \alpha_j h^{p_j} + O(h^s), \qquad h \rightarrow 0, \qquad h > 0. \tag{8.2}$$

An asymptotic error expansion resembles a Taylor expansion of a smooth function, but the exponents $p_j > 0$ need not be integers. The term $\alpha_1 h^{p_1}$ is called the *primary* error term. The term $\alpha_2 h^{p_2}$ is called the secondary error term and so on. Frequently, it is convinient to focus on the first few terms and simply write

$$E_h = \alpha h^p + \beta h^q + O(h^r), \tag{8.3}$$

where

$$0 < p < q < r.$$

Example 8.1 Let $I \subseteq \mathbb{R}$ be an open interval and let $f \in C^7(I, \mathbb{R})$. Let $x \in I$ and consider the problem of computing the derivative $T = f'(x)$ of a smooth function using the finite difference approximation

$$A_h = \frac{f(x+h) - f(x-h)}{2h}$$

Show that the error $E_h = T - A_h$ satisfies an asymptotic error expansion of the form

$$E_h = \alpha h^2 + \beta h^4 + O(h^6), \qquad h \to 0, \qquad h \neq 0.$$

Solution Let $h_0 > 0$ be so small that $J = [x - h_0; x + h_0] \subset (a, b)$. Let $0 < h < h_0$. Then by Taylor's formula, there exists $\xi_1, \xi_2 \in I$ such that

$$\begin{aligned} f(x+h) = f(x) + f'(x)h + \frac{1}{2!}f''(x)h^2 + \frac{1}{3!}f^{(3)}(x)h^3 \\ + \frac{1}{4!}f^{(4)}(x)h^4 + \frac{1}{5!}f^{(5)}(x)h^5 + \frac{1}{6!}f^{(6)}(x)h^6 + \frac{1}{7!}f^{(7)}(\xi_1)h^7 \end{aligned} \tag{8.4}$$

and

$$\begin{aligned} f(x-h) = f(x) - f'(x)h + \frac{1}{2!}f''(x)h^2 - \frac{1}{3!}f^{(3)}(x)h^3 \\ + \frac{1}{4!}f^{(4)}(x)h^4 - \frac{1}{5!}f^{(5)}(x)h^5 + \frac{1}{6!}f^{(6)}(x)h^6 - \frac{1}{7!}f^{(7)}(\xi_2)h^7. \end{aligned} \tag{8.5}$$

The even powers of h cancel when we subtract equation (8.5) from (8.4). We have

$$f(x+h) - f(x-h) = 2f'(x)h + \frac{2}{3!}f^{(3)}(x)h^3 + \frac{2}{5!}f^{(5)}(x)h^5 + \frac{1}{7!}\left(f^{(7)}(\xi_1) + f^{(7)}(\xi_2)\right)h^7$$

It follows that

$$T - A_h = \alpha h^2 + \beta h^4 + \frac{1}{7!}\left(f^{(7)}(\xi_1) + f^{(7)}(\xi_2)\right)h^6.$$

where

$$\alpha = -\frac{1}{3!}f^{(3)}(x), \quad \beta = -\frac{1}{5!}f^{(5)}(x).$$

Since J is a closed and bounded interval and $f^{(7)}$ is continuous there exists a constant $M > 0$, such that

$$\forall y \in J \, : \, |f^{(7)}(y)| \leq M$$

We can therefore conclude that

$$\left|f'(x) - A_h - \alpha h^2 - \beta h^4\right| \leq Ch^6, \qquad C = \frac{2}{7!}M.$$

This shows that

$$f'(x) - A_h = \alpha h^2 + \beta h^4 + O(h^6), \qquad h \to 0, \qquad h \neq 0.$$

■

In most cases, it is much more difficult to prove the existence of an asymptotic error and we will not discuss this issue further. Frequently, the coefficients α and β as well as the exponents p, q, and r are also unknown. Nevertheless, it is possible to obtain reliable error estimates by computing A_h for several different values of h!

We begin by emphasising the importance of the primary error term. We have

$$E_h = \alpha h^p \left(1 + \frac{\beta h^q}{\alpha h^p} + O(h^n)\right).$$

This shows that

$$\frac{E_h}{\alpha h^p} \to 1, \qquad h \to 0, \qquad h \neq 0.$$

It follows that

$$E_h \approx \alpha h^p$$

is a good approximation, when h is sufficiently small.

The following theorem show that it is possible to compute an accurate approximation of the primary error term using A_h and A_{2h}.

Theorem 8.1 *If A_h satisfies the asymptotic error expansion*

$$T - A_h = \alpha h^p + \beta h^q + O(h^r), \quad \alpha \neq 0, \quad h \to 0, \quad h \neq 0 \tag{8.6}$$

then

$$\frac{A_h - A_{2h}}{2^p - 1} = \alpha h^p + \left(\frac{2^p - 1}{2^q - 1}\right) \beta h^p + O(h^r). \tag{8.7}$$

Proof. By replacing h by $2h$ in (8.6) we obtain

$$T - A_{2h} = 2^p \alpha h^p + 2^q \beta h^q + O(h^r). \tag{8.8}$$

Here we have actively used that

$$O(c \cdot h^r) = O(h^r),$$

for all constants $c \neq 0$. By subtracting equation (8.6) from equation (8.8) we obtain

$$A_h - A_{2h} = (2^p - 1)\alpha h^p + (2^q - 1)\beta h^q + O(h^r). \tag{8.9}$$

Here we have actively used that $f, g \in O(h^r)$ implies $f - g \in O(h^r)$. We obtain (8.7) by dividing with $2^p - 1 \neq 0$. ■

This results motivates the following definition

Definition 8.1 *Richardson's error estimate E_h^{est} is given by*

$$E_h^{est} = \frac{A_h - A_{2h}}{2^p - 1}.$$

By Theorem 8.1 we have

$$\frac{\alpha h^p}{E_h^{\text{est}}} \to 1, \qquad h \to 0, \qquad h \neq 0$$

which shows that

$$\alpha h^p \approx E_h^{\text{est}}$$

is a good approximation when h is sufficiently small. However, we can not compute E_h^{est} unless we know the correct value of p. The following theorem show that it is possible to determine the correct value of p merely by observing the behavoir of certain auxiliary numbers. We require the following definition.

Definition 8.2 *If A_h, A_{2h} and A_{4h} are all defined, and $A_h \neq A_{2h}$, then Richardson's fraction F_h is defined by*

$$F_h = \frac{A_{2h} - A_{4h}}{A_h - A_{2h}}.$$

The following theorem gives key properties of Richardson's fraction.

Theorem 8.2 *If A_h satisfies the asymptotic error expansion*

$$T - A_h = \alpha h^p + \beta h^q + O(h^r), \quad \alpha \neq 0, \quad h \to 0, \quad h \neq 0 \tag{8.10}$$

then Richardson's fraction F_h satisfies

$$F_h = 2^p \cdot \frac{1 + c_2 h^m + O(h^n)}{1 + c_1 h^m + O(h^n)} \tag{8.11}$$

where the exponents m and n are given by

$$m = q - p, \qquad n = r - p, \tag{8.12}$$

and the coefficients c_1 and c_2 are given by

$$c_1 = \left(\frac{2^p - 1}{2^q - 1}\right)\frac{\alpha}{\beta}, \qquad c_2 = 2^m c_1. \tag{8.13}$$

In particular, F_h converges to 2^p as h tends to zero and

$$F_h - 2^p = O(h^m), \quad h \to 0, \quad h \neq 0. \tag{8.14}$$

Proof. By replacing h by $2h$ in equation (8.9) we obtain

$$A_{2h} - A_{4h} = 2^p(2^p - 1)\alpha h^p + 2^q(2^q - 1)\beta h^q + O(h^r). \tag{8.15}$$

The fraction F_h can therefore be written as

$$F_h = \frac{A_{2h} - A_{4h}}{A_h - A_{2h}} = \frac{2^p(2^p - 1)\alpha h^p + 2^q(2^q - 1)\beta h^q + O(h^r)}{(2^p - 1)\alpha h^p + (2^q - 1)\beta h^q + O(h^r)}.$$

We now divide each term in the numerator by 2^p to obtain

$$F_h = 2^p \cdot \frac{(2^p - 1)\alpha h^p + 2^m(2^q - 1)\beta h^q + O(h^r)}{(2^p - 1)\alpha h^p + (2^q - 1)\beta h^q + O(h^r)}.$$

Since $\alpha \neq 0$ and $h \neq 0$, we can divide the numerator and the denominator by αh^p to obtain

$$F_h = 2^p \cdot \frac{1 + 2^m \left(\frac{2^q-1}{2^p-1}\right)\frac{\beta}{\alpha} h^m + O(h^n)}{1 + \left(\frac{2^q-1}{2^p-1}\right)\frac{\beta}{\alpha} h^m + O(h^n)} = 2^p \cdot \frac{1 + c_2 h^m + O(h^n)}{1 + c_1 h^m + O(h^n)}. \tag{8.16}$$

This completes the proof of equation (8.11). Finally, since $p < q$ and $p < r$ it is clear that

$$F_h - 2^p = O(2^m), \qquad h \to 0, \qquad h \neq 0.$$

This completes the proof. ∎

Theorem 8.2 states that it is possible to determine the value of p simply by computing F_h for decreasing values of h. Moreover, by studying the decay of $F_h - 2^p$ as h tends to zero we can also make an educated guess as to the value of $q = m + p$.

Remark 8.1 The error estimate E_h^{est} and the fraction F_h are named in honour of Lewis Fry Richardson (1881-1953) who pioneered these techniques. Richardson made numerous contribution to numerical analysis, accoustics, weather prediction and the mathematical analysis of warfare.

Exercises

1. Let $f : \mathbb{R} \to \mathbb{R}$ be infinitely differentiable and let $x \in \mathbb{R}$. Consider the problem of computing the derivative $f'(x)$ using the approximation

$$A_h = \frac{f(x+h) - f(x)}{h}.$$

Show that the error E_h satisfies an asymptotic error expansion of the form

$$T - A_h = \alpha h + \beta h^2 + O(h^3), \quad h \to 0, \quad h > 0.$$

Obtain explicit formula for the coefficients α and β.

2. Examine your solution of Problem 1. What is the smallest value of k such that your analysis holds true for $f \in C^k$?

3. Let $f : \mathbb{R} \to \mathbb{R}$ be infinitely differentiable and let $x \in \mathbb{R}$. Consider the problem of computing the derivative $f''(x)$ using the approximation

$$A_h = \frac{f(x+h) - 2f(x) - f(x-h)}{h^2}.$$

Show that the error E_h satisfies an asymptotic error expansion of the form

$$T - A_h = \alpha h^2 + \beta h^4 + O(h^6), \quad h \to 0, \quad h > 0.$$

Obtain explicit formula for the coefficients α and β.

4. Examine your solution of Problem 3. What is the smallest value of k such that your analysis holds true for $f \in C^k$?

5. Let $f : \mathbb{R} \to \mathbb{R}$ be infinitely differentiable and let $x \in \mathbb{R}$. Consider the problem of computing the derivative $f'(x)$ using the on one sided approximation

$$A_h = \frac{-3f(x) + 4f(x+h) - f(x+2h)}{2h}.$$

Show that the error E_h satisfies an asymptotic error expansion of the form

$$T - A_h = \alpha h^2 + \beta h^3 + O(h^4), \quad h \to 0, \quad h > 0$$

Obtain explicit formula for the coefficients α and β.

6. Examine your solution of Problem 5. What is the smallest value of k such that your analysis holds true for $f \in C^k$?

8.2 Practical error estimation

The results given in the previous section refer to the *exact* value of A_h. In practice, rounding errors will cause the computed value $\hat{A}_h$ to differ from the exact value A_h. Moreover, while the approximation

$$E_h \approx E_h^{\text{est}} = \frac{A_h - A_{2h}}{2^p - 1}$$

is good for h sufficiently small our analysis does not suggest when h *is* sufficiently small. In this section we show how to determine when rounding errors are irrelevant and when the error estimate can be trusted.

The key is to examine the behavior of the computed value $\hat{F}_h$ of Richardson's fraction F_h. For the exact value F_h we have the expression

$$F_h = 2^p \cdot \frac{1 + c_2 h^m + O(h^n)}{1 + c_1 h^m + O(h^n)},$$

where

$$c_1 = \left(\frac{2^p - 1}{2^q - 1}\right)\frac{\alpha}{\beta}, \quad c_2 = 2^m c_1.$$

The higher order terms obscure the behavoir of F_h. If we discard the higher order terms we obtain the fraction

$$G_h = 2^p \cdot \frac{1 + c_2 h^m}{1 + c_1 h^m},$$

whose behavior will be characterized shortly. It is easy to verify that

$$\frac{F_h}{G_h} \to 1, \qquad h \to 0, \qquad h \neq 0,$$

which implies that G_h is a good approximation of F_h for sufficiently small values of h. We have the following lemma.

Lemma 8.1 *The fraction G_h converges monotonically to 2^p. The convergence is controlled entirely by the sign of $\nu = \beta/\alpha$. If $\nu > 0$, then the convergence is strictly decreasing and if $\nu < 0$, then the convergence is stricly increasing.*

Proof. It is clear that function $h \to G_h$ is defined and differentiable as long as $1 + c_1 h^m \neq 0$. Let $m = q - p$. Then

$$G_h = 2^p \cdot \frac{1 + c_2 h^m}{1 + c_1 h^m},$$

and the derivative with respect to h is simply

$$G_h' = 2^p \cdot \frac{c_2 m h^{m-1}(1 + c_1 h^{m-1}) - (1 + c_2 h^m) c_1 m h^{q-1}}{(1 + c_1 h^q)^2} = 2^p \cdot \frac{(c_2 - c_1)}{(1 + c_1 h^q)^2} m h^{m-1}$$

Since $c_2 = 2^m c_1$ this expression reduces to

$$G_h' = 2^p \cdot \frac{(2^m - 1) c_1}{(1 + c_1 h^q)^2} m h^{m-1}$$

Now since $0 < m$, we conclude that the sign of G_h' is identical to the sign of c_1. Since

$$c_1 = \left(\frac{2^p - 1}{2^q - 1}\right)\frac{\alpha}{\beta}$$

we conclude that G_h is a strictly increasing function of h if $\nu > 0$ and G_h is a strictly decreasing function of h is $\nu < 0$. This completes the proof. ■

We now consider the behavior of F_h as h tends to zero. For sufficiently small h, F_h will converge monotonically towards 2^p and $F_h - 2^p = O(h^m)$. Initially, the difference between F_h and the computed value $\hat{F}_h$ is caused by rounding errors in the computation of A_h, A_{2h} and A_{4h}. As long as the computed value $\hat{F}_h$ follows the pattern predicted for F_h there is no reason to suspect that these rounding errors are significant. Moreover, since the relative importance of the higher order terms is decaying, the accuracy of Richardson's error estimate E_h^{est} is increasing. At some point, the computation of $A_h - A_{2h}$ and $A_{2h} - A_{4h}$ will start to suffer from subtractive cancellation and the values of $\hat{F}_h$ will start to deviate from the predicted pattern. This does not imply that the corresponding error estimates are worthless, but their accuracy will not increase.

Eventually, the computation of $A_h - A_{2h}$ and $A_{2h} - A_{4h}$ will suffer from catastropic cancellation. The values of $\hat{F}_h$ will differ significantly from F_h and the computed error estimate $\hat{E}_h^{\text{est}}$ will be inaccurate.

The proper application of Richardson's techniques requires substantial training. We begin with some examples where the exact result is known.

Example 8.2 Differentiation of simple function

Solution ■

Example 8.3 Integration of simple function

Solution ■

Example 8.4 Compute the range of an artillery shell

Solution ■

The existence of an asymptotic error expansion typically requires that all relevant functions are sufficiently differentiable. This is an example where the degree of differentiability is minimal. Richardson's techniques can still be utilized, but the order of the primary error term is not an integer.

Example 8.5 Integration of square root function.

Solution ■

Exercises

1. Let A_h denote a method which satisfies an asymptotic error expansion and let p denote the order of the primary error term. Consider the calculation of Richardson's fraction F_h.
   (a) Why does this calculation suffer from subtractive cancellation?
   (b) Why is subtractive cancellation less of an issue if p is small?

8.2.1 Error bounds

Consider the problem of computing a target value T. Ideally, we wish to find values a and b such $a < T < b$ and $b - a$ is sufficiently small. Occasionally, it is sufficient to bound T from below ($a < T$) or from above ($T < b$). In this section we explore how Richardson's techniques can be used to establish bounds for T.

Let us assume that A_h satisfies an asymptotic error expansion of the form

$$T - A_h = \alpha h^p + \beta h^q + O(h^r), \qquad h \to 0, \qquad h > 0.$$

Then the sign of the error $E_h = T - A_h$ is identical to the sign of α for h sufficiently small. Similarly, the sign of $T - A_h - \alpha h^p$ is identical to the sign of β for h sufficiently small. The sign of α is immediately available as the sign of Richardson's error estimate E_h^{est}. The sign of β can be determined by observing the convergence of Richardson's fraction F_h. If F_h is (ultimately) decreasing, then α and β have the same sign. If F_h is (ultimately) increasing, then α and β have different sign. The following lemma summaries the inequalities which can be established depending on the signs of α and β.

Lemma 8.2 *1. If $\alpha > 0$ and $\beta > 0$, then*

$$A_h < A_h + \alpha h^p < T \tag{8.17}$$

2. If $\alpha < 0$ *and* $\beta < 0$*, then*

$$T < A_h + \alpha h^p < A_h. \tag{8.18}$$

3. If $\alpha > 0$ *and* $\beta < 0$*, then*

$$A_h < T < A_h + \alpha h^p. \tag{8.19}$$

4. If $\alpha < 0$ *and* $\beta > 0$*, then*

$$A_h + \alpha h^p < T < A_h. \tag{8.20}$$

In short, if α and β have the same sign then we can find either a lower bound of T or an upper bound of T, but not both. However, if α and β have different sign, then we can find a bracket around T.

Exercises

1. Find a real application where a lower bound $a < T$ of the target value T is more useful than an upper bound ($T < b$).
2. Find a real application where an upper bound $T < b$ of the target value T is more useful than a lower bound ($a < T$).
3. Prove Lemma 8.2.

Computer problems

1. Write and test a `MATLAB` function `Xrichardson` which computes Richardson's fractions and error estimates. It must accept as input a sequence of list of approximation $\{A_j\}_{j=1}^m$ and as well as the order of the primary error term. It is assumed that the approximation A_j corresponds to the stepsize $h_j = 2^{-j}h_1$. It is the responsibility of the user to examine Richardson's fraction and determine the correct order of the primary error term.
2.

Chapter 9

Numerical integration

Contents

In this chapter we conside the problem of approximating the integral

$$I = \int_a^b f(x)dx$$

where $f : [a, b] \to \mathbb{R}$ is a continuous function. If f has a known anti-derivative F, i.e., a differentiable function such that

$$F'(x) = f(x)$$

then we can simply compute

$$\int_a^b f(x)dx = F(b) - F(a).$$

However, frequently we can not find an explicit formula for an anti-derivative or it is not practical to apply the formula. The only alternative is to use numerical methods. As always, our goal is provide approximations, error bounds and reliable error estimates.

It is convenient to view the integral I as a function which maps a continuous function f to the real number $\int_a^b f(x)dx$. It is well known that

$$\int_a^b \alpha f(x) + \beta g(x)dx = \alpha \int_a^b f(x)dx + \beta \int_a^b g(x)dx$$

for all $\alpha, \beta \in \mathbb{R}$ and functions $f, g \in C([a, b], \mathbb{R})$. Formally, we say that $I : C([a, b], \mathbb{R}) \to \mathbb{R}$ is the linear functional given by

$$I(f) = \int_a^b f(x)dx.$$

9.1 Integration based on interpolation

It is natural to approximate f with a simpler function g and then approximate $I(f)$ with $I(g)$, i.e.,

$$\int_a^b f(x)dx \approx \int_a^b g(x)dx$$

Polynomials are a particularly easy to integrate and the

9.2 The trapezoidal rule

The trapezoidal rule is the linear functional $T : C([a,b],\mathbb{R}) \to \mathbb{R}$ given by

$$T(f) = \frac{b-a}{2}\Big[f(a) + f(b)\Big].$$

Example 9.1 Use the trapezoidal rule to approximate the integral $\int_0^1 e^x dx$ and compute the error.

Solution Let $f : [0,1] \to \mathbb{R}$ be given by $f(x) = e^x$. The target value is

$$I(f) = \int_0^1 e^x dx = e - 1 \approx 1.7183.$$

The trapezoidal rule is

$$T(f) = \frac{1}{2}\Big[f(0) + f(1)\Big] = \frac{1}{2}\Big[1 + e\Big] \approx 1.8591.$$

The error is

$$I(f) - T(f) = \frac{1}{2}(e-3) \approx -1.4086 \times 10^{-1}.$$

■

The trapezoidal rule can be derived by approximating f by a straight line through the points $(a, f(a))$ and $(b, f(b))$. Therefore, it is no surprise that the trapezoidal rule is exact for all polynomials of degree at most 1. However, there is no shame in verifying simple matters.

Example 9.2 Show that the trapezoidal rule is exact for all polynomials of degree at most 1.

Solution The error $E : C([a,b],\mathbb{R}) \to \mathbb{R}$ is given by

$$E(f) = I(f) - T(f).$$

The error is a linear functional because I and T are linear functionals. A polynomial p of degree at most 1 can be written as $p(x) = ax + b$ for $a, b \in \mathbb{R}$. Since E is a linear functional we have

$$E(p) = E(ax + b) = aE(x) + bE(1).$$

We conclude that $E(p) = 0$ for all polynomials p of degree at most 1 if and only if $E(1) = 0$ and $E(x) = 0$. We now explicitly compute $E(1)$ and $E(0)$. We have

$$E(1) = \int_a^b 1dx - \frac{b-a}{2}\Big[1 + 1\Big] = 0$$

and

$$E(x) = \int_a^b xdx - \frac{b-a}{2}\Big[a + b\Big] = \frac{1}{2}(b^2 - a^2) - \frac{b-a}{2}(b+a) = 0.$$

We conclude that the trapezoidal rule is exact for all polynomials of degree at most 1. ■

A composite trapezoidal rule can be constructed as follows.

Definition 9.1 *Let n be a positive integer, let $h = (b-a)/n$ and let $x_j = a + jh$ for $j = 0, 1, 2, \dots, n$. Then the composite trapezoidal rule T_h corresponding to the uniform stepsize h is given by*

$$T_h(f) = \sum_{j=0}^{n-1} \frac{h}{2}\Big[f(x_j) + f(x_{j+1})\Big]. \tag{9.1}$$

We now seek to analyse the error, i.e., the difference between the integral and the composite trapezoidal rule. We begin by studying the trapezoidal rule for the interval $[0, h]$. Here, the target value is

$$I(f) = \int_0^h f(t)dt$$

and the corresponding trapezoial rule is

$$T(f) = \frac{h}{2}\Big[f(0) + f(h)\Big].$$

Let F denote any anti-derivative of f and let $g : [0, h] \rightarrow \mathbb{R}$ be given by

$$g(x) = F(x) - F(0) - \frac{x}{2}\Big[f(x) + f(0)\Big]. \tag{9.2}$$

This auxiliary function is relevant precisely because

$$g(h) = I(f) - T(f).$$

The properties of g will allow us to make precise statements about the error. We begin by deriving an error formula.

Theorem 9.1 *Let $h > 0$ and let $f \in C^2([0, h])$ and let $g : [0, h] \rightarrow \mathbb{R}$ be given by (9.2). Then there exists $\xi \in (0, h)$ such that*

$$g(h) = -\frac{1}{12} f''(\xi)h^3.$$

Proof. The auxiliary function g is C^2 because f is C^2. By direct application of the rules of differentiation we have

$$\begin{aligned} g'(x) &= \frac{1}{2}\Big(f(x) - f(0)\Big) - \frac{x}{2} f(x), \\ g''(x) &= -\frac{x}{2} f''(x). \end{aligned}$$

Since $g(0) = 0$, then the extended mean value theorem implies there exist $x_1 \in (0, h)$ such that

$$\frac{g(h)}{h^3} = \frac{g(h) - g(0)}{h^3 - 0^3} = \frac{g'(x_1)}{3x_1^2}.$$

Since $g'(0) = 0$, then extended mean value theorem implies there exists $x_2 \in (0, x_1)$ such that

$$\frac{g'(x_1)}{3x_1^2} = \frac{g'(x_1) - g'(0)}{3x_1^2 - 3 \cdot 0^2} = \frac{g''(x_2)}{6x_2} = -\frac{1}{12} f''(x_3).$$

We conclude that

$$g(h) = -\frac{1}{12} f''(\xi)h^3,$$

for at least one $\xi \in (0, h)$, namely $\xi = x_3$. ■

Let p be a polynomial of degree at most 1, then $p''(x) = 0$ for all x. By Theorem 9.1 the trapezoidal rule is exact for all polynomials of degree at most 1. This is consistent with Example 9.2. We can now derive a formula for the error for the composite trapezoidal rule.

Theorem 9.2 *Let $-\infty < a < b < \infty$ and let $f \in C^2([a,b],\mathbb{R})$. Let T_h denote the composite trapezoidal rule corresponding to the uniform stepsize h. Then there exists $\xi \in [a,b]$ such that*

$$T - T_h = -\frac{1}{12} f''(\nu)(b-a)h^2.$$

Proof. For each subinterval $[x_j, x_{j+1}]$ we have ξ_j such that

$$\int_{x_j}^{x_{j+1}} f(t)dt = \frac{h}{2}\Big[f(x_j) + f(x_{j+1})\Big] - \frac{1}{12} f''(\xi_j)h^3.$$

It follows that

$$\int_a^b f(t)dt = \sum_{j=0}^{n-1} \frac{h}{2}\Big[f(x_j) + f(x_{j+1}\Big] - \frac{1}{12}\sum_{j=0}^{n-1} f''(\xi_j)h^3.$$

By the intermediate value theorem for continuous functions there exists a $\nu \in [a,b]$ such that

$$f''(\nu) = \frac{1}{n}\sum_{j=0}^{n-1} f''(\xi_j).$$

It follows that

$$\int_a^b f(t)dt - \sum_{j=0}^{n-1} \frac{1}{h}\Big(f(x_j) + f(x_{j+1}\Big) = -\frac{1}{12} f''(\nu)nh^3 = -\frac{1}{12} f''(\nu)(b-a)h^2.$$

This completes the proof. ■

It is important to recognize the strengths and the limitations of each theorem. Theorem 9.2 is no exception. It ensures, that there is at least one $\xi \in [a,b]$ such that

$$T - T_h = -\frac{1}{12} f''(\nu)(b-a)h^2$$

but it gives no indication of the exact location of $\xi \in [a,b]$. Theorem 9.2 is primarily useful when additional information is available.

Example 9.3 Let $f \in C^2([a,b],\mathbb{R})$ be a convex function. Show that the composite trapezoidal rule T_h satisfies $I(f) \leq T_h(f)$.

Solution Since f is C^2 and convex we have $f''(x) \geq 0$ for all x. By Theorem 9.2 there exists a ξ such that

$$I(f) - T_h(f) = -\frac{1}{12} f''(\xi)h^2$$

We conclude that $I(f) - T_h(f) \leq 0$ or equivalently $I(f) \leq T_h(f)$. ■

We also have the following error bound.

Corollary 9.1 *Let $-\infty < a < b < \infty$ and let $f \in C^2([a,b],\mathbb{R})$. Then the composite trapezoidal rule T_h satisfies the error bound*

$$|I(f) - T_h(f)| \leq \frac{1}{12}\|f''\|_\infty (b-a)h^2 \tag{9.3}$$

where

$$\|f''\|_\infty = \max\{|f''(x)| \mid x \in [a,b]\}. \tag{9.4}$$

Proof. The proof follows as a direct application of the error formula given by Theorem 9.2. The details are left to the reader. ■

Example 9.4 Let $f : [0, 1] \to \mathbb{R}$ be given by $f(x) = e^x$. Find $h_0 > 0$, such that $|I(f) - T_h(f)| \leq \tau = 10^{-4}$ for all $h \leq h_0$.

Solution It is clear that $f \in C^2([0, 1], \mathbb{R})$ so Corollary 9.1 can be applied. We have $f''(x) = e^x$ and $\|f''\|_\infty = e$. It follows that

$$|I(f) - T_h(f)| \leq \frac{e}{12}h^2.$$

We can determine h_0 by solving the equation $\frac{e}{12}h_0^2 = 10^{-4}$. We have

$$h_0 = \sqrt{\frac{12 \cdot 10^{-4}}{e}} \approx 2.1011 \times 10^{-2}.$$

If we use at least $n = 48$ subintervals, then we are certain that the error will be less than $\tau = 10^{-4}$. ■

Corollary 9.1 is the statement that the error can be reduced below a given threshold τ if the stepsize h is sufficiently small, i.e., $h \leq h_0$. In practice, the error bound is pessimistic and an accurate approximation can be achieved using a larger stepsize than predicted.

It is important to appreciate that Corollary 9.1 refers to the *exact* value T_h of the composite trapezoidal rule. The computed value $\hat{T}_h$ will typically differ from the exact value because of rounding errors. Fortunately, the techniques developed in Chapter 8 can be applied to determine when the rounding errors are irrelevant and extract accurate error estimates. This is the topic of the Section 9.5.

Exercises

1. Let $-\infty < a < b < \infty$. Show that the composite trapezoidal rule always provideds an upper bound of the integral $\int_a^b \exp(x)dx$.

2. Let $-\infty < a < b < \infty$ and let $f \in C^2([a, b])$ be a concave function. Show that the composite trapezoidal rule provides a lower bound for $\int_a^b f(t)dt$.

The following two problem studies the trapezoidal rule when the function f is C^1 rather than C^2.

3. Let $h > 0$ and let $f \in C^1([-h, h])$ and define $g : [0, h] \to \mathbb{R}$ by

$$g(x) = \int_{-x}^{x} f(t)dt - x\Big(f(x) + f(-x)\Big)$$

(a) Show that

$$g(x) = F(x) - F(-x) - x\Big(f(x) + f(-x)\Big)$$

where F is any anti derivative of f.

(b) Show that

$$|g(h)| \leq \frac{1}{3}\|f'\|_\infty h^2 \tag{9.5}$$

Hint: The extended mean value theorem.

4. Let $\infty < a < b < \infty$, let $f \in C^1([a, b], \mathbb{R})$ and let T_h denote the trapezoidal rule corresponding to the uniform stepsize h. Show that

$$|I(f) - T_h(f)| \leq \frac{1}{12}\|f'\|_\infty h$$

where

$$\|f'\|_\infty = \max\{|f'(x)| \ : \ x \in [a, b]\}.$$

Computer problems

1. Verify Theorem 9.1 in the case of $f : [0,1] \to \mathbb{R}$ given by $f(x) = xe^x$. You will have to accurately solve the nonlinear equation

$$\int_0^1 te^t dt - \frac{1}{2}\Big(f(0) + f(1)\Big) = -\frac{1}{12} f''(\xi)$$

with respect to $\xi \in (0,1)$.

9.3 The method of undetermined coefficients

The trapezoidal rule can be derived through geometrical observations. It is useful to have a procedure which generalizes much more easily. One such method is called the method of undetermined coefficients.

Example 9.5 Let $f : [0,1] \to \mathbb{R}$ be a continuous function. Construct a linear functional of the form of the form

$$R(f) = c_1 f(0) + c_2 f(1)$$

which is exact for all polynomials of degree at most 1.

Solution Our target is the linear functional $T : C([0,1], \mathbb{R}) \to \mathbb{R}$ given by

$$I(f) = \int_0^1 f(x)dx.$$

The error $E : C([0,1], \mathbb{R}) \to \mathbb{R}$ is the linear functional given by

$$E(f) = I(f) - R(f).$$

We want to find c_1 and c_2 such that $E(p) = 0$ for all polynomials p of degree at most 1. A polynomial of degree at most 1 can be written as $p(x) = ax + b$. The error satisifies

$$E(p) = aE(x) + bE(1).$$

It follows that $E(p) = 0$ for all polynomials of degree at most 1 if and only if $E(1) = 0$ and $E(x) = 0$. We have

$$\begin{aligned} E(1) &= I(1) - R(1) = 1 - (c_1 + c_2), \\ E(x) &= I(x) - R(x) = \frac{1}{2} - c_2. \end{aligned}$$

It follows, that $E(1) = E(x) = 0$ if and only if $c_1 = c_2 = \frac{1}{2}$. In this case

$$R(f) = \frac{1}{2} f(0) + \frac{1}{2} f(1) = \frac{1}{2}\Big[f(0) + f(1)\Big]$$

which we recognizes as the trapezoidal rule for the interval $[0,1]$. ■

Exercises

1. Let $f : [0,1] \to \mathbb{R}$ be a differentiable function. Show that the rule

$$R(f) = \frac{1}{2}\Big[f(0) + f(1)\Big] + \frac{1}{12}\Big[f'(1) - f'(1)\Big]$$

is exact for all polynomials of degree at most 3.

2. Let $f : [0, 1] \to \mathbb{R}$ be a differentiable function. Find constants c_i such that the rule
$$R(f) = c_1 f(0) + c_2 f(\tfrac{1}{2}) + c_3 f(1) + c_4 f'(\tfrac{1}{2})$$
is exact for all polynomials of degree at most 3.
3. Analysis of the midpoint method
4. Analysis of the left endpoint method
5. Analysis of the right end point method
6. Problems with unknown weights
7. Problems with unknown evaluation points
8. Problems with unknonw weights and evaluations points

9.4 Simpson's rule

Simpson's rule is the linear functional $S : C([a, b], \mathbb{R}) \to \mathbb{R}$ given by
$$S(f) = \frac{b-a}{6}\Big[f(a) + 4f(m) + f(b)\Big],$$
where
$$m = \frac{a+b}{2}$$
denotes the midpoint between a and b.

Example 9.6 Use Simpson's rule to approximate the integral $\int_0^1 e^x dx$ and compute the error.

Solution Let $f : [0, 1] \to \mathbb{R}$ be given by $f(x) = e^x$. The target value is
$$I(f) = \int_0^1 \exp(x)dx = e - 1 \approx 1.7183.$$
The value of Simpson's rule is
$$S(f) = \frac{1}{6}\Big[f(0) + 4f(\tfrac{1}{2}) + f(1)\Big] = \frac{1}{6}\Big[1 + 4\sqrt{e} + e\Big] \approx 1.7189.$$
It follows that the error is
$$I(f) - S(f) = \frac{5}{6}e - \frac{2}{3}\sqrt{e} - \frac{7}{6} \approx -5.7932 \times 10^{-4}.$$
■

We can derive Simpson's rule by constructing a rule which is exact for all polynomials of degree at most 2.

Example 9.7 Let $c_i \in \mathbb{R}$ for $i = 1, 2, 3$ and consider the rule $A : C([0, 1], \mathbb{R}) \to \mathbb{R}$ given by
$$A(f) = c_1 f(0) + c_2 f(\tfrac{1}{2}) + c_3 f(1).$$
Show that A is exact for all polynomials of degree at most 2 if and only if
$$c_1 = \frac{1}{6}, \quad c_2 = \frac{2}{3}, \quad c_3 = \frac{1}{6}.$$

Solution The rule A is a linear functional. Therefore, it is exact for all polynomials of degree at most 2 if and only if it is exact for the 3 polynomials 1, x, and x^2. We have 3 equations for c_1, c_2 and c_3, namely

$$\begin{bmatrix} \int_0^1 dx \\ \int_0^1 x dx \\ \int_0^1 x^2 dx \end{bmatrix} = \begin{bmatrix} 1 \\ \frac{1}{2} \\ \frac{1}{3} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 0 & \frac{1}{2} & 1 \\ 0 & \frac{1}{4} & 1 \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix}. \tag{9.6}$$

It is easily verified that the only solution is

$$c_1 = \frac{1}{6}, \quad c_2 = \frac{2}{3}, \quad c_3 = \frac{1}{6}.$$

We see that

$$A(f) = \frac{1}{6}f(0) + \frac{2}{3}f(\tfrac{1}{2}) + \frac{1}{6}f(1) = \frac{1}{6}\Big[f(0) + 4f(\tfrac{1}{2}) + f(1)\Big]$$

is identical to Simpson's rule for the interval $[0, 1]$. ■

By design, Simpson's rule is exact for all polynomials of degree at most 2. However, Simpson's rule is also exact for all polynomials of degree 3.

Example 9.8 Consider the interval $[0, 1]$. Show that Simpson's rule is exact for all polynomials of degree at most 3.

Solution It is enough to show that Simpson's rule is exact for the polynomial 1, x, x^2 and x^3. We already know that Simpson's rule is exact for the polynomials 1, x, and x^2. The target value is

$$\int_0^1 x^3 dx = \frac{1}{4}.$$

Simpson's rule is

$$S = \frac{1}{6}\Big[f(0) + 4f(\tfrac{1}{2}) + f(1)\Big] = \frac{1}{6}\left[0 + \frac{4}{8} + 1\right] = \frac{1}{4}.$$

We conclude that Simpson's rule is exact for all polynomials of degree at most 3. ■

A composite Simpson's rule can be constructed as follows.

Definition 9.2 *Let $f : [a, b] \to \mathbb{R}$ be a continuous function. Let $n = 2k$ be a even integer, let $h = 1/n$ and let $x_j = a + jh$ for $j = 0, 1, 2, \ldots, n$. Then the composite Simpson's rule S_h corresponding to the uniform stepsize h is given by*

$$S_h = \sum_{j=0}^{k-1} \frac{h}{3}\Big[f(x_{2j}) + 4f(x_{2j+1}) + f(x_{2j+2})\Big]. \tag{9.7}$$

We now seek to analyse the error, i.e. the difference between the integral and the composite Simpson's rule. We begin by studing Simpson's rule for the interval $[-h, h]$. Here the target value is

$$I(f) = \int_{-h}^{h} f(t)dt$$

and the corresponding Simpson's rule is

$$S(f) = \frac{h}{3}\Big[f(-h) + 2f(0) + f(h)\Big].$$

Let F denote any anti-derivative of f and let $g : [0, h] \to \mathbb{R}$ be given by

$$g(x) = F(x) - F(-x) - \frac{x}{3}\Big[f(x) + 2f(0) + f(-x)\Big]. \tag{9.8}$$

This function is relevant because the error because

$$g(h) = I(f) - T(f).$$

It is the properties of g that will allow us to make precise statements about the error. We begin by deriving an error formula.

Theorem 9.3 *Let $h > 0$ and let $f \in C^4([-h, h], \mathbb{R})$ and let $g : [0, h] \to \mathbb{R}$ be given by* (9.8)*. Then there exists $\xi \in (-h, h)$ such that*

$$g(h) = -\frac{1}{90} f^{(4)}(\xi)h^5. \tag{9.9}$$

Proof. Let g denote the auxiliary function given by equation (9.8). It is clear that g is as differentiable as f. It is straightforward to verify that

$$\begin{aligned} g'(x) &= \frac{2}{3}\Big(f(x) - 2f(0) + f(-x)\Big) - \frac{x}{3}\Big(f'(x) - f'(-x)\Big), \\ g''(x) &= \frac{1}{3}\Big(f'(x) - f'(-x)\Big) - \frac{x}{3}\Big(f''(x) + f''(-x)\Big), \\ g^{(3)}(x) &= -\frac{x}{3}\Big(f^{(3)}(x) - f^{(3)}(-x)\Big). \end{aligned}$$

We will now systematically exploit the fact that

$$g(0) = g'(0) = g''(0) = 0.$$

Since $g(0) = 0$, the extended mean value theorem implies there exists $x_1 \in (0, h)$ such that

$$\frac{g(h)}{h^5} = \frac{g(x) - g(0)}{x^5 - 0^5} = \frac{g'(x_1)}{5x_1^4}.$$

Since $g'(0) = 0$, the extended mean value theorem implies there exists $x_2 \in (0, x_1)$ such that

$$\frac{g'(x_1)}{5x_1^4} = \frac{g'(x_1) - g'(0)}{5x_1^4 - 5 \cdot 0^4} = \frac{g''(x_2)}{20x_2^3}.$$

Since $g''(0) = 0$, the extended mean value theorem implies there exists $x_3 \in (0, x_2)$ such that

$$\frac{g''(x_2)}{20x_2^3} = \frac{g''(x_2) - g''(0)}{20x_2^3 - 20 \cdot 0^3} = \frac{g^{(3)}(x_3)}{60x_3^2}.$$

We conclude that

$$\frac{g(h)}{h^5} = -\frac{1}{90}\frac{f^{(3)}(x_3) - f^{(3)}(x_3)}{2x_3}.$$

Now since $f \in C^4$, the mean value theorem implies there exists $x_4 \in (-x_3, x_3)$ such that

$$\frac{f^{(3)}(x_3) - f^{(3)}(x_3)}{2x_3} = f^{(4)}(x_4).$$

We conclude that

$$g(h) = -\frac{1}{90} f^{(4)}(\xi)h^5$$

for at least one $\xi \in (-h, h)$, namely $\xi = x_4$. This completes the proof. ■

If p is a polynomial of degree at most 3, then $p^{(4)}(x) = 0$ for all x. Theorem thm:simpson:error-formula implies that Simpson's rule is exact for all polynomials of degree at most 3. This is consistent with Example 9.8.

We can now derive an error formula for the composite Simpson's rule.

Theorem 9.4 *Let $f \in C^4([a,b],\mathbb{R})$ and let S_h denote the composite Simpson's rule corresponding to the uniform stepsize $h > 0$. Then there exists $\xi \in [a,b]$ such that the error $I(f) - S_h(f)$ satisfies*

$$I(f) - S_h(f) = -\frac{1}{180}(b-a)f^{(4)}(\xi)h^4.$$

Theorem 9.3 gives no indication of the exact location of $\xi \in [a,b]$.

Proof. For each subinterval $[x_{2j}, x_{2j+2}]$ there exists by Theorem 9.3 a $\xi_j \in (x_{2j}, x_{2j+1})$ such that

$$\int_{x_{2j}}^{x_{2j+1}} f(t)dt - \frac{h}{3}\Big[f(x_{2j} + 4f(x_{2j+1}) + f(x_{2j})\Big] = -\frac{1}{90}f^{(4)}(\xi_j)h^5$$

It follows that

$$T - S_h = \sum_{j=0}^{k-1} -\frac{1}{90}f^{(4)}(\xi_j)h^5 = -\frac{1}{90}h^5\sum_{j=0}^{k-1} f^{(4)}(\xi_j).$$

Now by the intermediate value theorem applied to the continuous function $f^{(4)}$ there exists ξ such that

$$\frac{1}{k}\sum_{j=0}^{k-1} f^{(4)}(\xi_j) = f^{(4)}(\xi).$$

We conclude that

$$T - S_h = -\frac{1}{90}kh^5 f^{(4)}(\xi) = -\frac{1}{180}(2kh)h^4 f^{(4)}(\xi) = -\frac{1}{180}(b-a)h^4 f^{(4)}(\xi).$$

This completes the proof. ■

Problems

1. Let $h > 0$, let $f \in C^{(3)}([-h,h])$ and let $g : [0,h] \to \mathbb{R}$ be given by

$$g(x) = \int_{-x}^{x} f(t)dt - \frac{x}{3}\Big[f(-x) + 4f(0) + f(x)\Big].$$

   (a) Show that the error of Simpson's rule satisfies

$$g(h) = I(f) - S(f).$$

   (b) Show that

$$|g(h)| \le \frac{1}{36}\|f^{(3)}\|_\infty h^4.$$

   where

$$\|f^{(3)}\|_\infty = \max\{|f^{(3)}(x)| \,:\, x \in [-h,h]\}.$$

   Hint: Apply the extended mean value theorem repeatedly.

2. Let $-\infty < a < b < \infty$, let $f \in C^3([a,b],\mathbb{R})$ and let S_h denote the composite trapezoidal rule corresponding to the uniform stepsize h. Show that

$$|I(f) - S_h(f)| \le \frac{1}{72}(b-a)\|f^{(3)}\|_\infty h^3.$$

   where

$$\|f^{(3)}\|_\infty = \max\{|f^{(3)}(x)| \,:\, x \in [a,b]\}.$$

Computer problems

1. Verify Theorem 9.3 in the case of $f : [-1, 1] \to \mathbb{R}$ given by $f(x) = xe^x$. You will have to accurately solve the nonlinear equation

$$\int_0^1 te^t dt - \frac{1}{3}\Big[f(-1) + 4f(0) + f(1)\Big] = -\frac{1}{90}f^{(4)}(\xi)$$

for $\xi \in (-1, 1)$.

9.5 Practial error estimation

Let $f : [a, b] \to \mathbb{R}$ be continuous and consider the problem of computing $\int_a^b f(x)dx$. The composite trapezoidal rule $T_h(f)$ and the composite Simpson's rule $S_h(f)$ can both be viewed as functions of the stepsize h.

Example 9.9 Let $T = \int_0^1 x^3 dx$ and let A_h denote the value of the composite trapezoidal rule corresponding to the stepsize $h > 0$. Show that the error $T - A_h$ obeys an asymptotic error expansion.

Solution Let n be a positive integer, $h = \frac{1}{n} > 0$ and let $x_j = jh$ for $j = 0, 1, 2, \ldots, n$. Then the composite trapezoidal rule A_h is given by

$$A_h = \sum_{j=0}^{n-1} \frac{1}{2}h\Big(x_j^3 + x_{j+1}^3\Big) = \frac{1}{2}h^4 \sum_{j=0}^{n-1}\Big(j^3 + (j+1)^3\Big).$$

In general, a sum of cubes can be computed using

$$\sum_{k=1}^{m} k^3 = \left(\frac{m(m+1)}{2}\right)^2.$$

It follows, that

$$A_h = \frac{1}{2}h^4\left[\left(\frac{(n-1)n}{2}\right)^2 + \left(\frac{n(n+1)}{2}\right)^2\right] = \frac{1}{4}h^4(n^4 + n^2) = \frac{1}{4}(nh)^4 + \frac{1}{4}(nh)^2h^2$$

Since $nh = 1$ we can conclude that

$$T - A_h = -\frac{1}{4}h^2$$

In this case, the asymptotic error expansion is very simple, but this so is the integrand. ■

Example 9.10 Apply the composite trapoizal rule with step h and approximate the integral $T = \int_0^1 \exp(x)dx$. Show that the error $T - A_h$ obeys an asymptotic error expansion.

Solution Let n be a positive integer and let $h = \frac{1}{n}$. The corresponding value of the composite trapezodal rule is given by

$$A_h = \frac{1}{2}h \sum_{j=0}^{n-1} \exp(jh) + \exp((j+1)h) = \frac{1}{2}h(1 + \exp(h)) \sum)j = 0^{n-1} \exp(jh)$$

■

In general, we have the following result.

Theorem 9.5 *Let $f \in C^{(2k+1)}([a,b],\mathbb{R})$ and let T_h denote the trapezoidal rule corresponding to the uniform stepsize $h > 0$. Then the error satifies an asymptotic error expansion of the form*

$$T - T_h = \alpha_1 h^2 + \alpha_2 h^4 + \ldots \alpha_h h^{2k} + O(h^{2k+1}), \quad h \to 0, \tag{9.10}$$

where

$$\alpha_k = -\frac{B_{2k}}{(2k)!}\Big[f^{2k-1}(b) - f^{(2k-1)}(a)\Big] \tag{9.11}$$

Simpson's rule is closely related to the trapezoidal rule. In fact, Simpson's rule can be obtained by adding Richardson's error estimate to the trapezoidal rule. To see this we consider the interval $[-h,h]$. Let T_h denote the composite trapezoidal rule obtained using stepsize h. Then

$$T_h = \frac{1}{2}h\Big[f(-h) + f(0)\Big] + \frac{1}{2}h\Big[f(0) + f(h)\Big] = \frac{1}{2}hf(-h) + hf(0) + \frac{1}{2}hf(h).$$

The trapzoidal rule corresponding to the stepsize $2h$ is simply

$$T_{2h} = \frac{1}{2}(2h)\Big[f(-h) + f(h)\Big] = hf(h) + hf(-h).$$

It follows that Richardson's error estimate is

$$E_h^{\text{est}} = \frac{T_h - T_{2h}}{3} = h\frac{-\frac{1}{2}f(-h) + f(0) - \frac{1}{2}f(h)}{3} = -h\frac{f(-h) - 2f(0) + f(h)}{6}$$

Adding Richardson's error estimate to the trapozidal rule yields

$$T_h + \frac{T_h - T_{2h}}{3} = \frac{1}{3}hf(-h) + \frac{4}{3}hf(0) + \frac{1}{3}hf(h).$$

This is precisely the expression for Simpson's rule corresponding to the interval $[-h,h]$.

1. Asymptotic error expansion

2. Error estimates

Exercises

1. Let $I = \int_0^1 x^5 dx$ and let T_h denote the composite trapezoidal rule corresponding to the stepsize $h > 0$. Find the asymptotic error expansion of $I - T_h$.

Computer problems

Chapter 10

Numerical solution of ordinary differential equations

10.1 Examples

10.2 Grids and grid functions

10.3 Elementary methods

10.3.1 Euler's explicit method

10.3.2 Euler's implicit method

10.3.3 The trapezoidal rule

10.3.4 Heun's method

10.3.5 The classical fourth order Runge-Kutta method

10.4 Practical error estimation

10.5 Event location

Exercises

Computer problems