Numerical Methods for Nonlinear Equations

Václav Kučera

Prague, 2022

EUROPEAN UNION
European Structural and Investment Funds
Operational Programme Research,
Development and Education

MINISTRY OF EDUCATION,
YOUTH AND SPORTS

To Monika and Terezka
for their endless love and support

These materials have been produced within and supported by the project "Increasing the quality of education at Charles University and its relevance to the needs of the labor market" kept under number CZ.02.2.69/0.0/0.0/16_015/0002362.

Contents

Chapter 1

Nonlinear equations

1.1 Introduction

These lecture notes represent a brief introduction to the topic of numerical methods for nonlinear equations. Sometimes the term 'nonlinear algebraic equations' can be found in the literature, which evokes polynomial equations. This is however not the case, the term 'algebraic' is used to distinguish our equations from nonlinear *differential* equations, which is a completely different world. In our case, we shall look for ***roots*** of a given function f, which will be a general, at least continuous function. Here 'roots' means points where the function f attains the value zero.

From the historical perspective, originally the focus was on finding roots, ideally of polynomial equations of higher and higher degree. However, in the past century, the focus shifted from roots to fixed points. From the point of view of modern mathematics, there is nothing special about a root, it is just a point, where f attains a certain value, zero, which we consider 'special', but we might as well have chosen this special value as 1 or π or -17. On the other hand a fixed point is a topological invariant and therefore many powerful tools of modern mathematics can be used to prove nonconstructive existence of fixed points. As we shall see later on, from the viewpoint of numerical methods for nonlinear (algebraic) equations, fixed points are the natural perspective.

If we want to look more closely at the history of root finding, polynomial equations are very interesting. Formulas for the roots of general quadratic, cubic and quartic equations were known by the first half of the 16th century. Then the race to find similar formulas for the quintic (5th degree) equation began. Three centuries later, the Abel-Ruffini theorem was proved in 1824, which states that the roots of a general quintic equation cannot be expressed in terms of *radicals*, i.e. finite combinations of $+, -, *, /, \sqrt[n]{\cdot}$. Galois theory is the modern viewpoint and one can show e.g. that the real root of $x^5 - x - 1 = 0$ is not expressible using radicals over the rational numbers. However, the key here is to define exactly what is meant by 'formula for the roots'. Abel and Ruffini say that radicals are hopeless for quintic and higher equations. Radicals rely on our ability to extract the n-th root, i.e. for $n = 5$ solve the auxiliary equation $x^5 - a = 0$ for a given a. However, if one admits the use of so-called ultraradicals (Bring radicals), where one solves the modified equation $x^5 + x - a = 0$ for a given a, then one can solve any quintic equation. Another

approach to obtain formulas for general quintic roots is through the use of elliptic modular functions. Felix Klein gives an elegant solution using the symmetry group of the icosahedron. Later others came and solved the case of general equations of degree 7, then 11, etc. using more and more special functions. The final chapter was written by Hiroshi Umemura in 1984, when he published formulas for roots of general polynomial equations of *arbitrary* degree. If he had done so in 1850, he would have been celebrated as one of the greatest mathematicians of all time. But today his result is a practically unknown curiosity sometimes mentioned in a footnote. The reason is that he expresses the roots using so-called Siegel modular functions, which are defines as infinite series of matrix exponentials. Nobody knows how to practically evaluate or even reasonably approximate these functions. So from the point of view of practice, the formulas are completely useless. Also from the point of view of theory, they bring nothing new. The idea that you can solve any problem analytically if you define more and more complicated special functions with special properties that nobody knows how to evaluate belongs in the 19th century. Modern mathematics uses other tools to show existence, uniqueness, desired properties, etc. indirectly or non-constructively. If we want to know what the value is numerically, we turn to numerical mathematics instead of formulas.

In this textbook, we shall describe and analyze the basic numerical methods for nonlinear equations and their systems. We will spend a lot of time in the seemingly simple case of 1D, i.e. scalar nonlinear equations. The reason is that the mathematical tools and the proofs are more intuitive and can often be demonstrated using pictures. This is not the case of systems of equations, where the theory becomes much more technical. Bear in mind that the literature concerning numerics for nonlinear equations os huge, with hundreds of methods being developed and analyzed. Here we barely scratch the surface.

1.2 Mathematical background

We will consider a function $f : \mathbb{R} \to \mathbb{R}$ (or perhaps f might be defined only on an interval) and we seek $x_* \in \mathbb{R}$ which is the **root** of f, i.e. $f(x_*)$=0. This is the scalar case. For systems of equations, we will consider $F : \mathbb{R}^N \to \mathbb{R}^N$ and its root $x_* \in \mathbb{R}^N$, i.e. $F(x_*) = 0$.

We will be dealing with iterative methods for root finding. These construct a sequence $x_n, n \in \mathbb{N}_0$, which (ideally) converges to x_*. We can usually write the considered numerical method as $x_{n+1} = G(x_n)$, where $G : \mathbb{R}^N \to \mathbb{R}^N$ is somehow derived from F. Assuming that G is continuous and that the method works ($x_n \to x_*$) we have

$$\begin{array}{ccc} x_{n+1} & = & G(x_n) \\ \downarrow & & \downarrow \\ x_* & = & G(x_*), \end{array}$$

therefore x_*, which is a root of F, is a **fixed point** of G. Therefore, in our context, it is more natural to look at fixed points of mappings rather than roots. This was also the general trend in mathematics and highlights one of the differences between 19th

century and 20th century mathematics – while roots and fixed points are fundamentally connected, the latter is a more natural notion from the topological viewpoint, therefore topological tools can be used in proofs, etc. Fixed point theorems are one of the basic tools in mathematical analysis.

For completeness, we shall state several basic theorems from mathematical analysis concerning the existence of fixed points or roots.

Theorem 1 (Banach fixed point theorem)**.** *Let X be a complete metric space and let $G : X \to X$ be a* **contraction***, i.e. there exists $\alpha \in [0, 1)$ s.t. $\|G(x) - G(y)\| \leq \alpha\|x - y\|$ for all $x, y \in X$. Then there exists a unique fixed point $x_* \in X$ of G. Moreover, let the sequence $x_n \in X, n \in \mathbb{N}_0$ be defined by $x_{n+1} = G(x_n)$ for arbitrary $x_0 \in X$. Then $x_n \to x_*$ and we have the estimate*

$$\|x_n - x_*\| \leq \frac{\alpha^n}{1-\alpha}\|x_1 - x_0\|. \tag{1.1}$$

Proof. We proceed in several steps:
1. The entire sequence $\{x_n\}_{n=0}^{\infty}$ is well defined, since $x_0 \in X$ and G maps X to X.
2. For arbitrary $m \in \mathbb{N}$, we can estimate by induction

$$\|x_m - x_{m-1}\| = \|G(x_{m-1}) - G(x_{m-2})\| \leq \alpha\|x_{m-1} - x_{m-2}\| \leq \ldots \leq \alpha^{m-1}\|x_1 - x_0\|.$$

3. $\{x_n\}_{n=0}^{\infty}$ is a Cauchy sequence: Let $m \geq n$, then by the previous estimate

$$\begin{aligned}\|x_m - x_n\| &\leq \|x_m - x_{m-1}\| + \|x_{m-1} - x_{m-2}\| + \ldots + \|x_{n+1} - x_n\| \\ &\leq \alpha^{m-1}\|x_1 - x_0\| + \alpha^{m-2}\|x_1 - x_0\| + \alpha^n\|x_1 - x_0\| \\ &= \alpha^n\|x_1 - x_0\| \sum_{k=0}^{m-n-1} \alpha^k \leq \alpha^n\|x_1 - x_0\| \sum_{k=0}^{\infty} \alpha^k \\ &= \frac{\alpha^n}{1-\alpha}\|x_1 - x_0\| \longrightarrow 0, \quad \text{as } n \to \infty,\end{aligned} \tag{1.2}$$

   since $\alpha \in [0, 1)$.
4. X is complete, therefore $\{x_n\}_{n=0}^{\infty}$ has a limit x_*. Since G is continuous, we have

$$0 \leq \|G(x_*) - x_*\| = \lim_{n\to\infty} \|G(x_n) - x_n\| \leq \lim_{n\to\infty} \alpha^{n-1}\|x_1 - x_0\| = 0,$$

   hence $G(x_*) = x_*$ and x_* is a fixed point of G.
5. To prove estimate (1.1), we simply take the limit as $m \to \infty$ in (1.2), noting that the right-hand side is independent of m and that $x_m \to x_*$ in the left-hand side.

□

Theorem 1 is a perfect theorem from the viewpoint of numerical analysis. It gives existence and uniqueness of the exact solution, constructs an iterative numerical method, proves convergence of the method to the exact solution and provides an apriori error estimate.

We note that the error estimate indicates that the closer α is to zero, the faster convergence of x_n we can expect. This will be an important observation in the context of Newton's method. In general, it is not easy to straightforwardly prove that a mapping is a contraction. For differentiable functions in 1D, we can use the following simple observation.

Lemma 2. *Let $I \subset \mathbb{R}$ be a closed interval. Let $g \in C^1(I)$ and let there exist $\alpha \in [0,1)$ such that $|g'(x)| \leq \alpha$ for all $x \in I$. Then g is a contraction on I.*

Proof. Let $x, y \in I$ be arbitrary. By the mean value theorem, there exists $\xi \in I$ such that

$$|g(x) - g(y)| = |g'(\xi)||x - y| \leq \alpha|x - y|.$$

□

Provided g is continuously differentiable, it is sufficient to have $|g'(x_*)| \leq 1$ to be a contraction on some neighborhood of x_*. This ensures that the iterative process from Banach's fixed point theorem converges for x_n sufficiently close to x_*. This is called ***local convergence*** and is usually the best we can generally hope to prove for our considered numerical methods.

Corollary 3. *Let g be continuously differentiable on some neighborhood U of x_*. Let $|g'(x_*)| \leq 1$. Then there exists a closed neighborhood V of x_* such that g is a contraction on V.*

The situation in $\mathbb{R}^N$ is more complicated, however Corollary 3 can be generalized – This is Ostrowski's theorem 31.

Contractivity is a very useful, yet rare property. The following result is one of the more general fixed point results.

Theorem 4 (Brouwer's fixed point theorem)**.** *Let $B \subset \mathbb{R}^N$ be a closed ball. Let $F : B \to B$ be continuous. Then F has a fixed point.*

Classical proofs of Brouwer's theorem are extremely nonconstructive. This is ironic, because Luitzen Egbertus Jan Brouwer was a strong figure in the constructivist school of mathematics which believed that only constructive mathematics is valid and rejected things like proof by contradiction. There exist modern proofs that are constructive (and relatively simple), but not as explicitly and simply constructive as Banach's fixed point theorem.

We note that the ball B in Theorem 4 can be replaced by any convex compact set (or a set homeomorphic to such a set). The assumption of convexity and compactness is also sufficient in general Banach spaces (Schauder fixed point theorem). Unlike Banach's theorem, the fixed point is in general not unique for Brouwer.

In terms of roots, not fixed points, one can prove for example the following general results.

Theorem 5. *Let $B_R = \{x \in \mathbb{R}^N; \|x\| \leq R\}$ and let $F : B_R \to \mathbb{R}^N$ be continuous. Assume $F(x) \cdot x > 0$ for all $x : \|x\| = R$. Then there exists a root x_* of F in B_R.*

Theorem 6. *Let $F : \mathbb{R}^N \to \mathbb{R}^N$ be continuous and let*

$$\lim_{\|x\| \to \infty} \frac{F(x) \cdot x}{\|x\|} = \infty.$$

Then for all $y \in \mathbb{R}^N$ there exists a solution x of $F(x) = y$.

Theorem 7 (Zarantonello)**.** *Let F be a continuous mapping from the Hilbert space H to itself such that F maps bounded sets to bounded sets. Let there exist $c > 0$ such that $(F(x) - F(y), x - y) \geq c\|x - y\|^2$ for all $x, y \in H$. Then for all $y \in H$ there exists a unique solution x of $F(x) = y$. Moreover x depends continuously on y.*

We note that the existing literature is full of such results, which have various applications, typically in nonlinear partial differential equations. However, since nonlinearities are difficult, there are no universal results covering all possible situations.

Exercises

Exercise 1 (Going from roots to fixed points).

Let $f(x) = x^2 - x - 2$. We wish to solve the equation $f(x_*) = 0$ by reformulating it as a fixed point problem $g(x_*) = x_*$ which we will solve using the simple iteration procedure from Theorem 1. There are many possible choice of g derived from the equation $f(x) = 0$ to obtain $g(x) = x$ by various manipulations:

1. $g(x) = x^2 - 2$
2. $g(x) = \sqrt{x + 2}$
3. $g(x) = 1 + 2/x$
4. $g(x) = \dfrac{x^2 + 2}{2x - 1}$

Check that these choices of g really correspond to the equation $f(x) = 0$. Try each possibility whether or not the iterative process $x_{n+1} = g(x_n)$ converges and justify the results by checking whether or not g is a contraction in each case.

1.3 Rate of convergence

We will be concerned with iterative numerical methods for the solution of nonlinear equations. Such methods generate a sequence of approximations $\{x_n\}_{n=0}^{\infty} \subset \mathbb{R}^N$ which (ideally) converges to the true solution x_*. In order to compare these methods among themselves, it is useful to measure how fast such sequences converge.

Definition 8 (Rate of convergence)**.** *Let $\{x_n\}_{n=0}^{\infty} \subset \mathbb{R}^N$ be a given sequence which converges to x_*. We say that the sequence has convergence* **rate** *(or* **order***) $p \geq 1$ if there exists $C > 0$ such that*

$$\lim_{n\to\infty} \frac{\|x_{n+1} - x_*\|}{\|x_n - x_*\|^p} = C. \tag{1.3}$$

We say that a numerical method has order p, if all sequences generated by the method have convergence rate at least p for the considered problem and set of initial conditions, and at least one such sequence has convergence rate exactly p.

If we denote the error of the method at the current iteration as $e_n = x_* - x_n$, the relation (1.3) states that (in the limit) we expect the next error to be approximately $\|e_{n+1}\| \approx C\|e_n\|^p$. Obviously, under ideal circumstances it is desired to have methods with higher p.

Definition 9. *We distinguish several basic cases of convergence rate:*

- $p = 1, C \in (0, 1)$ – ***Linear*** *convergence,*
- $p = 1, C = 0$, *or* $p > 1$ – ***Superlinear*** *convergence,*
- $p = 2$ – ***Quadratic*** *convergence.*

Obviously, the list could go on – superquadratic, cubic, etc. We will have no need for such terminology. We note that Definition 8 is a simplified version of the so-called ***Q-factor*** (quotient factor), which defines the rate of convergence as

$$\inf_{p \geq 1} \left(\sup_{\substack{\{x_n\}_{n \in \mathbb{N}_0} \\ x_n \to x_*}} \limsup_{n \in \mathbb{N}_0} \frac{\|x_{n+1} - x_*\|}{\|x_n - x_*\|^p} = \infty \right), \tag{1.4}$$

where the supremum is taken over all sequences generated by the method. This rather technical definition avoids the subtle issue with (1.3) concerning special cases, where there does not exist a p such that the limit in (1.3) is a nonzero finite constant C. This is similar to the case of superlinear convergence, as defined above, where $C = 0$ for $p = 1$. We do not wish to go too deeply into this subject, as for example [OR70], where a whole chapter is devoted to the study of (1.4) and its relation to a similar concept – the R-factor (root factor).

We conclude with several remarks:

- Roughly speaking, a linearly convergent method adds a constant number of correct digits to the result. Let $|x_n - x_*| \approx 10^{-l}$. Then in the limit, we expect $|x_{n+1} - x_*| \approx C.10^{-l}$. For example, if $C = 0.1$, we expect $|x_{n+1} - x_*| \approx C.10^{-(l+1)}$. More generally, in each iteration we expect to obtain $-\log_K(C)$ digits of the correct result in number base K.

- On the other hand, a quadratically convergent method will approximately **double** the number of correct digits in each iteration. Since if $|x_n - x_*| \approx 10^{-l}$, we expect $|x_{n+1} - x_*| \approx C.10^{-2l}$. Here the role of the constant C becomes marginal with respect to the squaring of the previous error. Once a quadratically convergent method (e.g. Newton's method) really starts converging, we reach very high precision in only a few iterations. The problem with these methods is the so-called pre-asymptotic phase, where this doubling is not yet present and it might take a while (or perhaps never), before the method actually starts converging quadratically.

  In general a method with convergence rate $p > 1$ multiplies the number of correct digits by the factor p in each iteration (at least in the limit).

- For more complicated methods, it is sometimes hard to prove the existence of the limit in the form of equality (1.3), one usually proceeds by estimating the errors rather than proving a series of equalities. That is why many authors define convergence rate p with an inequality in (1.3), i.e. $\lim \ldots \leq C$. Strictly speaking one is then proving that the method has convergence rate *at least* p and should also provide a lower bound, at least for some simple example, to show that the convergence rate is *exactly* p.

- Defintion 8 only takes into account how much the error decreases from iteration to iteration. It does not take into account how expensive or cheap each iteration is. We will address this question when comparing Newton's method with the secant method on page 38. Newton's method is quadratically convergent, while the secant method has convergence rate approximately 1.618. From the viewpoint of Definition 8, Newton is the clear winner. However each iteration is twice as expensive as in the secant method (in terms of function evaluations). Taking this into account, it is not that clear which of the methods is the winner.

Chapter 2

Scalar equations

In this chapter, we will be dealing with scalar nonlinear equations. We will consider the equation $f(x) = 0$ with the root x_*, where f will be at least continuous on some interval containing x_*. We will distinguish between three basic types of methods:

- **Open methods.** These methods construct a sequence $\{x_n\}_{n=0}^{\infty}$ of approximations which should converge to x_*. Examples include the Newton and secant methods. In general they converge only locally (x_0 must be close enough to x_* for convergence), however they can achieve higher convergence rates and are thus faster (provided they converge).

- **Bracketing methods.** Instead of a sequence of approximations x_n, we will construct a sequence of intervals $\{I_n\}_{n=0}^{\infty}$, where $x_* \in I_n$ for all n. Examples of these methods include the bisection and false position (regula falsi) methods. They are typically very simple and slow, but converge (almost) always. The mathematical justification for bracketing is that the exact solution x_* might not be exactly representable in finite precision arithmetic. Thus it makes sense to look for a small interval containing x_* instead of a single approximation.

- **Hybrid methods.** These methods use combinations of open and bracketing methods to obtain more robust (perhaps even globally convergent) methods, which may exhibit high convergence rates under ideal circumstances. Sophisticated criteria are used to switch between several methods in each iteration to choose the best and/or safest next iteration. Dekker's method and Brent's method are examples of hybrid approaches.

We note that there is an entire field of research concerning **globalization** strategies, the aim of which is to make locally convergent methods globally convergent. We will demonstrate one such approach in Section 3.5. However, as usual with nonlinearities, there are no universal recipes – methods that always work.

2.1 Bisection method

Bisection is the simplest bracketing method. It is a straightforward application of the intermediate value theorem – a continuous function attains all values between

its values at the endpoints of an interval. Therefore, if $f(a)f(b) \leq 0$ then there exists a root between a and b. —Given a, b we take the midpoint $m = \frac{1}{2}(a+b)$ and based on the sign changes of $f(a), f(b)$ and $f(m)$, we choose either $[a, m]$ or $[m, b]$ as the new smaller interval containing x_*. Written as an algorithm:

Given $I_0 = [a_0, b_0]$ and a tolerance tol. Set $n = 0$.
While $(b_n - a_n) > tol$:
    $m_n = \frac{1}{2}(a_n + b_n)$.
    If $f(a_n)f(m_n) \leq 0$
        $a_{n+1} := a_n,\ b_{n+1} := m_n,$
    else
        $a_{n+1} := m_n,\ b_{n+1} := b_n.$

Bisection is linearly convergent method in the sense of Definition 8, where instead of the errors $|x_{n+1} - x_*|$ and $|x_n - x_*|$ in the definition of the limit, we take $|I_{n+1}|$ and $|I_n|$.

Theorem 10. *The bisection method is* **linearly** *convergent. It is* **globally** *convergent: for any $I_0 = [a_0, b_0]$ such that $f(a_0)f(b_0) \leq 0$, we have convergence, i.e. $a_n \to x_*$, $b_n \to x_*$. In each iteration we gain one bit of information.*

Proof. All of the statements follow immediately from the fact that $|I_{n+1}| = |I_n|/2$. □

We conclude with several remarks:

- It is more efficient to compare the signs of $f(a), f(m)$ in the implementation. It is more efficient than multiplying $f(a)f(m)$ only to test the sign of the product.

- Bisection cannot be directly used to approximate even roots, e.g. to solve $x^2 = 0$ (x_* is the root of f and f'). This is because there are no sign changes near x_* in this case. A simple workaround is the fact that an even root of f is an odd root of f'. Therefore use bisection to find a root of f' and check whether it is also a root of f.

- Starting from an initial interval of length one, we always need 52 iterations in double precision arithmetic to obtain I_n whose endpoints are two neighboring double numbers. This is irrespective of whether f is a 'simple' or 'complicated' function. The method ignores any information about f other than simple sign changes. For example, Newtons method takes into account the derivative of f. Thus Newton's method gives the exact solution of a linear equation in one iteration, while bisection doesn't 'care' whether or not f is linear.

- Surprisingly, there exist generalizations of the bisection method to systems of equations. All one needs is some criterion whether an interval (hypercube or perhaps a simplex) in $\mathbb{R}^N$ contains a root. These can be based on some more or less practically implementable version of the topological index of a mapping. The resulting formulas and mathematical background are very complex. Such methods are tempting, since they would in principle always converge, similarly

as in the scalar case. The question is, whether they are useful. Consider a system of 100 equations for 100 unknowns. You start with an initial hypercube $I_0 \in \mathbb{R}^{100}$, perform bisection and test which one of the smaller cubes contains a root. In $\mathbb{R}$ splitting an interval in half gives two sub-intervals. However, in $\mathbb{R}^{100}$, splitting the edges of a cube in half results in 2^{100} sub-cubes. This is approximately 10^{30} sub-cubes in each iteration, each of which must be tested if it contains a root – just to gain 'one bit of information'. It is clear that such methods can in principle be practical only for very small systems. They are completely out of the question, for example, for finite element solvers for nonlinear partial differential equations where one must solve millions of equations for millions of unknowns.

2.2 Fixed point iteration

Here we will take a closer look at the basic fixed point iteration procedure from Banach's fixed point theorem. As a reminder, instead of $f(x) = 0$ with the root x_*, we consider $g(x) = x$, for which x_* is a fixed point. For a given x_0, we consider the iterative procedure

$$x_{n+1} = g(x_n). \tag{2.1}$$

Theorem 11 (Linear convergence)**.** *Let $g \in C^1(U)$ for some neighborhood U of x_*. Let $0 < |g'(x_*)| < 1$. Then there exists a neighborhood V of x_* such that for all $x_0 \in V$ the sequence defined by (2.1) converges to x_** **linearly**.

Proof. Since $g \in C^1(U)$ and $|g'(x_*)| < 1$, there exists a (closed) neighborhood V of x_* such that $|g'(x)| < 1$ for all $x \in V$. By Banach's fixed point theorem $x_n \to x_*$.

Concerning the linear convergence rate, the mean value theorem gives us

$$x_{n+1} - x_* = g(x_n) - g(x_*) = g'(\xi_n)(x_n - x_*)$$

for some ξ_n between x_n and x_*. Since $x_n \to x_*$, also $\xi_n \to x_*$ and we have

$$\lim_{n\to\infty} \frac{|x_{n+1} - x_*|}{|x_n - x_*|} = \lim_{n\to\infty} |g'(\xi_n)| = |g'(x_*)| > 0,$$

which is linear convergence, by definition (1.3). □

Theorem 12 (Higher order convergence)**.** *Let $p \in \mathbb{N}$ and let $g \in C^p(U)$ for some neighborhood U of x_*. Then the following are equivalent:*

(i) $x_n \to x_*$ with **rate** *p for all x_0 in some neighborhood V of x_*,*
(ii) $g^{(j)}(x_*) = 0$ for $j = 1, \dots, p-1$ and $g^{(p)}(x_*) \neq 0$.

Proof. *(ii)* $\Rightarrow$ *(i)*. Banach's fixed point theorem gives us convergence. By Taylor's expansion, there exists ξ_n between x_n and x_* such that

$$x_{n+1} - x_* = g(x_n) - g(x_*) = \underbrace{\sum_{j=1}^{p-1} \frac{g^{(j)}(x_*)}{j!}(x_n - x_*)^j}_{=0} + \frac{g^{(p)}(\xi_n)}{p!}(x_n - x_*)^p,$$

where the terms in the sum are zero due to the assumption $g^{(j)}(x_*) = 0$ for $j = 1, \dots, p-1$. It follows that

$$\lim_{n\to\infty} \frac{|x_{n+1} - x_*|}{|x_n - x_*|^p} = \lim_{n\to\infty} \frac{|g^{(p)}(\xi_n)|}{p!} = \frac{|g^{(p)}(x_*)|}{p!} > 0,$$

which is convergence rate p, by definition (1.3).

$(i) \Rightarrow (ii)$. We prove the assertion by contradiction. By definition of convergence rate p we have the existence and finiteness of the limit

$$\lim_{n\to\infty} \frac{|x_{n+1} - x_*|}{|x_n - x_*|^p} = C > 0. \tag{2.2}$$

Let there exist some $J \in \{1, \dots, p-1\}$ such that $g^{(J)}(x_*) \neq 0$ and let J be the smallest index with this property. Then by the Taylor expansion, we have

$$x_{n+1} - x_* = g(x_n) - g(x_*) = \underbrace{\sum_{j=1}^{J-1} \frac{g^{(j)}(x_*)}{j!}(x_n - x_*)^j}_{=0} + \frac{g^{(J)}(\xi_n)}{J!}(x_n - x_*)^J,$$

therefore

$$\lim_{n\to\infty} \frac{|x_{n+1} - x_*|}{|x_n - x_*|^p} = \lim_{n\to\infty} \frac{|g^{(J)}(\xi_n)|}{J!} . \frac{|x_n - x_*|^J}{|x_n - x_*|^p} = \infty,$$

since $p > J$, $|x_n - x_*| \to 0$ and $|g^{(J)}(\xi_n)| \to |g^{(J)}(x_*)| > 0$. This is a contradiction with the finiteness of the limit due to (2.2). □

We conclude with several remarks:

- Here we were concerned with *one-point* iterative processes, which means that x_{n+1} depends only on the previous iterate via (2.1). We have seen that given sufficient regularity of g, such iterative processes (methods) can have only integer convergence rates. For *multi-point* methods, where x_{n+1} depends on several previous iterates (e.g. $x_{n+1} = g(x_n, x_{n-1})$), we can have general real convergence rates. We will see this in the *secant method*.

- We can view Newton's method as a "recipe" how to obtain from a given f a new function g such that the fixed point iteration has quadratic convergence rate, i.e. g satisfies $g'(x_*) = 0$, see Exercise 5.

- There exist general "recipes" how to obtain suitable g with arbitrarily high convergence rate p. These are the so-called ***Householder methods***. These methods are however considered as impractical, since they need to evaluate derivatives of f up to order $p-1$ and the complexity of the resulting formulas rapidly grows with growing p.

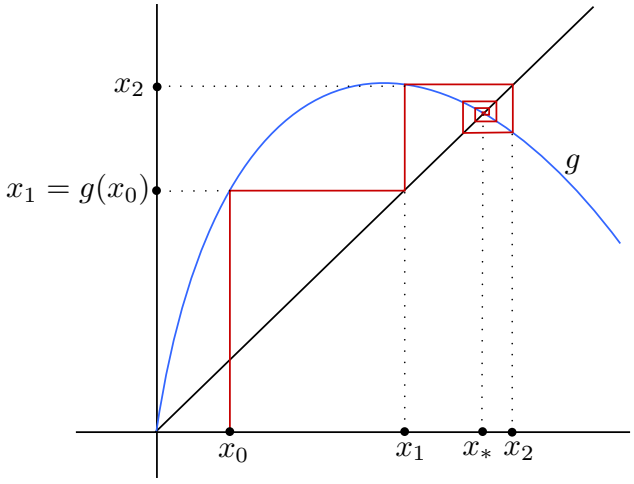

Figure 2.1: Cobweb plot.

2.2.1 Cobweb plots

There is a simple and elegant graphical method, how to gain insight and intuition on the iterative process (2.1) called ***cobweb plots*** or ***Verhulst diagrams*** (after the discoverer, Pierre François Verhulst, a Belgian mathematician from the first half of the 19th century).

The procedure is simple (cf. Figure 2.1). Draw the graph of g and the graph of the identity mapping $y = x$ (i.e. the diagonal). Choose some x_0. Then $x_1 = g(x_0)$ can be visualized as going vertically from x_0 to the graph of g. Next, we want to evaluate $x_2 = g(x_1)$, however x_1 'lives' on the y-axis and in order to evaluate $g(x_1)$, we need to transfer it to the x-axis. Without measuring, this can be done simply by going horizontally from the value x_1 on the y-axis until we intersect the diagonal. After this we go down to the x-axis to obtain the value x_1 on this 'correct' axis. Then we simply iterate the previous procedure:

1) from the current point go vertically to the graph of g,

2) go horizontally to the diagonal.

The iterates x_n can then be read off the x-axis where we go vertically.

Exercises

Exercise 2 (Banach is not sharp).

The assumptions of Banach's fixed point theorem do not need to be satisfied in order to have convergence to a unique fixed point. Take $g(x) = \sin(x)$. Prove that for all $x_0 \in (-\pi/2, \pi/2)$, we have $x_n \to x_* = 0$, even though $g'(x_*) = 1$.

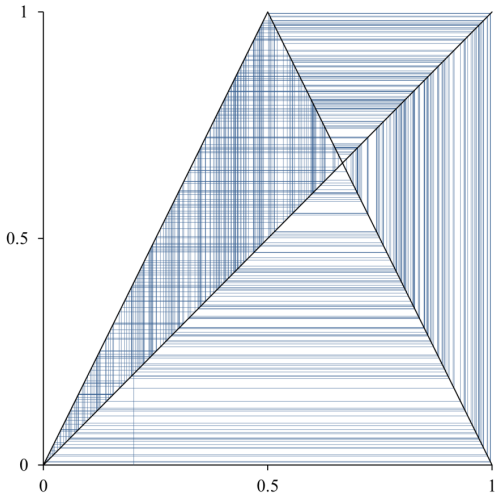

Figure 2.2: Cobweb plot of the 'tent map'.

Exercise 3 (Punching the 'cos' button on a calculator).

Type any real number into a calculator. Push the cosine button repeatedly many times. Prove that you will always converge to the same number (0.739... in radians).

Exercise 4 (Iteration is complicated beyond Banach).

Consider the so-called ***tent map***, which is the piecewise linear function

$$g(x) = \begin{cases} 2x, & x \in [0, \frac{1}{2}], \\ 2 - 2x, & x \in (\frac{1}{2}, 1]. \end{cases}$$

This function maps the interval $[0, 1]$ onto itself and iterating this function leads to surprisingly complex and chaotic behavior, as can be seen from the cobweb plot in Figure 2.2. As usual define the iterative procedure $x_{n+1} = g(x_n)$ for a chosen x_0. We call x_0 a ***periodic point of period*** P, if $x_P = x_0$, which implies that $x_{n+P} = x_n$ for all n. Show that for the tent map defined above, the set of periodic points of all periods is dense in the interval $[0, 1]$. This means that even the smallest change in x_0 will dramatically change the behavior of the resulting sequence $\{x_n\}$.

Hint: To find a fixed point of g (which is a periodic point with period $P = 1$), draw the graph of g and a graph of the identity function $f(x) = x$ and look where they intersect. Similarly, to find points of period $P = 2$, draw the graph of $g \circ g$ and look at the intersections with $f(x) = x$. And so on. The result can be seen from realizing what the graphs of $g \circ g \circ \ldots \circ g$ looks like.

2.3 Newton's method

Newton's method is one of the most used basic methods for the numerical solution of nonlinear equations. The method can be found under various names in the literature:

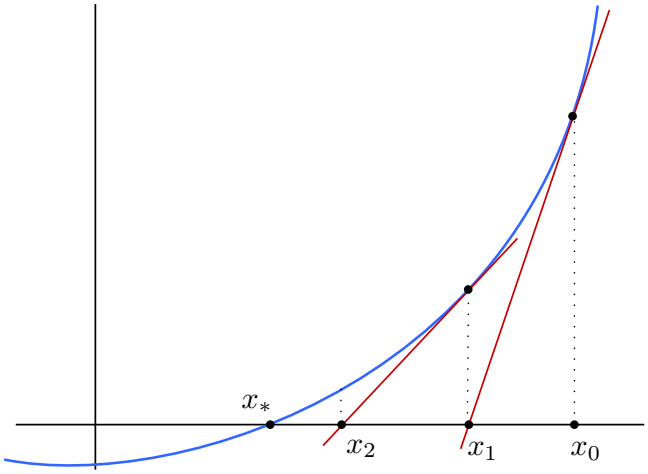

Figure 2.3: Newton's method.

usually Newton succeeded by one or more names from the list Raphson, Simpson, Fourier and Kantorovich (in Banach spaces). This is due to historical reasons which we shall mention later on.

The basic derivation is as follows: Given x_0, we do a first order Taylor expansion:

$$0 = f(x_*) = f(x_0) + f'(x_0)(x_* - x_0) + R.$$

Next, we neglect the remainder R and obtain the approximate identity

$$0 \approx f(x_0) + f'(x_0)(x_* - x_0),$$

which we solve for the desired x_*:

$$x_* \approx x_0 - \frac{f(x_0)}{f'(x_0)}. \tag{2.3}$$

We intuitively expect that if the neglected remainder R was small enough then the right-hand side of (2.3) could be a better approximation of x_* than x_0. Thus we denote it as x_1 and apply the procedure iteratively.

Definition 13 (Newton's method)**.** *Let x_0 be given.* **Newton's method** *consists of the iterative procedure*

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

A graphical interpretation of Newton's method can be seen in Figure 2.3. We draw a tangent to f at the point x_0 and see where that tangent intersects the x-axis. This is the new iterate x_1 and we repeat the procedure iteratively. We note that for 'nice' functions, the convergence is quite fast – we did not draw x_3 in Figure 2.3, as it would already be visually almost indistinguishable from x_*. This graphical interpretation of Newton's method is the reason it is sometimes called the method of tangents.

Another interpretation of Newton's method is that it is the fixed point iteration procedure with $g(x) = x - f(x)/f'(x)$. We can thus apply the results of Section 2.2 to analyze the method (see Exercise 5).

We note that if f is a linear function, then for arbitrary x_0, Newton's method gives the root exactly in the first iteration: $x_1 = x_*$. Compare this to the bisection method, which takes the same number of iterations irrespective of the simplicity of f. This is due to the fact that Newton's method takes into account more information about f than the simple sign change in two points.

We will now prove the basic theorem on local convergence of Newton's method. We shall do so in two versions – the basic version under stronger assumptions and an improved version under weaker assumptions. The reason we do this is that the first version has a simpler, more straightforward proof. However, this proof does not generalize well to systems of equations, for reasons we shall explain in Section 3.1. On the other hand, the second, slightly more technical proof generalizes straightforwardly to $\mathbb{R}^N$ and even general Banach spaces.

Theorem 14 (Local quadratic convergence of Newton I)**.** *Let $f \in C^2(U)$ for some neighborhood U of x_*. Let $f'(x_*) \neq 0$ and let x_0 be sufficiently close to x_*. Then Newton's method satisfies $x_n \to x_*$* **quadratically**.

Proof. We proceed as in the derivation of Newton's method, but this time we do not neglect the remainder of the Taylor expansion but write it down explicitly in Lagrange's form:

$$0 = f(x_*) = f(x_n) + f'(x_n)(x_* - x_n) + \frac{1}{2}f''(\xi_n)(x_* - x_n)^2,$$

where ξ_n lies between x_n and x_0. We divide by $f'(x_n)$ and rearrange to get

$$0 = x_* + \underbrace{\frac{f(x_n)}{f'(x_n)} - x_n}_{=-x_{n+1}} + \frac{1}{2}\frac{f''(\xi_n)}{f'(x_n)}(x_* - x_n)^2,$$

hence

$$x_{n+1} - x_* = \frac{1}{2}\frac{f''(\xi_n)}{f'(x_n)}(x_* - x_n)^2.$$

If we denote the error of the method as $e_n = x_* - x_n$, we obtain the ***error equation*** for Newton's method:

$$e_{n+1} = -\frac{f''(\xi_n)}{2f'(x_n)}\, e_n^2. \tag{2.4}$$

Convergence: Since $f \in C^2(U)$, $f'(x_*) \neq 0$, we have that the factor from (2.4) is uniformly bounded on some neighborhood V of x_*: there exists $M > 0$

$$\left|\frac{f''(x)}{2f'(y)}\right| \leq M \text{ for all } x, y \in V.$$

Now if we choose $x_0 \in V$ close enough to x_* so that $M|e_0| \leq 1/2$, we get from (2.4)

$$|e_1| \leq Me_0^2 = \big(M|e_0|\big)|e_0| \leq \tfrac{1}{2}|e_0|.$$

Therefore, $x_1 \in V$, since it is closer to x_* than x_0. Moreover, $M|e_1| \leq M|e_0| \leq 1/2$. We can therefore proceed by induction to get

$$|e_{n+1}| \leq Me_n^2 = \big(M|e_n|\big)|e_n| \leq \tfrac{1}{2}|e_n| \leq \ldots \leq \tfrac{1}{2^{n+1}}|e_0|. \tag{2.5}$$

Therefore, $e_n \to 0$, i.e. $x_n \to x_*$. We note that the crude estimate (2.5) would correspond to only linear convergence.

Quadratic convergence: From (2.5) and the fact that $x_n \to x_*$, we immediately have

$$\lim_{n\to\infty} \frac{|e_{n+1}|}{|e_n|^2} = \lim_{n\to\infty} \left| \frac{f''(\xi_n)}{2f'(x_n)} \right| = \left| \frac{f''(x_*)}{2f'(x_*)} \right| = C.$$

We note that $C > 0$ if $f''(x_*) \neq 0$. On the other hand, we see that in the special case when $f''(x_*) = 0$, we can expect even faster than quadratic convergence of Newton's method.

□

We note that the basis of the proof is the derivation of an identity relating the current error to the previous error(s) – the error equation (or inequality). From this we deduce convergence and the order of convergence. This will be the case in the proofs of similar theorems for the other methods we shall consider.

Now we give an improved version of Theorem 14 under weaker assumptions. The main difference is that instead of $f \in C^2$, we assume f' to be Lipschitz continuous. This is a seemingly trivial differences which however becomes important in $\mathbb{R}^N$ and in Banach spaces, where f'' is a more complicated object than f'.

Remark 15. *If a function is Lipschitz continuous on a set U, then its derivative exists almost everywhere in U and the derivative is in the Lebesgue space $L^\infty(U)$. Therefore, assuming f' to be Lipschitz continuous means that f'' exists almost everywhere and is in L^∞. Compare this to Theorem 14, where f'' exists everywhere and is a continuous function. In other words, the difference is assuming $f \in C^2$ versus $f \in W^{2,\infty}$.*

First, we need a substitute lemma for the remainder of Taylor's polynomial under the weaker assumptions. Here by "γ-Lipschitz" we mean "Lipschitz continuous with Lipschitz constant γ".

Lemma 16 (Substitute for Taylor)**.** *Let $f : (a, b) \to \mathbb{R}$ be such that f' is γ-Lipschitz. Then for all $x, y \in (a, b)$*

$$\big|f(y) - f(x) - f'(x)(y - x)\big| \leq \tfrac{1}{2}\gamma(y - x)^2. \tag{2.6}$$

Proof. Simple substitution gives us

$$f(y) - f(x) = \int_x^y f'(z)\,\mathrm{d}z = \left| \begin{smallmatrix} z=x+t(y-x) \\ \mathrm{d}z=(y-x)\,\mathrm{d}t \\ (0,1)\to(x,y) \end{smallmatrix} \right| = \int_0^1 f'\big(x + t(y - x)\big)(y - x)\,\mathrm{d}t,$$

therefore, due to the γ-Lipschitz continuity of f', we get

$$\begin{aligned} \big|f(y) - f(x) - f'(x)(y - x)\big| &= \left| \int_0^1 \Big[f'\big(x + t(y - x)\big) - f'(x)\Big](y - x)dt \right| \\ &\leq |y - x| \int_0^1 \gamma t|y - x|\,\mathrm{d}t = \tfrac{1}{2}\gamma|y - x|^2. \end{aligned}$$

□

We note that the right-hand side of (2.6) is essentially an estimate of the remainder of a first order Taylor polynomial. If we take e.g. the Lagrange form of the remainder $\frac{1}{2}f''(\xi)(y-x)^2$, we can estimate it by $\frac{1}{2}\max|f''|(y-x)^2$. In Lemma 16, we have the Lipschitz constant γ of f' instead of $\max|f''|$, which is closely related but weaker, as explained in Remark 15.

Theorem 17 (Local quadratic convergence of Newton II)**.** *Let f' be γ-Lipschitz on some neighborhood U of x_*. Let $f'(x_*) \neq 0$ and let x_0 be sufficiently close to x_*. Then Newton's method satisfies $x_n \to x_*$* **quadratically**.

Proof. By definition of x_{n+1}, we have

$$x_{n+1} - x_* = x_n - \frac{f(x_n)}{f'(x_n)} - x_* = \frac{1}{f'(x_n)}\big(\underbrace{f(x_*)}_{=0} - f(x_n) - f'(x_n)(x_* - x_n)\big). \quad (2.7)$$

Since $f'(x_*) \neq 0$ and f' is continuous, there exists $\rho > 0$ such that $|f'(x)| \geq \rho > 0$ for all $x \in U$. Using this estimate and Lemma 16 with $x = x_n, y = x_*$, we can estimate (2.7) on U:

$$|x_{n+1} - x_*| \leq \frac{1}{\rho} \cdot \frac{\gamma}{2}(x_n - x_*)^2.$$

This is the analogue of error equation (2.4) from which we derived quadratic convergence of the method. The rest of the proof thus continues similarly as in Theorem 14 and we shall omit it. □

There is another type of theorem concerning the convergence of Newton's method other than the local fixed point view from the previous theorems. The theorem gives more global convergence under more restrictive assumptions of convexity (concavity) of f. The statement and proof of the theorem can be easily seen in Figure 2.3, its formalization is only a technical issue.

Theorem 18 (Fourier)**.** *Let $f \in C^2[a,b]$ with $f(a)f(b) < 0$ such that f'' does not change sign in $[a,b]$. Let x_0 be such that $f(x_0)$ has the same sign as f''. Then for Newton's method $x_n \to x_*$ monotonically.*

Proof. The formal proof simply follows the intuition from Figure 2.3, however it is somewhat lengthy and has little 'added value', thus we rather move on to more important topics. We refer the interested reader to [Seg00]. □

Historical note

We end this section with a short overview of the history of Newton's method. In 1669, Newton considered the equation

$$x^3 - 2x - 5 = 0 \quad (2.8)$$

which has a root in the interval $(2, 3)$ due to opposite signs of the polynomial at these points. Newton wrote the root as $x_* = 2 + p$ which he substituted into (2.8) to obtain the new equation

$$p^3 + 6p^2 + 10p - 1 = 0. \tag{2.9}$$

Newton assumed that p is small, hence the higher order terms $p^3 + 6p^2 \approx 0$ are *very* small and he neglected them. Thus equation (2.9) reduces to $10p - 1 = 0$ with the solution $p = 0.1$. Next, Newton wrote $p = 0.1 + q$, substituted into (2.9) to obtain the equation

$$q^3 + 6.3q^2 + 11.23q + 0.061 = 0. \tag{2.10}$$

Again, Newton neglected the quadratic and cubic terms to obtain $11.23q + 0.061 = 0$ with the solution $q = -0.0054$. This procedure is then applied iteratively, writing $q = -0.0054 + r$ and substituting into (2.10), and so on. We note that already the second approximation is $2 + p + q = 2,0946$, which is a good approximation of the true root 2.094551482... Newton then proceeds to use this procedure on Kepler's equation $x - e\sin(x) = M$ from astronomy which is the relation between mean anomaly M and eccentric anomaly x. Newton used the technique he developed for polynomials by taking the Taylor expansion of $\sin x$. He noticed what we now call quadratic convergence of the iterates.

In 1690, Joseph Raphson improved the procedure by substituting the corrections back into the original equation instead of producing a new equation in each iterate. The advantage is that one saves a lot of work when performing the calculations by hand, since one can reuse some of the already computed quantities. Raphson thought he invented a new method, although in fact it is equivalent to Newton's original approach.

Notice that up to now, there is no mention of derivatives in the procedure. These were introduced into the linearization process by Thomas Simpson in 1740. Finally, it was Joseph Fourier, who wrote down Newton's method as we know it today. One can see that the development of the method was not straightforward and that it is, ironically, hard to recognize Newton's method in the tedious procedure that Newton himself used. It is for these reasons that the method is sometimes called Newton, Newton-Raphson, Newton-Raphson-Simpson, Newton-Raphson-Simpson-Fourier or some other combination of the mentioned names, possibly including Kantorovich in Banach spaces.

Exercises

Exercise 5 (Newton as a fixed point iteration procedure).

Use Theorem 12 to prove that Newton's method is locally quadratically convergent by verifying the assumption for g – the function which is iterated in Newton's method.

Exercise 6 (Computing the square root).

Given $A > 0$, compute $\sqrt{A}$ using Newton's method. Observe the extremely fast convergence for this simple problem. Prove convergence for any $x_0 > 0$.

Remark: There are two basic possibilities:

- If you start from the equation $x^2 = A$ and apply Newton's method, you end up with the so-called ***Babylonian method*** for computing square roots which was derived geometrically in ancient Babylonia some 3500 years ago and rediscovered 2000 years ago in ancient Greece (Hero's method). As a challenge, try deriving the resulting iterative procedure using *only* simple geometry, without any symbolic calculations or calculus, as was done in Babylonia.
- Another possibility is to start from the equation $1/x^2 = A$ to compute $1/\sqrt{A}$ and then multiply the final approximation by A. The final method has the advantage that it uses only multiplication without any division. Thus it is can be efficiently implemented (in more sophisticated form, e.g. ***Goldschmidt's algorithm***) directly in the hardware using logic gates.

Exercise 7 (Division without division).

Given $A \neq 0$, compute $1/A$ using only the operations $+, -, *$.

Hint: Try Newton's method for a suitable equation with the solution $x_* = 1/A$. Choose f so that the resulting formula from Newton's method contains only addition (subtraction) and multiplication. Analyze for which x_0 the method converges to $1/A$.

Remark: Improved versions of this method (e.g. Goldschmidt's algorithm) are actually hardware-implemented in many microprocessors to calculate division. The cost of such a calculation is of the same order as the cost of multiplication. This also plays an important role in special applications such as extremely high precision calculations (millions of digits) or cryptography (calculations with very large integers with thousands or millions of digits).

Remark: In the matrix setting, the method can be adapted to produce a quickly converging algorithm for the approximation of the inverse $\mathbb{A}^{-1}$ of a matrix $\mathbb{A}$ which uses only matrix addition and multiplication. This is the so-called ***Schulz iterative method*** for the matrix inverse from 1933: $\mathbb{X}_{n+1} = \mathbb{X}_n(2\mathbb{I} - \mathbb{A}\mathbb{X}_n)$. A reasonable initial value is $\mathbb{X}_0 = \mathbb{A}^{\mathrm{T}}/(\|\mathbb{A}\|_1\|\mathbb{A}\|_\infty)$.

2.3.1 What could possibly go wrong?

In this section we shall take a look at what can happen in Newton's method (and other locally convergent methods) beyond the neighborhood where Theorem 17 guarantees quadratic convergence. We shall see that the method can have (and typically has) very wild behavior. This is the price we pay for the very fast local convergence. Here is a basic list of the things that can go wrong if x_0 is not close enough to x_*:

1. Convergence to a different root than the one we desire.

2. Slower than quadratic convergence.

3. Divergence to $\pm\infty$.

4. $f'(x_n)$ is undefined or is equal to zero.

5. Periodic cycling of the iterates.

6. Chaotic behavior.

We demonstrate these phenomena in 1D, where one can gain at least some intuition from pictures and where the analysis is not too technical. The situation becomes much more complicated in $\mathbb{R}^N$, where the problems we mention here sometimes render the method unusable, since the neighborhood of (quadratic) convergence is impractically small. We shall discuss the individual points from the list above in more detail.

1. Convergence to a different root than the one we desire

Sometimes this is not a problem, sometimes it is a huge problem. An example of the latter case can be taken from finite element solvers for partial differential equations of physics. For example, for nonlinear first order hyperbolic conservation laws (such as the Euler equations describing compressible fluid flows) the equations themselves have many solutions, only one of which – the entropy solution – is physically admissible. It may happen that your Newton's method for your chosen finite element or volume method converges to an unphysical solution and you are unable to come up with an initial x_0 for which Newton's method would converge to the correct entropy solution. There is no general recipe how to fix this issue and sometimes elaborate techniques are required to find the correct solution..

2. Slower than quadratic convergence.

One can expect this to happen outside the neighborhood guaranteed by Theorem 17, but at least we have *some* form of convergence, albeit not quadratic. However, we may have slower than quadratic convergence even for all x_0 arbitrarily close to x_*. Clearly this happens when some assumption of Theorem 17 is violated. This may happen for two reasons:

Insufficient regularity. If f' is not Lipschitz continuous near x_*, one may observe slower than quadratic convergence. See Exercise 8 where Newton's method for $x + x^{4/3} = 0$ is considered, which results in a fractional convergence rate between 1 and 2.

Zero derivative. One can also observe slower than quadratic convergence when $f'(x_*) = 0$. This means that x_* is a root with multiplicity greater than one. We illustrate this on the simple equation $x^2 = 0$ with $x_* = 0$. Newton's method for this equation is

$$x_{n+1} = x_n - \frac{x_n^2}{2x_n} = \frac{1}{2}x_n \quad \Longrightarrow \quad |e_{n+1}| = \frac{1}{2}|e_n|,$$

which is ***linear*** convergence (in fact exactly the same convergence as for the bisection method). After some experimenting one notices that if we modified Newton's method to be $x_{n+1} = x_n - 2f'(x_n)/f(x_n)$ we would have

$$x_{n+1} = x_n - \frac{2x_n^2}{2x_n} = 0,$$

hence we get x_* exactly in the first iteration, at least for this particular equation. More generally, this modification gives x_* in one iteration for $f(x) = a(x - x_*)^2$, which is the general quadratic case of a double root:

$$x_{n+1} = x_n - 2\frac{a(x_n - x_*)^2}{2a(x_n - x_*)} = x_n - (x_n - x_*) = x_*.$$

Even more generally, for a root of multiplicity $r \in \mathbb{N}$ and the model equation $a(x - x_*)^r = 0$, the modification of Newton's method $x_{n+1} = x_n - rf'(x_n)/f(x_n)$ gives x_* in one iteration:

$$x_{n+1} = x_n - r\frac{a(x_n - x_*)^r}{ra(x_n - x_*)^{r-1}} = x_*.$$

It turns out that this modification works in general, not only for the model problems:

Theorem 19 (Roots with multiplicity)**.** *Let x_* be a root of multiplicity $r \in \mathbb{N}$ of f, i.e. $f^{(j)}(x_*) = 0$ for $j = 0, \ldots, r-1$ and $f^{(r)}(x_*) \neq 0$. Let $f \in C^r(U)$ where U is some neighborhood of x_*. Then the modified Newton method*

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

converges **locally quadratically** *to x_*.*

Proof. The proof is a technical calculation without much added value, thus we omit it and refer the interested reader to [RR78, Section 8.6]. □

In the theorem above, one needs to know the multiplicity of x_* in advance. This is not always the case, in fact one might not even notice apriori that x_* is a root of higher multiplicity (consider the equation $e^x - x - 1 = 0$ with $x_* = 0$). One universal solution is to consider the equation $\bar{f}(x) := f(x)/f'(x) = 0$ instead of $f(x) = 0$. Then x_* is always a root of multiplicity 1 for $\bar{f}$ irrespective of its multiplicity for f. One can then apply his/her favorite method to the modified equation.

3. Divergence to $\pm\infty$.

Now we get to the cases when Newton's method fails altogether. Among these, divergence to infinity is the 'nicest' because it is at least noticeable very early and can be easily detected in the implementation, unlike the other cases.

Consider the model equation $f(x) := \arctan(x) = 0$. Since $f'(x) \to 0$ as $x \to \pm\infty$, one can expect that if we choose x_0 sufficiently far away from $x_* = 0$, then x_1 will be even larger due to the division by $f'(x_n)$ in Newton's method. This is indeed the case, as can be seen by a simple analysis or by a simple numerical experiment. In fact, there exists a neighborhood of x_*, such that outside of this neighborhood

Newton's method diverges to $\pm\infty$. The size of this neighborhood can be computed, as we shall do in Exercise 9.

In general, divergence to infinity is usually caused by the denominator $f'(x_n)$ being very close to zero, in which case x_{n+1} is very large or even causes an overflow. There are tricks how to overcome this issue, which fall under the general category of "globalization strategies". Here we only briefly mention one possible strategy.

Damped Newton method. The idea is to do smaller steps than Newton's method recommends. In the simplest case, we can write this as

$$x_{n+1} = x_n - \lambda_n \frac{f(x_n)}{f'(x_n)}, \tag{2.11}$$

where we apply the damping parameters $\lambda_n \in (0, 1]$ which tells us how much of the Newton correction to take. Obviously, we should not destroy the quadratic convergence rate of Newton's method, once we are sufficiently close to x_*. Thus we require $\lambda_n \to 1$ when $x_n \to x_*$ (or perhaps $\lambda_n = 1$ for sufficiently large n). One can try to come up with an explicit formula for λ_n, such as $\lambda_n = 1/(1 + |f(x_n)|)$ which tries to use the residual of the equation to measure 'closeness' to x_*. A more sophisticated idea is to use a ***backtracking strategy***: try (2.11) with $\lambda_n = 1$, i.e. Newton's method. If the residual increases, i.e. $|f(x_{n+1})| \geq |f(x_n)|$, we try $\lambda_n = 1/2$ instead. Again, we test for decrease of the residual and if needed, try $\lambda_n = 1/4$ instead. We can write this procedure as

Given x_n, compute $\tilde{x} = x_n - \frac{f(x_n)}{f'(x_n)}$.
While $|f(\tilde{x})| \geq |f(x_n)|$:
    $\tilde{x} := \frac{1}{2}(\tilde{x} + x_n)$.
Set $x_{n+1} := \tilde{x}$.

The danger of simple procedures like the one above is that the iterates can converge to a local extreme, instead of the root. This happens because if we start close to a local extreme, then $f'(x_n) \approx 0$ and the Newton correction is detected to be too large. Therefore the step is taken to be much smaller, perhaps so small that we only get closer to the local extreme and the situation is much worse in the next iterate.

More sophisticated strategies similar to the one above are important in numerical optimization, where they fall under the category of *line search methods*, where much more sophisticated strategies for the choice of the new iterate $\tilde{x}$ and the necessary condition it must satisfy are considered.

We note that the backtracking procedure above can be viewed as a rudimentary *hybrid method*, which consists of a fast converging method combined with a (ideally) globally convergent method/strategy. More on these methods in Section 2.5.2.

4. Derivative undefined or equal to zero.

This case is quite similar to the previous case. From the practical point of view, one rarely encounters an exact zero in floating point arithmetic. Usually, due to rounding errors, we will 'only' have $f'(x_n) \approx 0$, not exactly $f'(x_n) = 0$. In this case we are in the previous case discussed above.

From the mathematical point of view, it can be shown that the set of 'bad' initial conditions $\{x_0 : \exists n \in \mathbb{N} \text{ s.t. } f'(x_n) = 0\}$ is at most countable. In other words, this set is very small from the point of view of $\mathbb{R}$ (measure zero) and it is in general unlikely to choose such a x_0 by chance.

5. Periodic cycling of the iterates.

We say that the sequence $\{x_n\}_{n=0}^{\infty}$ has period P, if $x_{n+P} = x_n$ for all n. It seems reasonable to assume that one can come up with an example of a function f and an initial estimate x_0, such that the resulting sequence from Newton's method has, for example, period 2. One may then say that such an artificial example will be very rare and it will be virtually impossible to choose the one special point x_0 by accident. However, one can go a bit further and construct a function f, such that for any $x_0 \neq x_*$, Newton's method has always period 2, see Figure 2.4. Indeed, we want an f such that Newton's method satisfies

$$x_{n+1} - x_* = x_* - x_n, \quad \forall n \in \mathbb{N},$$

which we can rewrite using the definition of Newton's method as

$$x_n - \frac{f(x_n)}{f'(x_n)} - x_* = x_* - x_n. \tag{2.12}$$

We want this equation to be satisfies for all x_n, so we can omit the subscript n and rewrite (2.12) as

$$\frac{f'(x)}{f(x)} = \frac{1}{2(x - x_*)}.$$

This is a simple ordinary differential equation for the unknown function f, which has the solution

$$f(x) = C \operatorname{sgn}(x - x_*)\sqrt{|x - x_*|}, \tag{2.13}$$

for any $C \in \mathbb{R}$, cf. Figure 2.4. Indeed, one may verify that for any $x_0 \neq x_*$, Newton's method jumps periodically back and forth between two values. We note that this is possible, since the function from (2.13) violates the assumptions of Theorem 17, since $f'(x_*)$ is infinite, hence f' cannot be Lipschitz continuous on any neighborhood of x_*.

One can then dismiss the previous example by saying that it is one very special function which we will never solve by Newton's method anyway, since we can easily solve $f(x) = 0$ analytically in this case. However a more interesting situation is described in Exercise 10, where the function $f(x) = x^3 - 2x + 2$ is considered. If we choose $x_0 = 0$, Newton's method gives the sequence $0, 1, 0, 1, 0, 1, \ldots$. One can say that this is very nice but if we choose any other x_0, we will not get a periodic sequence anymore. This is true, but the interesting phenomenon behind this example is contained in the following lemma:

Lemma 20. *Let $f(x) = x^3 - 2x + 2$. There exists a neighborhood U of 0, such that for any $x_0 \in U$, the sequence from Newton's method satisfies*

$$\begin{aligned} \lim_{n\to\infty} x_{2n} &= 0, \\ \lim_{n\to\infty} x_{2n+1} &= 1. \end{aligned}$$

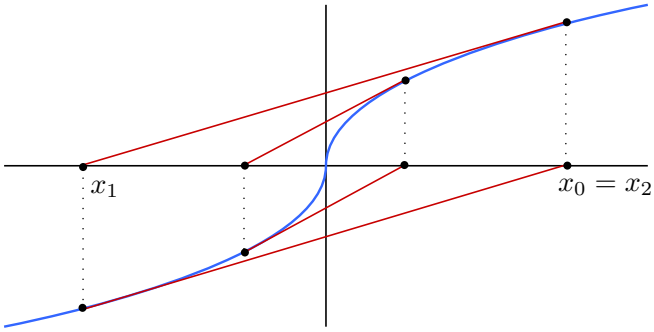

Figure 2.4: Periodic cycling of Newton's method, two different cycles.

Moreover, the convergence in the limits above is quadratic.

This means that if we start near 0, the sequence from Newton's method converges to $0, 1, 0, 1, 0, 1, \ldots$. Moreover, for this example, the odd/even terms will converge to 0 or 1 very quickly (quadratically)! It is then hard to dismiss this example of periodic cycling as a rare thing which we will never encounter – there is a whole open set of initial values for which Newton converges to a 2-periodic sequence very quickly. And choosing an x_0 from an open set is something that can easily happen in practice – this is not one isolated point which we will easily avoid.

In general, the situation is much more complicated. The following theorem can be obtained as a special case of the Sharkovskii or Li & Yorke theorems known from the theory of chaotic dynamical systems.

Theorem 21 (Sharkovskii, Li & Yorke)**.** *Let $f : \mathbb{R} \to \mathbb{R}$ be a polynomial with at least four distinct real roots. Then for any $P \in \mathbb{N}$ there exists an x_0 such that Newton's method has period P.*

It is good to consider the implications of Theorem 21. The function f is not some 'wild' counterexample form measure theory, it is a simple polynomial (degree 4 is sufficient). Even so, Newton's method has extremely complicated behavior when we look beyond the neighborhood where we have local quadratic convergence – for any period we choose, there is a x_0 such that Newton will have that period. There will be a point with period 5 and 17 and 123456789 and 10^{365}. Moreover, there will also be ***preperiodic*** points x_0. This means that x_n will first 'jump around' for a while before settling on e.g. period 17. This means there is a large (although countable) set of initial values for which Newton's method will (eventually) periodically cycle.

We note that the situation is much more complicated in general, since the phenomena from Theorem 21 and from Lemma 20 can combine. Then we can have also open sets from which we will converge to periodic cycling with some period.

6. Chaotic behavior.

Even more generally, we may consider the set of all points x_0 for which Newton's method does not converge to any root, even though we never hit a point where $f'(x_n) = 0$ (i.e. the whole sequence is well defined). We have the following theorem.

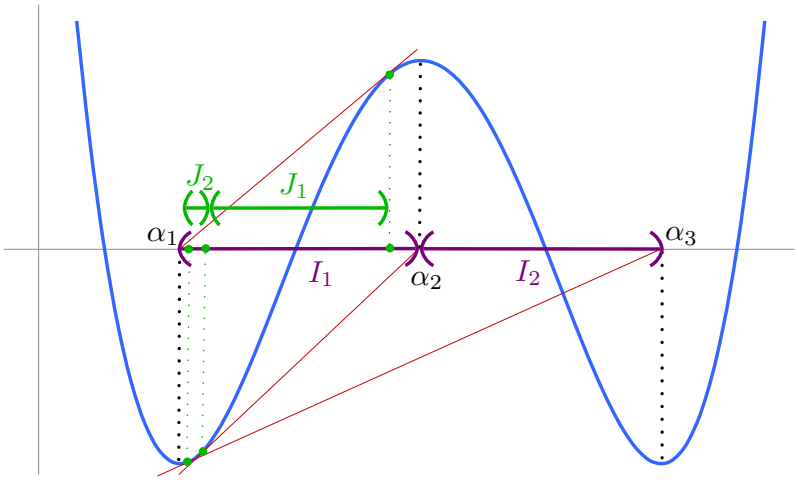

Figure 2.5: Simplest case of the Saari-Urenko theorem (Theorem 23) and its proof.

Theorem 22 (Barna)**.** *Let $f : \mathbb{R} \to \mathbb{R}$ be a polynomial of degree $D \geq 4$ with D distinct real roots. Let W be the set of x_0 such that:*

1. *the whole sequence $\{x_n\}_{n=0}^{\infty}$ from Newton's method is well defined,*
2. *$\{x_n\}_{n=0}^{\infty}$ does not converge to any root of f.*

Then W is homeomorphic to a **Cantor discontinuum***, i.e. it is uncountable, closed, has empty interior and has no isolated points.*

The set W form Theorem 22 is uncountable, hence it is large from the set-theoretic point of view. Obviously it contains the periodic points from Theorem 21. However this is 'only' a countable subset. So what do all the other points look like? Simply stated, these are the points x_0 for which Newton "chaotically" jumps around. Specifically, we have the following result if f is a polynomial as in Theorem 22.

Theorem 23 (Saari, Urenko)**.** *Let $f : \mathbb{R} \to \mathbb{R}$ be a polynomial of degree $D \geq 4$ with D distinct real roots. Let $\alpha_1 < \alpha_2 < \ldots < \alpha_{D-1}$ be the roots of f' (i.e. the local extremes). Define the intervals $I_j = (\alpha_j, \alpha_{j+1})$ for $j = 1, \ldots, D-2$. Choose an* arbitrary *sequence $\{S_n\}_{n=0}^{\infty}$ such that each $S_n \in \{1, \ldots, D-2\}$. Then there exists an $x_0 \in \mathbb{R}$ such that Newton's method satisfies*

$$x_n \in I_{S_n}, \quad \forall n = 0, 1, \ldots.$$

To fully appreciate Theorem 23, we illustrate it in the simplest case. Consider a quartic ($D = 4$) polynomial f as in Figure 2.5. We have three points $\alpha_1, \alpha_2, \alpha_3$, where f' is zero. These define two intervals I_1 and I_2. Now we can choose any sequence of ones and twos and the theorem gives an x_0 such that the iterates of Newtons method jump between I_1 and I_2 accordingly. For example, if we choose the sequence $\{S_n\}_{n=0}^{\infty}$ as

$$1\ 2\ 1\ 1\ 2\ 1\ 1\ 1\ 2\ 1\ 1\ 1\ 1\ 2 \ldots,$$

we get an x_0 such that the consecutive iterates $x_0, x_1, x_2, \ldots$ lie in the intervals

$$I_1, I_2, I_1, I_1, I_2, I_1, I_1, I_1, I_2, I_1, I_1, I_1, I_1, I_2, \ldots.$$

It is clear that such a sequence cannot converge to a root. It also cannot be a periodic sequence and it cannot converge to a periodic sequence, as in Exercise 10, since the iterates x_n jump between the intervals I_1 and I_2 aperiodically. The only possibility for convergence would be convergence to the common endpoint α_2. However this is also not possible, since $f'(\alpha_2) = 0$, therefore if x_n would be too close to α_2, due to the size of the Newton update the next iterate x_{n+1} would be very far away, certainly it would lie outside of the finite intervals I_1 and I_2, thus violating the theorem.

One can choose even 'wilder' sequences $\{S_n\}_{n=0}^{\infty}$ – we might choose a random sequence. Thus Newton's method will jump around 'randomly' (even though it is a fully deterministic process). We might choose the sequence to correspond to the digits of the binary expansion of $\sqrt{2}$ or π. These are irrational numbers, so again, the iterates cannot converge to a periodic sequence. How many are there such possible aperiodic sequences $\{S_n\}_{n=0}^{\infty}$? The answer is uncountably many, since the set of irrational numbers is uncountable.

We call the behavior above ***chaotic***, since the iterates can 'jump around' seemingly randomly, never converging to some reasonable behavior. Even so, the theorem on local quadratic convergence is still valid – there are neighborhoods around the roots, where Newton will converge. This is typical of chaotic dynamical systems – the mixing of very regular behavior (e.g. periodicity or convergence) with seemingly random behavior. There is an entire field of research devoted to this topic.

We note that the proof of Theorem 23 is not difficult and is in fact (relatively) constructive and intuitive. We indicate the basic idea in Figure 2.5.

Proof of Theorem 23 (basic idea). Consider the simplest case of a polynomial of degree 4, as in Figure 2.5. Let $g(x) = x - f(x)/f'(x)$ be the mapping defining Newton's method. Since $\lim_{x\to\alpha_1+} g(x) = +\infty$ and $\lim_{x\to\alpha_2-} g(x) = -\infty$, we have $g(I_1) = \mathbb{R}$. Similarly $g(I_2) = \mathbb{R}$. Therefore, for any point (or interval) in $\mathbb{R}$, we can find its preimage under g in both I_1 and I_2. For example, there exist intervals $J_1, J_2 \subset I_1$ such that $g(J_1) = I_1$ and $g(J_2) = I_2$. These can easily be found 'graphically', as in Figure 2.5 (depicted in green). Altogether, in our case, if we choose an interval $I \subset (\alpha_1, \alpha_3)$ then $g^{-1}(I)$ consists of two intervals, one of which is in I_1, while the other is in I_2. The whole 'trick' of the theorem is choosing which of the two preimages we take, according to the prescribed sequence.

Assume for example that the sequence $\{S_n\}_{n=0}^{\infty}$ is $1\,2\ldots$ Taking the first two symbols, we seek a point $x_0 \in I_1$ such that $x_1 = g(x_0) \in I_2$. The set of these points is exactly the aforementioned interval J_2. In other words, J_2 is one component of $g^{-1}(I_2)$. We would choose the other component (which lies in I_2), if the sequence started $2\,2\ldots$ In general, we consider $g^{-1}(g^{-1}(\ldots g^{-1}(I_i)\ldots))$ for $i = 1, 2$, where we take the individual preimages from I_1 or I_2 according to the prescribed sequence. The successive preimages are smaller and smaller intervals and if one proceeds carefully, in the limit one obtains a single point x_0. This is the rough idea behind the proof. □

Exercises

Exercise 8 (Insufficient regularity).

Consider the equation $x + x^{4/3} = 0$. Write down Newton's method for this equation. Derive the rate of convergence of x_n to the root $x_* = 0$ directly from the definition of convergence rate. Why is it not two? What assumption of Theorem 17 is violated? Try to find a generalization of Theorem 17 for functions with weaker regularity.

Hint: Instead of Lipschitz continuity of f', consider Hölder continuity: A function h is α-Hölder continuous if there exist constants $C, \alpha > 0$ such that $|h(x) - h(y)| \leq C|x - y|^\alpha$ for all x, y. Is f' from the exercise α-Hölder continuous for some α? Go through the proof of Theorem 17 with the assumption of Hölder continuity instead of Lipschitz.

Exercise 9 (Local convergence).

Use Newton's method to solve the equation $\arctan(x) = 0$. Due to Theorem 17 there exists $x_c > 0$ such that Newton's method converges for all $x_0 \in (-x_c, x_c)$. However, if $|x_0| > x_c$, Newton's method will diverge to $\pm\infty$. Calculate x_c explicitly using Newton's method.

Hint: If we choose $x_0 = x_c$, Newton's method gives us $x_1 = -x_c$, $x_2 = x_c$, etc. (draw a picture!). Therefore, if we write Newton's method as $x_{n+1} = g(x_n)$, then x_c solves the equation $g(x_c) = -x_c$. This equation (probably) cannot be solved analytically, so use Newton's method to calculate the solution x_c.

Exercise 10 (Periodic cycling).

Consider $f(x) = x^3 - 2x + 2$. The equation $f(x) = 0$ has a single real root $x_* \approx -1.7693$. Starting from $x_0 = 0$, Newton's method periodically cycles between the two points 0 and 1. Show that this behavior is locally attracting: There exists a neighborhood U of 0 (and also an analogous neighborhood of 1), such that for any $x_0 \in U$, the sequence from Newton's method satisfies

$$\begin{aligned} \lim_{n\to\infty} x_{2n} &= 0, \\ \lim_{n\to\infty} x_{2n+1} &= 1. \end{aligned}$$

Moreover, the convergence in the limits above is quadratic.

Hint: If we write Newton's method as $x_{n+1} = g(x_n)$, consider what happens when we iterate $g \circ g$ in the situation above. Apply the results of Section 2.2.

2.3.2 Stopping criteria

This section discusses the seemingly simple question of when to stop iterating when seeking for a root with sufficient accuracy. What we discuss here is not necessarily

restricted to Newton's method, it applies to general iterative processes for nonlinear equations. Moreover, by changing the absolute value $|.|$ to a norm $\|.\|$, we can apply these ideas to systems of equations.

On a general level, when testing whether a sequence $x_0, x_1, \ldots$ converges to a solution of $f(x_\star) = 0$, one has, in principle, only two basic possibilities:

1. Testing convergence.

2. Testing if we satisfy the equation.

In the first case, we essentially test if the sequence is a Cauchy sequence. In the second case, we measure the residual of the equation. In either case, we want x_n to satisfy some given tolerance ε.

1. Testing for a Cauchy sequence

The simplest criterion for a Cauchy sequence is the ***absolute criterion***

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

The problem with this (and other) absolute criteria is that it does not take into account the typical magnitude of the numbers that we are dealing with. If in our problem we expect x_n and x_* to typically be on the order of 10^{10}, it is not reasonable to prescribe the same tolerance ε as if the typical numbers in our problem are on the level of 10^{-10}. Choosing e.g. $\varepsilon = 10^{-10}$ leads to a criterion that may not even be satisfiable in finite precision arithmetic in the first example, while in the second example the criterion may be satisfied even if we do not have a single relevant digit of accuracy in the approximate solution. From this point of view it seems more reasonable to choose some form of ***relative criterion***, e.g.

$$\frac{|x_n - x_{n-1}|}{|x_n|} < \varepsilon. \tag{2.14}$$

This is a more robust approach, however it can also fail in certain situations.

Example. Consider Newton's method for the equation $x^2 = 0$. The iterative process is $x_n = \frac{1}{2}x_{n-1}$. Evaluating the left-hand side of (2.14) in this case gives us

$$\frac{|x_n - x_{n-1}|}{|x_n|} = \frac{\frac{1}{2}|x_{n-1}|}{\frac{1}{2}|x_{n-1}|} = 1.$$

Therefore, criterion (2.14) can *never* be satisfied for any $\varepsilon < 1$, even though $x_n \to x_*$. We note that changing the denominator in (2.14) to $|x_{n-1}|$ does not help, as the expression only evaluates to $\frac{1}{2}$ instead of 1.

One recommendation how to avoid the problem in the previous example is to use the following criterion

$$\frac{|x_n - x_{n-1}|}{\max\{|x_n|, x_{\text{typ}}\}} < \varepsilon. \tag{2.15}$$

Here, x_{typ} is a nonzero user-specified quantity representing the 'typical' magnitude of the argument x that we are working with (we shall encounter this quantity again

in (2.23), where we deal with the effect of rounding errors). The exact value of x_{typ} is not really important, as long as it has roughly the correct magnitude we are working with in our problem. This of course limits the usefulness of (2.15) in 'black-box' solvers, however when our problem comes e.g. from physics, we have at least a general idea of what to expect of the quantities involved.

It is obvious that measuring the distance of two iterates in general has nothing in common with the distance to the solution. The method could be caught in a temporary stagnation phase, where progress is slow for some reason, even though we are far from the solution.

1. Testing the residual

The other approach is to test how much we satisfy the equation, i.e. measure the magnitude of $|f(x_n)|$. The problem is that in general, the size of the residual can have little in common with the distance to the solution. This is true especially for systems of equations, and even for systems of linear equations. However, we do not have much else to work with in general. The ***absolute criterion*** in this case would be

$$|f(x_n)| < \varepsilon.$$

In a ***relative criterion***, we can measure e.g. the decrease of the residual relative to the initial one:

$$|f(x_n)| < \varepsilon |f(x_0)|.$$

This condition can however prove to be unsatisfiable, if $|f(x_0)|$ was already a small number. In this case one can use a combination of the relative and absolute criteria:

$$|f(x_n)| < \varepsilon_1 |f(x_0)| + \varepsilon_2.$$

Finally, one can combine the mentioned criteria to, for example:

$$\texttt{IF} \quad |f(x_n)| < \varepsilon_1 |f(x_0)| + \varepsilon_2 \quad \texttt{OR} \quad \frac{|x_n - x_{n-1}|}{\max\{|x_n|, x_{\text{typ}}\}} < \varepsilon_3 \quad \texttt{STOP}.$$

One might also try an `AND` instead of an `OR` in the above, but this might run into satisfiability issues.

Remarks:

- As usual, there are no universal recipes in numerical mathematics.
- The choice of ε should be a reasonable trade-off between accuracy, attainability and also rounding errors on the level of 'machine precision' $\varepsilon_{\text{mach}}$. A reasonable choice is to set e.g. $\varepsilon \sim \sqrt{\varepsilon_{\text{mach}}}$. This is a quantity, we will encounter also when balancing approximation and rounding errors on page 32.
- If one has some additional information, this can be used. For example, once Newton's method actually starts converging quadratically, we have

$$\underbrace{x_n - x_*}_{e_n} = x_n - x_{n+1} + \underbrace{x_{n+1} - x_*}_{O(|e_n|^2)},$$

therefore $|x_n - x_{n+1}|$ and $|e_n|$ can be expected to have the same magnitude, once we are in the quadratically convergent regime. Therefore testing the Cauchy property is justified in this case and gives a relevant estimate of the error.

2.4 Secant method

So far we have have considered the situation when f is explicitly given by a formula. This is the purely mathematical setting of Newton's and other methods. However this is often not the case – f could be e.g. given by the output from another calculation implemented in some lengthy code, or perhaps as output from an experiment. In this setting, we can evaluate the values of f, but we do not have easy access to the values of its derivative f'. What then can we do in Newton's method where we need to evaluate f'? The basic idea is to approximate the derivative somehow – we will treat this approach in the following section. This is also the basis of the secant method, cf. Section 2.4.2.

For completeness, let us mention one other possibility – *automatic differentiation*, which can be applied when f is given by a computer program. The goal of automatic differentiation is to take the program for evaluating f and turn it into a program for evaluating f'. There are various techniques, tools, and libraries that one can use to achieve this, here we only briefly describe the basic idea behind one of them. The general idea is to take variables in the program, e.g. `double v`, and replace it with pairs of numbers (variables), e.g. `double v[2]`. The first component `v[0]` corresponds to the value of the variable and the second component `v[1]` corresponds to the derivative. One can then define new rules for operators, functions, etc., which work on these new 'extended' variables according to the rules of differentiation, e.g. `(v*w)[0]=v[0]*w[0], (v*w)[1]=v[1]*w[0]+v[0]*w[1]`. One can thus rewrite the code for f to obtain the code for f'. This can even be as simple as overloading the definitions of data types, operators, functions, etc., while the code itself stays essentially the same. As mentioned, there are many approaches to this, which differ e.g. by the order in which we evaluate the differentiation of composite functions (forward mode or reverse mode), which can differ by their computational or memory efficiency.

2.4.1 Approximation of the derivative in Newton's method

The basic idea here is to approximate f' by a simple difference with a step h_n in each iteration. This leads to the following:

Definition 24 (Newton's method with differences)**.** *Let x_0 be given. Let $h_k \neq 0, k = 0, 1, \ldots$ be given.* **Newton's method with differences** *consists of the iterative procedure*

$$\begin{aligned} a_n &= \frac{f(x_n + h_n) - f(x_n)}{h_n}, \\ x_{n+1} &= x_n - \frac{f(x_n)}{a_n}. \end{aligned} \tag{2.16}$$

Obviously, $a_n \approx f'(x_n)$ is a first order approximation with respect to h_n. Specifically, by taking by taking $y = x_n + h_n$ and $x = x_n$ in Lemma 16 we get

$$\big|a_n - f'(x_n)\big| \leq \tfrac{1}{2}\gamma|h_n|, \tag{2.17}$$

where γ is the Lipschitz constant of f'.

We note that Newton's method with differences only requires the evaluation of f and not f', unlike Newton's method. The question is then what happens to the convergence rate.

Theorem 25 (Convergence of Newton with differences)**.** *Let f' be γ-Lipschitz on some neighborhood U of x_*. Let $f'(x_*) \neq 0$. Then there exists $\eta > 0$ such that if $\{h_n\}_{n=0}^{\infty}$ satisfies $0 < |h_n| < \eta$ then for any x_0 sufficiently close to x_* Newton's method with differences (2.16) converges to x_*. The convergence of the method is:*

- **linear***: if there exists $c > 0$ such that $|h_n| > c$ for all n,*
- **superlinear***: if $\lim_{n\to\infty} h_n = 0$,*
- **quadratic***: if there exists $C > 0$ such that $|h_n| \leq C|x_n - x_*|$.*

Proof. By definition of x_{n+1}, we have

$$\begin{aligned} x_{n+1} - x_* = x_n - \frac{f(x_n)}{a_n} - x_* &= \frac{1}{a_n}\big(\underbrace{f(x_*)}_{=0} - f(x_n) - a_n(x_* - x_n)\big) \\ &= \frac{1}{a_n}\big(\underbrace{f(x_*) - f(x_n) - f'(x_n)(x_* - x_n)}_{(*)} + \underbrace{(f'(x_n) - a_n)(x_* - x_n)}_{(**)}\big). \end{aligned} \tag{2.18}$$

Lemma 16 and (2.17) give us

$$\begin{aligned} \big|(*)\big| &\leq \tfrac{1}{2}\gamma|e_n|^2, \\ \big|(**)\big| &\leq \tfrac{1}{2}\gamma|h_n||e_n|. \end{aligned} \tag{2.19}$$

It remains to estimate a_n in the denominator of (2.18). Since $f'(x_*) \neq 0$ and f' is continuous, there exists $\rho > 0$ such that $|f'(x)| \geq \rho > 0$ for all $x \in U$. We use the triangle inequality in the form $|A| \geq |B| - |A - B|$ to get

$$|a_n| \geq |f'(x_n)| - |a_n - f'(x_n)| \geq \rho - \tfrac{1}{2}\gamma|h_n| \geq \tfrac{1}{2}\rho, \tag{2.20}$$

for $|h_n| \leq \eta$ sufficiently small.

Altogether, if we apply the estimates (2.19) and (2.20) in (2.18), we get the error inequality

$$|e_{n+1}| \leq \frac{1}{\frac{1}{2}\rho}\,\tfrac{1}{2}\gamma\big(|e_n|^2 + |h_n||e_n|\big) = \frac{\gamma}{\rho}\big(|e_n| + |h_n|\big)|e_n|. \tag{2.21}$$

All the statements of the theorem follow from this error inequality. □

One might ask whether there can be a practical choice of h_n so that we get a quadratically convergent method, since Theorem 25 requires $|h_n| \leq C|e_n|$ and $|e_n|$ is an unknown quantity which we can only estimate (if we knew e_n, we would immediately have the exact solution $x_* = x_n + e_n$). There is however one seemingly strange choice of h_n which naturally fulfills this requirement. ***Stephensen's method*** is obtained by taking $h_n = f(x_n)$ in (2.16):

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

The method is quadratically convergent by Theorem 25, since

$$|h_n| = |f(x_n)| = |f(x_n) - f(x_*)| \leq L|x_n - x_*|,$$

where L is the Lipschitz constant of f (we assume even f' Lipschitz continuous, hence f is also Lipschitz). We note that the method works relatively well, with all of the strengths and weaknesses of Newton's method, although at first it might seem strange to have f composed with itself in the definition of the method. Nevertheless it is a quadratically convergent method.

Choice of h_n in finite precision arithmetic

From the purely mathematical viewpoint, one should choose h_n very small in order to get a good approximation of f' by the difference and recover the quadratic convergence of Newton's method from Theorem 25. However in practice we cannot choose h_n arbitrarily small due to errors in floating point operations. Then even the evaluation of f is subject to rounding or approximation errors. Specifically, instead of $f(x)$, we are actually computing some approximation $f(x) + \varepsilon(x)$. Not much can be said about the error $\varepsilon(x)$, except that it is bounded from above for some range of the variable x: $|\varepsilon(x)| \leq \bar{\varepsilon}$. This upper bound would usually be assumed to be proportional to the 'machine epsilon' $\varepsilon_{\text{mach}}$ and to $|x|$, but we will not get into such details. Under these assumptions, we can estimate the error of the difference approximation of the derivative in finite precision arithmetic as

$$\begin{aligned} &\left| f'(x) - \frac{f(x+h) + \varepsilon(x+h) - \big(f(x) + \varepsilon(x)\big)}{h} \right| \leq \\ \leq &\left| f'(x) - \frac{f(x+h) - f(x)}{h} \right| + \left| \frac{\varepsilon(x+h) - \varepsilon(x)}{h} \right| \leq \frac{1}{2}\gamma|h| + \frac{2\bar{\varepsilon}}{|h|} = O\Big(|h| + \frac{\bar{\varepsilon}}{|h|}\Big). \end{aligned} \tag{2.22}$$

We wish to take $|h|$ so that the right-hand side estimate in (2.22) is minimal. If we minimize the expression $|h| + \bar{\varepsilon}/|h|$ with respect to $|h|$, we get the minimum at $|h| = \sqrt{\bar{\varepsilon}}$. Therefore the general recommendation is to take h on the order of $\sqrt{\bar{\varepsilon}}$ or $\sqrt{\varepsilon_{\text{mach}}}$. We say 'on the order of', because we only have a rough bound on $\varepsilon(x)$ and it is essentially useless to try to carefully evaluate the constants in (2.22).

In general, one also has to take into account the error relative to the current and 'typical' value of x_n. One possible recommendation from the literature is taking

$$|h_n| = \sqrt{\varepsilon_{\text{mach}}}\, \max\{|x_n|, x_{\text{typ}}\}. \tag{2.23}$$

Here, x_{typ} is a user-specified quantity representing the 'typical' magnitude of the argument x that we are working with (we have already encountered this quantity in Section 2.3.2). The reasoning behind (2.23) is this: it is reasonable to have h_n proportional to x_n – taking the perturbation $h = 0.1$ is clearly not the same if $x_n = 1$ and if $x_n = 10^{10}$. Also, we usually know where the problem comes from and what is the typical magnitude of x – if the equation comes from physics and x represents atmospheric pressure in pascals, we expect typically $x \approx 10^5$. If the equation comes from electrical engineering and x represents capacitance in farads, we expect $x \approx 10^{-6}$ or smaller. Hence it is reasonable to take $|h_n| = \sqrt{\varepsilon_{\text{mach}}}|x_n|$. However this can fail. Let us assume that we typically expect $x \approx 1$. It may happen that Newton's method in some iteration gives $x_n = 0$ (perhaps because of some chaotic phase before the method starts converging). Then we should take $|h_n| = \sqrt{\varepsilon_{\text{mach}}}\,|x_n| = 0$ which is nonsense. Or if Newton gives x_n very small, then $|h_n|$ should be unrealistically small and results in a bad approximation due to round-off errors. The quantity x_{typ} is added into (2.23) to circumvent this obstacle. Since it is only a fail-safe, the specific value of x_{typ} is not important, only a rough approximation is sufficient for the purpose.

Finally we note that we can use other approximations of the derivative f'. For example one can take the central difference

$$f'(x) \approx \frac{f(x+h) - f(x-h)}{2h} \tag{2.24}$$

which has an error of $O(|h|^2)$. This can be advantageous when the error $\bar{\varepsilon}$ in the evaluation of f is too large and the choice of h_n from (2.23) results in a large error. If we use the approximation (2.24) instead, then the optimal choice of $|h_n|$ is on the order of $\sqrt[3]{\bar{\varepsilon}}$ and the approximation improves. The price we pay is three evaluations of f per Newton iteration: $f(x_n + h_n), f(x_n), f(x_n - h_n)$.

2.4.2 Secant method

In Newton's method with differences, we need to evaluate the term $f(x_n + h_n)$ in each iteration. The idea behind the secant method is to reuse an already computed value of f. Namely, if we set $h_n = x_{n-1} - x_n$ then $f(x_n + h_n) = f(x_{n-1})$ which is a quantity that we have already computed in the previous iteration. Thus we save one function evaluation per iteration.

If we set $h_n = x_{n-1} - x_n$ in the definition of Newton's method with differences (2.16), we get

$$a_n = \frac{f(x_n + h_n) - f(x_n)}{h_n} = \frac{f(x_{n-1}) - f(x_n)}{x_{n-1} - x_n},$$
$$x_{n+1} = x_n - \frac{f(x_n)}{a_n} = x_n - \frac{f(x_n)(x_{n-1} - x_n)}{f(x_{n-1}) - f(x_n)} = \frac{x_{n-1}f(x_n) - x_n f(x_{n-1})}{f(x_n) - f(x_{n-1})}.$$

This is secant method.

Definition 26 (Secant method)**.** *Let x_0, x_1 be given. The* **Secant method** *consists of the iterative procedure*

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

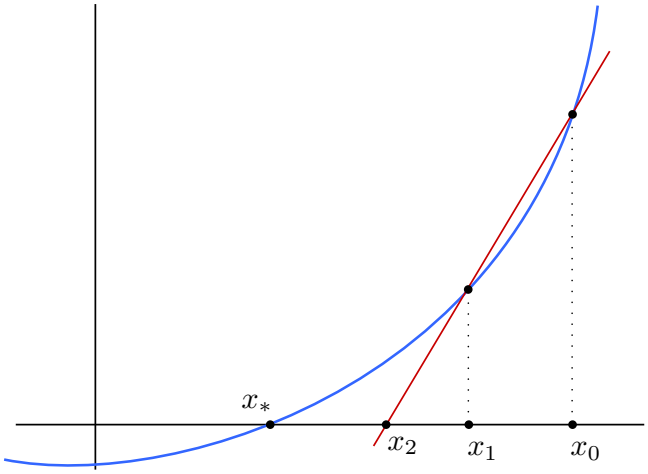

Figure 2.6: Secant method.

Remarks:

- We have only one evaluation of f per iteration, since we know f_{n-1} from the previous iteration. Compare to Newton or Newton with differences, where we need two function evaluations per iteration.
- We do not need to know f'.
- We need ***two*** initial values x_0 and x_1 instead of one.
- A more classical derivation and interpretation of the secant method is contained in Figure 2.6. We define a linear approximation of f by passing a line through the points $(x_{n-1}, f(x_{n-1})), (x_n, f(x_n))$ in the plane and define x_{n+1} as the root (or intersection with the x-axis) of this linear approximation. Hence the name ***secant*** method instead of method of tangents (a less common name for Newton's method).
- By Theorem 12, one-point iteration methods have only integer rates of convergence. By 'one-point' method we mean methods of the form $x_{n+1} = g(x_n)$, where the next iteration depends only on one previous iterate. The secant method is a two-point method, i.e. it can be written in the form $x_{n+1} = g(x_n, x_{n-1})$. Such methods can have general ***non-integer*** rates of convergence, as is the case for the secant method.

Theorem 27 (Local superlinear convergence of the secant method)**.** *Let $f \in C^2(U)$ for some neighborhood U of x_*. Let $f'(x_*) \neq 0$ and let x_0, x_1 be sufficiently close to x_*. Then the secant method converges* **superlinearly** *to x_*. The convergence rate is $\frac{1+\sqrt{5}}{2} \approx 1.618$.*

Proof. In order to simplify the notation, we define $f_n := f(x_n), f_{n-1} := f(x_{n-1})$, etc. We divide the proof into two parts.

1. Derivation of the error equation. By definition of x_{n+1}, we have

$$e_{n+1} = \frac{x_{n-1}f_n - x_n f_{n-1}}{f_n - f_{n-1}} - x_* = \frac{e_{n-1}f_n - e_n f_{n-1}}{f_n - f_{n-1}} = e_n e_{n-1} \underbrace{\frac{\frac{f_n}{e_n} - \frac{f_{n-1}}{e_{n-1}}}{f_n - f_{n-1}}}_{(\star)}. \tag{2.25}$$

The rest of this part of the proof lies in expressing the term $(\star)$. Since $f(x_*) = 0$, we have

$$(\star) = \frac{\frac{f_n - f(x_*)}{x_n - x_*} - \frac{f_{n-1} - f(x_*)}{x_{n-1} - x_*}}{f_n - f_{n-1}} = \frac{F(x_n) - F(x_{n-1})}{f_n - f_{n-1}}, \tag{2.26}$$

where we have defined the auxiliary function F as

$$F(x) = \frac{f(x) - f(x_*)}{x - x_*}.$$

We can express the denominator in (2.26) by the mean value theorem:

$$F(x_n) - F(x_{n-1}) = F'(\xi_n)(x_n - x_{n-1}), \tag{2.27}$$

for some ξ_n between x_n and x_{n-1}. The derivative F' from (2.27) can be explicitly computed as

$$F'(x) = \frac{f'(x)(x - x_*) - f(x) + f(x_*)}{(x - x_*)^2} = \frac{\frac{1}{2}f''(\tilde{\xi}_n)(x - x_*)^2}{(x - x_*)^2} = \tfrac{1}{2}f''(\tilde{\xi}_n), \tag{2.28}$$

for some $\tilde{\xi}_n$ between x (which is ξ_n in (2.27), hence the dependence of $\tilde{\xi}_n$ on n). The second equality in (2.28) follows from Taylor's expansion: $f(x_*) = f(x) + f'(x)(x_* - x) + \frac{1}{2}f''(\tilde{\xi}_n)(x_* - x)^2$.

By combining (2.27) with (2.28), we get

$$F(x_n) - F(x_{n-1}) = \tfrac{1}{2}f''(\tilde{\xi}_n)(x_n - x_{n-1}),$$

which we can substitute into (2.26) to obtain

$$(\star) = \tfrac{1}{2}f''(\tilde{\xi}_n)\,\frac{x_n - x_{n-1}}{f_n - f_{n-1}} = \tfrac{1}{2}f''(\tilde{\xi}_n)\,\frac{1}{f'(\hat{\xi}_n)},$$

which follows from the mean value theorem $f_n - f_{n-1} = f'(\hat{\xi}_n)(x_n - x_{n-1})$.

Having expressed the term $(\star)$ from (2.25), we finally arrive at the error equation for the secant method in the form

$$e_{n+1} = \frac{f''(\tilde{\xi}_n)}{2f'(\hat{\xi}_n)}\,e_n e_{n-1}, \tag{2.29}$$

where $\tilde{\xi}_n$ and $\hat{\xi}_n$ lie between x_n and x_{n-1}.

2. Convergence rate. From the error equation (2.29) we can easily derive convergence of the method: the factor $\frac{f''(\tilde{\xi}_n)}{2f'(\hat{\xi}_n)}$ can be bounded by some constant M on some neighborhood of x_* and if e_0, e_1 are sufficiently small with respect to M, we

get $|e_n| \to 0$ by induction similarly as in the proof of Theorem 14. What remains is to derive the convergence rate.

Let us assume that the secant method has rate of convergence p. If we define

$$S_n = \frac{|e_{n+1}|}{|e_n|^p}, \tag{2.30}$$

then by the definition of convergence rate p, there exists $C > 0$ such that

$$\lim_{n\to\infty} S_n = C > 0. \tag{2.31}$$

From the definition (2.30) of S_n, we can express

$$\begin{aligned} |e_n| &= S_{n-1}|e_{n-1}|^p \\ |e_{n+1}| &= S_n|e_n|^p = S_n\big(S_{n-1}|e_{n-1}|^p\big)^p = S_n S_{n-1}^p |e_{n-1}|^{p^2}. \end{aligned} \tag{2.32}$$

Now, we turn our attention to the error equation (2.29). We take its absolute value and divide by $|e_n||e_{n-1}|$ in order to have all the error terms on one side. We get

$$\frac{|f''(\tilde{\xi}_n)|}{2|f'(\hat{\xi}_n)|} = \frac{|e_{n+1}|}{|e_n||e_{n-1}|}. \tag{2.33}$$

Since the method converges, we have $\tilde{\xi}_n, \hat{\xi}_n \to x_*$. The left-hand side term from (2.33) therefore converges to a constant which is in general a finite nonzero number:

$$\lim_{n\to\infty} \frac{|f''(\tilde{\xi}_n)|}{2|f'(\hat{\xi}_n)|} = \frac{|f''(x_*)|}{2|f'(x_*)|} \neq 0. \tag{2.34}$$

Concerning the right-hand side of the error relation (2.33), we can express it using (2.32) and take the limit to obtain

$$\begin{aligned} \lim_{n\to\infty} \frac{|e_{n+1}|}{|e_n||e_{n-1}|} &= \lim_{n\to\infty} \frac{S_n S_{n-1}^p |e_{n-1}|^{p^2}}{S_{n-1}|e_{n-1}|^p |e_{n-1}|} = \lim_{n\to\infty} S_n S_{n-1}^{p-1} |e_{n-1}|^{p^2-p-1} = \\ &= C^p \lim_{n\to\infty} |e_{n-1}|^{p^2-p-1}, \end{aligned} \tag{2.35}$$

since $S_n, S_{n-1} \to C > 0$, by the definition of convergence rate (2.31). Altogether, if we take the limit of the error relation (2.33) and substitute (2.34) and (2.35), we get

$$0 \neq \frac{|f''(x_*)|}{2|f'(x_*)|} = \lim_{n\to\infty} \frac{|f''(\tilde{\xi}_n)|}{2|f'(\hat{\xi}_n)|} = \lim_{n\to\infty} \frac{|e_{n+1}|}{|e_n||e_{n-1}|} = C^p \lim_{n\to\infty} |e_{n-1}|^{p^2-p-1}. \tag{2.36}$$

Since $|e_{n-1}| \to 0$, the value of the last limit in (2.36) depends on the sign of the exponent:

$$\lim_{n\to\infty} |e_{n-1}|^{p^2-p-1} = \begin{cases} 0, & p^2-p-1 > 0, \\ \infty, & p^2-p-1 < 0, \\ 1, & p^2-p-1 = 0. \end{cases} \tag{2.37}$$

The first case is not possible, since (2.36) would reduce to $0 \neq 0$. The second case is also not possible, since the right-hand side of (2.36) would be infinite (here it is important that $C^p \neq 0$), while the left-hand side is a finite (but nonzero) number. Therefore, the only possibility for (2.36) to be satisfied is the third case $p^2 - p - 1 = 0$. This equation has a single positive root $p = \frac{1+\sqrt{5}}{2} \approx 1.618$, which is the desired convergence rate.

□

When reading the proof of Theorem 27, it is not easy to gain an intuitive insight as to why the error equation (2.29) leads to the convergence rate $\frac{1+\sqrt{5}}{2}$. Compare this to Newton's method, where the error equation is simply $e_{n+1} \approx e_n^2$, from which we immediately see the convergence rate of 2. Let us now try to gain some intuition on the case of the secant method.

We write the error equation for the secant method (2.29) in the simplified form

$$e_{n+1} \approx e_n e_{n-1}. \tag{2.38}$$

Let us assume for simplicity that the initial errors are on the order

$$e_0, e_1 \approx 10^{-1}.$$

By repeatedly applying the error equation (2.38), we get

$$e_2 \approx 10^{-2}, \quad e_3 \approx 10^{-3}, \quad e_4 \approx 10^{-5}, \quad e_5 \approx 10^{-8}, \quad e_6 \approx 10^{-13}, \dots$$

We can notice that the exponents are the Fibonacci numbers F_n defined by the recurrence $F_{n+1} = F_n + F_{n-1}$ with $F_0 = F_1 = 1$. Altogether, we have

$$e_n \approx 10^{-F_n}. \tag{2.39}$$

There are many classical results relating properties of Fibonacci numbers to the 'golden ratio'. For example, one can derive that

$$\lim_{n\to\infty} \frac{F_{n+1}}{F_n} = \frac{1+\sqrt{5}}{2}. \tag{2.40}$$

This follows e.g. from Binet's formula, which is an explicit formula for F_n. If we combine (2.39) with (2.40) for sufficiently large n, we get

$$e_{n+1} \approx 10^{-F_{n+1}} = \left(10^{-F_n}\right)^{\frac{F_{n+1}}{F_n}} \approx e_n^{\frac{1+\sqrt{5}}{2}}.$$

From this we can see the convergence rate $\frac{1+\sqrt{5}}{2}$ of the secant method. Of course these simple considerations only serve to gain intuitive insight and do not constitute a rigorous proof.

Which is better – Newton or secants?

At first glance, it may seem that the quadratically convergent Newton method is clearly better than the 'merely' superlinear secant method. This is true if we simply compare the convergence rates. However, Newton is twice as expensive as the secant method in terms of function evaluations per iteration – Newton needs two function evaluations, while secants only need one. Therefore, it would be more fair to compare ***one*** iteration of Newton with ***two*** iterations of the secant method. One iteration of Newton gives us

$$|e_{n+1}| \leq C|e_n|^2. \tag{2.41}$$

On the other hand, if we denote $p = \frac{1+\sqrt{5}}{2}$, two iterations of the secant method give us

$$|e_{n+2}| \leq C|e_{n+1}|^p \leq C\big|C|e_n|^p\big|^p = \tilde{C}|e_n|^{p^2}. \tag{2.42}$$

Now in order to compare (2.41) and (2.42), we need to know whether p^2 is bigger or smaller than 2. But if we go back to (2.37), p was obtained as the solution of $p^2 - p - 1 = 0$. Therefore, $p^2 = p + 1$ and if we remember that $p \approx 1.618$, then we immediately see that $p^2 \approx 2.618$, which is larger than two.

Altogether, if we compare Newton with secants in terms of 'convergence rate with respect to number of function evaluations', the secant method actually slightly beats Newton's method.

Remark 28. *There is one more issue with the secant method. Once we are very close to x_*, the differences $x_n - x_{n-1}$ and $f(x_n) - f(x_{n-1})$ become very small. This may present a problem in finite precision arithmetic, as we have discussed in Section 2.4.1. It may happen that $x_n - x_{n-1}$ is too small with respect to the recommended value of $\sqrt{\varepsilon_{\text{mach}}}$ and the difference approximation of the derivative can no longer be trusted. Newton's method does not suffer from this issue.*

False position method (regula falsi)

The false position method is a very old method, which can be seen from the fact that in some countries it is known by its Latin name 'regula falsi'. The idea is to take the bisection method and instead of evaluating f at the midpoint of (a_n, b_n) and comparing signs, we evaluate f at the point obtained from the application of the secant method to a_n, b_n. Thus we bring more information about f into play, than just taking the midpoint universally for all functions. Here is the resulting algorithm:

Given $I_0 = [a_0, b_0]$ and a tolerance tol. Set $n = 0$.
While $(b_n - a_n) > tol$:

$$s_n = \frac{a_n f(b_n) - b_n f(a_n)}{f(b_n) - f(a_n)}.$$

If $f(a_n)f(s_n) \leq 0$

$$a_{n+1} := a_n,\ b_{n+1} := s_n,$$

else

$$a_{n+1} := s_n,\ b_{n+1} := b_n.$$

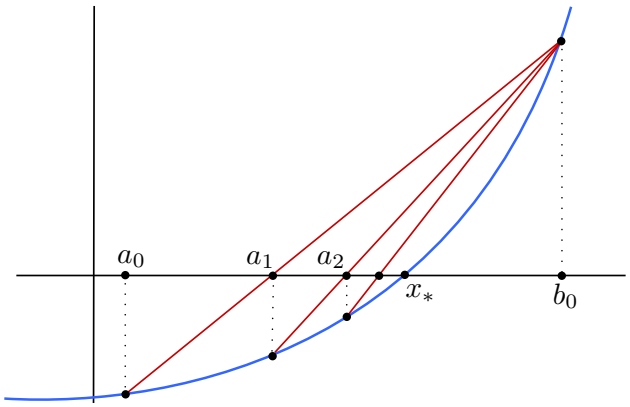

Figure 2.7: False position method (regula falsi).

We note that unlike in the bisection method, we do not have $|b_n - a_n| \to 0$ in general. This can be seen from the example in Figure 2.7. For such a function, $|b_n - a_n| \not\to 0$, however $a_n \to x_*$. In this sense, regula falsi converges to x_* for any continuous function – either $|b_n - a_n| \to 0$ or $a_n \to x_*$ or $b_n \to x_*$, cf. [YG88, Chapter 4.5] or [Seg00, Věta 6.5]. Moreover, it is truly a bracketing method, i.e. (a_n, b_n) always contains a root.

Regula falsi has only linear convergence in general. Sometimes it may happen that the convergence is even slower than for bisection, although this is usually not the case.

Exercises

Exercise 11 (Regula falsi as a fixed point iteration procedure).

Consider the example depicted in Figure 2.7. Specifically, let $f'' > 0$, $f' > 0$ on (a_0, b_0), where $f(a_0) < 0$ and $f(b_0) > 0$. Prove that regula falsi satisfies $a_n \to x_*$.

Hint: In this case $b_n = b_0$ for all n and only a_n changes according to $a_{n+1} = g(a_n)$ for some g. Find the formula for g and show that it is a contraction, given the assumptions on f.

2.5 Various more sophisticated methods

Current literature on the topic is full of many diverse numerical methods for the solution of nonlinear equations. Even a brief overview would be beyond the scope of these notes. Here we only present several basic examples of more sophisticated methods.

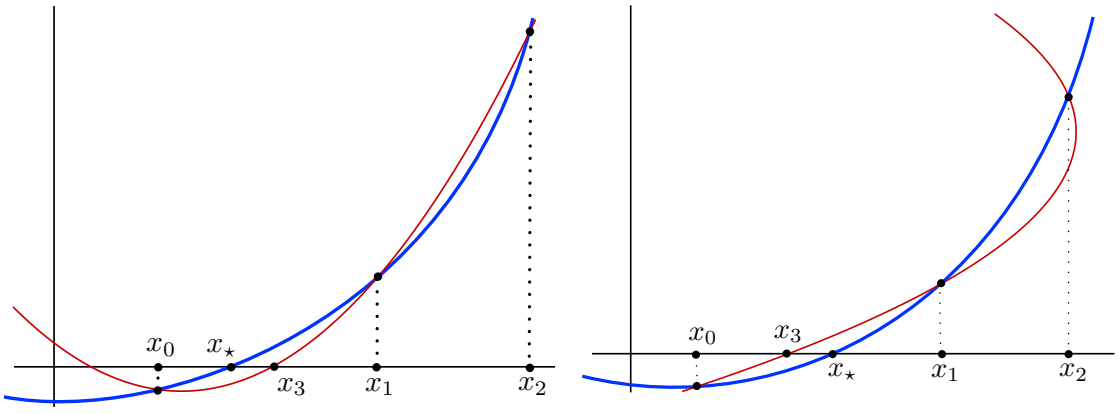

Figure 2.8: Muller's method (left) and IQI (right).

2.5.1 Methods based on quadratic interpolation

Newton's method was based on linearization of the nonlinear equation and solving the resulting linear equation. It is a natural idea to generalize this approach to the next level - quadratic approximations.

Muller's method

Muller's method is a direct generalization of the secant method to the quadratic case. Due to its simplicity, it seems to have been independently discovered and rediscovered many times by various people, however it is credited to David E. Muller in 1956. The idea is simple:

1. Use x_n, x_{n-1}, x_{n-2} and the values of f at these points to construct a quadratic polynomial p such that

$$p(x_{n-i}) = f(x_{n-i}), \quad i = 0, 1, 2.$$

2. Find the roots of p.

3. Define x_{n+1} as the root of p which is closest to x_n.

A graphical representation of Muller's method can be found in Figure 2.8. We will not bother to write down the final formula for x_{n+1}, this can be found elsewhere.

Remarks:

- A quadratic function need not have a real root. If this happens for the polynomial p in some iteration, the next x_{n+1} is not defined and the iteration process stops. One can fix this e.g. by switching to secants in that iteration or neglecting the imaginary parts of the roots.

- Another possibility is to accept the complex roots of p and use the method to search for complex roots of f (e.g. when f is a polynomial), which is actually a good idea.

- The convergence rate is approximately 1.839, which is the real root of $x^3 - x^2 - x - 1 = 0$, which stems from the error equation $e_{n+1} = C_n e_n e_{n-1} e_{n-2}$, where C_n is some quantity involving various derivatives of f. Compare this to the convergence rate 1.618 of the secant method, which is the root of $x^2 - x - 1 = 0$ stemming from the error equation $e_{n+1} = C_n e_n e_{n-1}$.
- Muller's method needs only one function evaluation per iteration. Therefore, similarly as in the secant method on page 38, if we wish to compare Newton and Muller, it is more fair to compare two iterations of Muller with one iteration of Newton. Two iterations of Muller's method have a combined convergence rate of $1.839^2 \sim 3.383$ which is much higher than the quadratic convergence rate of Newton.
- We now need ***three*** initial values x_0, x_1 and x_2.

Inverse quadratic interpolation (IQI)

A clever possibility how to avoid the problem with complex roots in Muller's method is the so-called Inverse quadratic interpolation (IQI) method introduced in 1826 by the French mathematician Germinal Pierre Dandelin (so technically it predates Muller's method). The idea is simple and elegant: interpolate the points $(x_i, f(x_i))$ in the plane not as a function "p of x", but rather as a function "$\tilde{p}$ of y", i.e. rotate the interpolation problem by 90°, cf. Figure 2.8. Now instead of seeking for the roots of the quadratic polynomial p from Muller's method, in IQI we get x_{n+1} by simply evaluating $\tilde{p}(0)$. The procedure is thus:

1. Use x_n, x_{n-1}, x_{n-2} and the values $f(x_n), f(x_{n-1}), f(x_{n-2})$ to construct a quadratic polynomial $\tilde{p}$ such that
$$\tilde{p}\big(f(x_{n-i})\big) = x_{n-i}, \quad i = 0, 1, 2. \tag{2.43}$$
2. Define $x_{n+1} = \tilde{p}(0)$.

Again, we do not write down explicit formulas for x_{n+1}, which can be found elsewhere. Obviously the method can break down, if there are two same values among $f(x_n), f(x_{n-1}), f(x_{n-2})$, since the interpolation problem (2.43) is not well posed in this case. This will obviously happen very rarely, but may cause instabilities in the algorithm when some of these values are very close to each other.

Otherwise, the basic properties of IQI are the same as of Muller: The rate of convergence of IQI is the same as in Muller's method (1.839), it also uses one function evaluation per iteration and also needs three initial values.

2.5.2 Hybrid algorithms

As we have seen in Section 2.3, Newton's method is very fast if we are sufficiently close too the root, however many bad things can happen outside of this neighborhood. On the other hand, bisection is guaranteed to converge to a root, albeit very

slowly, for any continuous function on any interval with a sign change in the endpoints. The idea of hybrid methods is to combine two or more methods, typically a very fast locally convergent method with a slow globally convergent method along with a set of criteria how to switch between the methods in every iteration. Typically this is done by combining open methods (Newton) and bracketing methods (bisection), however we can view the damped Newton method with backtracking (page 22) as a rudimentary example of such a method.

Dekker's method

The first sophisticated hybrid method was developed in 1969 by Theodorus Dekker (although variants of this method appeared in earlier works). The idea is to combine the bisection method with the secant method. We produce a sequence of brackets $[a_n, b_n]$ containing x_*, where the points b_n are the actual approximations of the root (b_n need not necessarily be the right endpoint of the bracket, it can be the left endpoint as well). We find two candidates for approximating x_s – those given by the secant and bisection methods. Then we choose which one we will be our next approximation b_{n+1}. Finally, we choose an appropriate a_{n+1} from all the available values to have a new bracket $[a_{n+1}, b_{n+1}]$. In detail, we have:

1. Let a_n, b_n and b_{n-1} be given such that $f(a_n)f(b_n) < 0$.
2. Compute the secant approximation from b_n, b_{n-1}:
$$s = \frac{b_{n-1}f(b_n) - b_n f(b_{n-1})}{f(b_n) - f(b_{n-1})}$$
3. Compute the bisection approximation (midpoint) from a_n, b_n:
$$b = \frac{a_n + b_n}{2}$$
4. If s lies between m and b_n, set $b_{n+1} := s$. Otherwise set $b_{n+1} := m$.
5. Set a_{n+1} to either a_n or b_n, so that $f(a_{n+1})f(b_{n+1}) < 0$.
6. If $|f(a_{n+1})| < |f(b_{n+1})|$, exchange a_{n+1} and b_{n+1}.
7. Set $n := n + 1$ and go to 1.

The criterion from point number 4 in the algorithm guarantees that the new approximation b_{n+1} does not jump too far from the current approximation b_n, which was one of the causes of non-convergence in Newton's method. Point number 5 ensures we will have a bracket containing the root in the next iteration. In point number 6 we assume that the residual of the equation is an indicator of the quality of approximation (which might or might not be true) and possibly change the role of a_{n+1} and b_{n+1} so that b_{n+1} is the 'better' approximation of x_*.

It can be proven that the method always converges and that $x_* \in [a_n, b_n]$ for all n. Sometimes the method can converge very slowly, even more slowly than bisection.

Brent's method

In 1973, Richard P. Brent published an improved version of Dekker's method. Without going into the technical detail, we only state that Brent's method chooses between bisection, secants and IQI in each iteration based on more sophisticated criteria. For general functions, the method can actually be slower than bisection – Brent shows that the method converges to the desired precision in at most N^2 iterations, if bisection needs N iterations. However, for 'nice' functions, the method typically performs IQI in most of the iterations, yielding superlinear convergence. We note that Brent's method is used in the `fzero` command in MATLAB.

Exercises

Exercise 12 (Falling out of an airplane).

A person falling from an airplane without a parachute falls $y(t)$ meters in t seconds, where

$$y(t) = \ln(\cosh(t\sqrt{gk}))/k,$$

where $g = 9.8065\,\mathrm{m/s^2}$ and $k = 0.00341\,\mathrm{m^{-1}}$. How long does it take for the person to fall one kilometer? Use Newton's method or the secant method to find the solution.

Remark: The formula for $y(t)$ can be obtained by solving the corresponding ordinary differential equation describing free fall in a gravity field with air friction proportional to the square of the velocity.

Chapter 3

Systems of equations

In this chapter, we will consider systems of N nonlinear equations for N unknowns. In other words, we will have a mapping $F : \mathbb{R}^N \to \mathbb{R}^N$ and we seek a root $x_* \in \mathbb{R}^N$, i.e. the solution of $F(x_*) = 0$. Since we are in $\mathbb{R}^N$, we can write everything in terms of individual components, namely $F = (f_1, \ldots, f_N)^T$ and $x = (x_1, \ldots, x_N)^T$.

In comparison to the scalar case things start to get qualitatively more complex:

- Much more complicated behavior and properties of the systems and numerical methods, causing the analysis to be much more technical and complex.
- A practical bracketing algorithm does not exist for high N.
- Computational costs tend to grow exponentially in terms of N in the worst case, or at least polynomially (so-called *curse of dimensionality*).

The situation is also much more complicated in comparison to the case of linear systems $\mathbb{A}x = \boldsymbol{b}$ and even in this case there are no universally functioning numerical methods. Consider the simple question of how many solutions the system of equations can have. In the linear case, the answer is $0, 1$ or ∞. For nonlinear equations, the answer is 0, any natural number, countably many and uncountably many. Moreover, even the number of roots can depend very sensitively on slight changes in the system.

Example. Consider the system

$$\begin{aligned} x_1^2 - x_2 + \gamma &= 0, \\ -x_1 + x_2^2 + \gamma &= 0, \end{aligned}$$

where $\gamma \in \mathbb{R}$ is a parameter. Note that each of the equations describes a parabola in the plane and γ determines their mutual position (draw a picture!). Depending on the value of γ, we have the following number of solutions:

$$\gamma = \begin{cases} 0.5, & 0 \text{ solutions}, \\ 0.25, & 1 \text{ solution}, \\ -0.5, & 2 \text{ solutions}, \\ -1.0, & 4 \text{ solutions}. \end{cases}$$

Example. Consider the system

$$\begin{aligned} x_1^5 x_2^7 - x_3^5 &= 1, \\ x_2^4 x_3^8 - x_1^4 &= 2, \\ x_1^8 x_3^4 - x_2^6 &= 3. \end{aligned}$$

The question is how many real solutions does the system have. I do know the answer, but one can estimate the maximum number. Since each equation is a polynomial, Bézout's theorem gives an upper bound on the number of roots as the product of the degrees of the multivariate polynomials. In our case this means that the system above can have up to $12^3 = 1728$ roots. We can see that even small, seemingly simple systems can have surprisingly complex behavior.

3.1 Tools from differential calculus

Here we will review the basic tools that we will need from differential and integral calculus in $\mathbb{R}^N$.

Definition 29. *A function $F : \mathbb{R}^N \to \mathbb{R}^M$ is* **differentiable** *at $x \in \mathbb{R}^N$ if there exists a linear mapping $A : \mathbb{R}^N \to \mathbb{R}^M$ such that*

$$\lim_{\substack{h \to 0 \\ h \in \mathbb{R}^N}} \frac{\|F(x+h) - F(x) - Ah\|}{\|h\|} = 0. \tag{3.1}$$

The mapping A is called the **derivative** *or* **differential** *of F at x and we will denote it as $F'(x)$.*

Remarks:

- The definition is independent of the specific choice of the norms in (3.1).
- If $F'(x)$ exists, then it is determined uniquely.
- If $F'(x)$ exists, then so do the partial derivatives $\frac{\partial f_i(x)}{\partial x_j}$ and the differential F' can be represented by the Jacobi matrix $\frac{DF}{Dx} = \{\frac{\partial f_i}{\partial x_j}\}_{i,j}$ in the sense that $F'(x)v = \frac{DF}{Dx}(x)v$ for all $v \in \mathbb{R}^N$.

We will need the following version of the ***mean value theorem***.

Lemma 30. *Let $F : D \subset \mathbb{R}^N \to \mathbb{R}^M$ be differentiable on a convex open set D and let F' be continuous on D. Then for all $x, y \in D$*

$$F(y) - F(x) = \int_0^1 F'(x + t(y-x))(y-x)\,\mathrm{d}t. \tag{3.2}$$

Proof. Since (3.2) is a vector identity, it is sufficient to prove it for each component separately. Define the scalar function $\varphi_i(t) = f_i(x + t(y-x))$. Then $\varphi_i'(t) = \nabla f_i(x + t(y-x)) \cdot (y-x)$. Therefore

$$f_i(y) - f_i(x) = \varphi_i(1) - \varphi_i(0) = \int_0^1 \varphi_i'(t)\,\mathrm{d}t = \int_0^1 \nabla f_i(x + t(y-x)) \cdot (y-x)\,\mathrm{d}t.$$

This is the i-th component of (3.2). □

We note that Definition 29 can be directly extended to mappings $F : X \to Y$, where X, Y are normed vector spaces – this is the ***Fréchet derivative***. Then the differential is a bounded linear operator from X to Y: $F'(x) \in \mathcal{L}(X, Y)$, hence $F' : X \to \mathcal{L}(X, Y)$. Therefore, if we want to go one step further and define $F'' = (F')'$, then we get $F''(x) \in \mathcal{L}(X, \mathcal{L}(X, Y))$. In the finite dimensional case, $F'(x) \in \mathcal{L}(X, Y)$ can be represented by a matrix, but $F''(x) \in \mathcal{L}(X, \mathcal{L}(X, Y))$ is representable by a tensor with components $\frac{\partial^2 f_i}{\partial x_j \partial x_k}$. This is the reason, why we avoid working with second derivatives (e.g. in the convergence theorem for Newton's method), as one must either be comfortable with tensor notation, or write everything in terms of the individual components which requires three indexes. It is then much easier to work with e.g. Lipschitz continuity of F' (i.e. $F'' \in L^\infty$, see Remark 15) than with continuity of F''. Obviously, things get much worse when working with even higher derivatives.

3.1.1 Fixed point iteration

If we take a mapping $G : \mathbb{R}^N \to \mathbb{R}^N$ with the fixed point x_* and define the simple fixed point procedure

$$x_{n+1} = G(x_n), \quad x_0 \in \mathbb{R}^N,$$

then Banach's fixed point theorem ensures convergence to a fixed point x_* if G is contractive, however contractivity is hard to prove directly for complicated mappings. In 1D, Corollary 3 gave a simple sufficient condition for local contractivity in the form of the assumption $g'(x_*) < 1$ and continuity of g'. One can ask what is an analogous condition in $\mathbb{R}^N$. Obviously, if G is (locally) contractive in some norm, then Banach's fixed point theorem is valid and $x_n \to x_*$. However contractivity is norm-dependent, hence G may be contractive in some norm, but not in another and choosing the correct norm may be a difficult task. In terms of Corollary 3, we might ask if the norm of $G'(x_*)$ is less then one. But again – which matrix norm should we take? Obviously, there has to be some norm-independent criterion – ***Ostrowski's theorem*** states that we should look at the spectral radius ρ of G'.

Theorem 31 (Ostrowski)**.** *Let x_* be the fixed point of G. Let G be differentiable at x_*. If $\rho(G'(x_*)) < 1$ then $x_n \to x_*$ for x_0 sufficiently close to x_*.*

Proof. The full proof can be found in [OR70]. Here we only indicate the basic idea. The basis is to construct a suitable norm in which G is (locally) contractive, which after some technicalities boils down to finding a norm in which $\|G'(x_*)\| < 1$. The norm is constructed using the following lemma from linear algebra: Let $\mathbb{A} \in \mathbb{C}^{N,N}$ be a matrix and let $\varepsilon > 0$, then there exists a consistent matrix norm $\|\cdot\|_{\mathbb{A},\varepsilon}$ such that $\|\mathbb{A}\|_{\mathbb{A},\varepsilon} \leq \rho(\mathbb{A}) + \varepsilon$. The rest of the proof is simple: denote $\mathbb{A} := G'(x_*)$. Then $\rho(\mathbb{A}) < 1$, hence there exists $\varepsilon > 0$ such that $\rho(\mathbb{A}) + \varepsilon < 1$. The mentioned lemma gives a specific norm $\|\cdot\|_{\mathbb{A},\varepsilon}$ such that $\|\mathbb{A}\|_{\mathbb{A},\varepsilon} \leq \rho(\mathbb{A}) + \varepsilon < 1$, i.e. a norm in which G is contractive. We note that the norm $\|\cdot\|_{\mathbb{A},\varepsilon}$ is explicitly constructed using the Jordan decomposition of $\mathbb{A}$. Also note that the definition of the norm $\|\cdot\|_{\mathbb{A},\varepsilon}$ depends on the specific choice of $\mathbb{A}, \varepsilon$ and 'degenerates' as $\varepsilon \to 0$. □

In the scalar case, Theorem 12 gives conditions on higher order convergence rates of the fixed point iteration process. Namely, if $g'(x_*) = 0$, then we have at

least quadratic convergence. And if $g''(x_*) \neq 0$, then we have exactly quadratic convergence. This can be generalized to systems.

Theorem 32. *Let x_* be the fixed point of G. Let G be constinuosly differentiable on a neighborhood of x_* and let $G''(x_*)$ exist. Let $G'(x_*) = 0$. Then $x_n \to x_*$ at least quadratically. If moreover $G''(x_*)hh \neq 0$ for all $0 \neq h \in \mathbb{R}^N$, then the convergence is exactly quadratic.*

3.2 Newton's method in $\mathbb{C}$

3.3 Newton's method

Now we shall derive and analyze Newton's method for systems of equations of the form $F(x_*) = 0$. Similarly as in 1D, the basis will be a suitable linearization, namely the first order Taylor expansion of F.

Lemma 33. *Let $f : \mathbb{R}^N \to \mathbb{R}$ be twice differentiable, let $x, a \in \mathbb{R}^N$. Then*

$$f(x) = f(a) + \nabla f(a) \cdot (x - a) + O(\|x - a\|^2). \tag{3.3}$$

Proof. Define $g(t) = f(a + t(x - a))$, then $g : \mathbb{R} \to \mathbb{R}$ and a we have $g'(t) = \nabla f(a + t(x - a)) \cdot (x - a)$. Thus we can apply the standard Taylor expansion to get

$$\underbrace{g(1)}_{f(x)} = \underbrace{g(0)}_{f(a)} + \underbrace{g'(0)}_{\nabla f(a)\cdot(x-a)} + \underbrace{\tfrac{1}{2}g''(\xi)}_{O(\|x-a\|^2)},$$

where $\xi \in [0, 1]$. □

Now we can apply (3.3) to the individual components f_i of F. Noticing that the i-th row of $F'(a)$ is $\nabla f_i(a)^T$, we can write everything together as

$$F(x) = F(a) + F'(a)(x - a) + R,$$

where $\|R\| = O(\|x - a\|^2)$. Similarly as in the 1D case, Newton's method is based on neglecting the remainder R and taking $x = x_*, a = x_0$:

$$0 = F(x_*) \approx F(x_0) + F'(x_0)(x_* - x_0).$$

Finally, we express the desired x_*:

$$x_* \approx x_0 - (F'(x_0))^{-1} F(x_0).$$

Definition 34 (Newton's method)**.** *Let x_0 be given.* **Newton's method** *consists of the iterative procedure*

$$x_{n+1} = x_n - (F'(x_n))^{-1} F(x_n). \tag{3.4}$$

In numerical mathematics, there are only a handful of cases, when one actually needs to approximate a matrix inverse. In formula (3.4) we certainly do not need to invert F' just to multiply it by a vector F. Thus it is useful to reformulate Newton's method using the solution of a linear-algebraic system, which can be obtained by multiplying (3.4) by $F'(x_n)$.

Definition 35 (Newton's method – alternative form)**.** *Let x_0 be given. Denote the Newton* **update** *as $\Delta x_n := x_{n+1} - x_n$. Then the alternative form of* **Newton's method** *consists of the iterative procedure: Given x_n, find Δx_n and update x_n such that*

$$\begin{aligned} F'(x_n)\Delta x_n &= -F(x_n), \\ x_{n+1} &= x_n + \Delta x_n. \end{aligned} \tag{3.5}$$

Remark 36. *There are specific situations when Newton's method is more practical in the original form (3.4) than (3.5). Consider for example the situation, when we have a nonlinear system of two equations for two unknowns with a prescribed constant right-hand side and suppose we need to quickly solve it many times with different right-hand sides. The formula for the inverse of a 2 by 2 matrix is simple and if we explicitly evaluate formula (3.4), it may happen that many cancellations and simplifications occur, resulting in a simple closed formula for the function G to be iterated, without the need to solve the linear algebraic systems from (3.5). Or perhaps we are in a (rare) situation, when we actually know $(F')^{-1}$ or its good approximation. This might happen when F' has a very simple structure or when we have some additional information from the problem itself (its derivation, the physical model it describes, etc.).*

3.3.1 Affine invariance and contravariance

Here we describe an interesting and important property of Newton's method – affine invariance and contravariance. The basic question is what happens when we perform a simple linear or affine transform of the system $F(x) = 0$. This could correspond e.g. to a scaling and translation of the unknowns or scaling and permutation of the equations. Ideally, one hopes that such operations do not fundamentally change the behavior of Newton's method – for example that swapping the first two equations in the system does not cause Newton's method to converge more slowly or cease to converge at all. Then the performance of the method would rely on such *ad hoc* things as proper scaling of the equations. As we shall prove in this section, Newton's method is 'invariant' under such transforms.

Let $\mathbb{A}, \mathbb{B} \in \mathbb{R}^{N,N}$ be given regular matrices and let $b \in \mathbb{R}^N$ be a given vector. We transform the original system $F(x) = 0$ to the new system $G(y) = 0$, where the new and old variables and systems are related by

$$G(y) = \mathbb{A}F(\mathbb{B}y + b) \quad \text{and} \quad x = \mathbb{B}y + b. \tag{3.6}$$

This means that we are transforming both the source and target spaces $\mathbb{R}^N$ by an affine and linear mapping, respectively. We note that the reason we do not transform affinely also in the target space, i.e. $G = \mathbb{A}F + a$ for some vector $a \in \mathbb{R}^N$, is that this would fundamentally change the problem, as we would no longer seek the root $G(y) = 0$ but the solution of $G(y) = a$.

Now we use Newton's method to solve both the problems. Newton for $F(x) = 0$ reads:

$$F'(x_n)\Delta x_n = -F(x_n), \quad x_{n+1} = x_n + \Delta x_n. \tag{3.7}$$

Newton for $G(y) = 0$ reads

$$G'(y_n)\Delta y_n = -G(y_n), \quad y_{n+1} = y_n + \Delta y_n,$$

which can be rewritten using (3.6) and the rule for differentiating composite functions as

$$\mathbb{A}F'(\mathbb{B}y_n + b)\mathbb{B}\Delta y_n = -\mathbb{A}F(\mathbb{B}y_n + b), \quad y_{n+1} = y_n + \Delta y_n.$$

Since $\mathbb{A}$ is regular, we can eliminate it from the equation for the update Δy_n to get

$$F'(\mathbb{B}y_n + b)\mathbb{B}\Delta y_n = -F(\mathbb{B}y_n + b). \tag{3.8}$$

This means that Newton's method for $G(y) = 0$ is independent of $\mathbb{A}$, i.e. it is independent of the transformation of the target space. This property is called ***affine invariance*** and we have just proven the following lemma:

Lemma 37. *Newton's method is* **affine invariant***: The sequence $y_n, n \in \mathbb{N}$, from Newton's method for G is independent of $\mathbb{A}$.*

The other property we shall prove is ***affine contravariance***, which is defined as follows. Let us solve $F(x) = 0$ and $G(y) = 0$ by Newton's method starting from the initial conditions x_0 and y_0, respectively, which satisfy $x_0 = \mathbb{B}y_0 + b$. This ensures that the two versions of Newton start from the 'same' initial point, only transformed appropriately via (3.6). Affine contravariance then means that the entire trajectories of the two versions of Newton are also related via (3.6), i.e. that $x_n = \mathbb{B}y_n + b$ for all n.

Lemma 38. *Newton's method is* **affine contravariant***: Let $x_0 = \mathbb{B}y_0 + b$, then $x_n = \mathbb{B}y_n + b$ for all n, where x_n and y_n are generated by Newton's method for finding the roots of F and G, respectively.*

Proof. We only prove the case of $n = 1$, the rest is obtained similarly by induction. The first step of (3.7), i.e. Newton's method for F, is

$$F'(x_0)\Delta x_0 = -F(x_0), \quad x_1 = x_0 + \Delta x_0.$$

Substituting $x_0 = \mathbb{B}y_0 + b$ into the first equation for Δx_0 gives

$$F'(\mathbb{B}y_0 + b)\Delta x_0 = -F(\mathbb{B}y_0 + b). \tag{3.9}$$

But this is the same equation as (3.8), i.e. Newton for G – only the solution to (3.8) is $\mathbb{B}\Delta y_0$ while the solution of (3.9) is Δx_0. Therefore these two solutions must be equal and we have $\Delta x_0 = \mathbb{B}\Delta y_0$ (the matrix $F'(x_0)$ is tacitly assumed to be regular, since x_1 would otherwise be undefined). Now if we update x_0, we get

$$x_1 = x_0 + \Delta x_0 = \mathbb{B}y_0 + b + \mathbb{B}\Delta y_0 = \mathbb{B}(y_0 + \Delta y_0) + b = \mathbb{B}y_1 + b,$$

so that $x_1 = \mathbb{B}y_1 + b$, which we wanted to prove. □

Remark 39. *Newton has both the affine invariance and contravariance properties, however this is a rare occurrence. Other methods usually have one property or the other, but not both – see for example the 'good' and 'bad' Broyden methods from Section 3.4.*

Exercises

Exercise 13 (Quadratic convergence of Newton via Ostrowski).

Consider Newton's method in $\mathbb{R}^N$, where we iterate the function

$$G(x) = x - (F'(x))^{-1}F(x).$$

Use Ostrowski's Theorem 31 to show that the iterates will converge for x_0 sufficiently close to x_*. Verify the criterion for quadratic convergence.

Hint: We need to show that $G'(x_*) = 0$. If one is not comfortable with computing G' directly, one can look at its individual entries $\frac{\partial g_i}{\partial x_j}$. Also in order to avoid differentiating the matrix inverse, consider calculating $G'(x_*)$ for the more general function $G(x) = x - \mathbb{B}(x)F(x)$, where $\mathbb{B}(x)$ is some matrix.

Exercise 14 (Simple 2 by 2 system).

Consider the system of equations

$$\begin{aligned} x^3 + y &= 1, \\ y^3 - x &= 1. \end{aligned}$$

Find the unique real solution to this system using Newton's method and also analytically. How robust is the convergence of the method with respect to the choice of (x_0, y_0)?

3.3.2 Local quadratic convergence of Newton's method

Here we will prove local quadratic convergence of Newton's method for systems of equations. The theorem we prove is a simplified version of the so-called Kantorovich theorem. We will need to prepare two auxiliary tools – first we prove an estimate of inverses of perturbed operators. We note that this lemma can be immediately extended to Banach spaces, however we state the result in finite dimension.

Lemma 40 (Banach perturbation lemma)**.** *Let $\mathbb{A} \in \mathbb{R}^{N,N}$ with $\|\mathbb{A}\| < 1$ for some consistent matrix norm $\|\cdot\|$. Then $(\mathbb{I} + \mathbb{A})^{-1}$ exists and*

$$\|(\mathbb{I} + \mathbb{A})^{-1}\| \leq \frac{1}{1 - \|\mathbb{A}\|}. \tag{3.10}$$

Proof. First we prove existence of the inverse. Let $0 \neq x \in \mathbb{R}^N$. Then the reverse triangle inequality gives us

$$\|(\mathbb{I} + \mathbb{A})x\| \geq \|x\| - \|\mathbb{A}x\| \geq (1 - \|\mathbb{A}\|)\|x\| > 0 \quad \text{for any } x \neq 0,$$

therefore $\mathbb{I} + \mathbb{A}$ is not singular, hence it has an inverse.

Now we can estimate the inverse. Since $\|\cdot\|$ is a consistent norm, we have

$$\begin{aligned} 1 = \|\mathbb{I}\| &= \|(\mathbb{I}+\mathbb{A})(\mathbb{I}+\mathbb{A})^{-1}\| = \|(\mathbb{I}+\mathbb{A})^{-1} + \mathbb{A}(\mathbb{I}+\mathbb{A})^{-1}\| \\ &\geq \|(\mathbb{I}+\mathbb{A})^{-1}\| - \|\mathbb{A}\|\|(\mathbb{I}+\mathbb{A})^{-1}\| = (1-\|\mathbb{A}\|)\,\|(\mathbb{I}+\mathbb{A})^{-1}\|. \end{aligned}$$

If we express $\|(\mathbb{I}+\mathbb{A})^{-1}\|$ from the inequality above, we immediately get (3.10). □

Next, we need an analogue to Lemma 16, the substitute for the remainder of Taylor's polynomial.

Lemma 41. *Let $F : D \subset \mathbb{R}^N \to \mathbb{R}^N$ be differentiable on a convex open set D and let F' be γ-Lipschitz on D. Then for all $x, y \in (a, b)$*

$$\big\|F(y) - F(x) - F'(x)(y-x)\big\| \leq \tfrac{1}{2}\gamma\|y-x\|^2. \tag{3.11}$$

Proof. The Mean value theorem (Lemma 30) gives

$$F(y) - F(x) = \int_0^1 F'(x + t(y-x))(y-x)\,\mathrm{d}t.$$

Therefore, by Lipschitz continuity of F',

$$\begin{aligned} \big\|F(y) - F(x) - F'(x)(y-x)\big\| &= \left\| \int_0^1 \big(F'(x+t(y-x)) - F'(x)\big)(y-x)\,\mathrm{d}t \right\| \\ &\leq \gamma\|y-x\|^2 \int_0^1 t\,\mathrm{d}t = \tfrac{1}{2}\gamma\|y-x\|^2. \end{aligned}$$

□

Theorem 42 (Local quadratic convergence of Newton)**.** *Let $F : D \subset \mathbb{R}^N \to \mathbb{R}^N$ be differentiable on a convex open set D. Let the following be satisfied:*

1. *F has a root $x_* \in D$,*
2. *$F'(x_*)$ is invertible and $\|F'(x_*)^{-1}\| \leq \beta < \infty$,*
3. *F' is γ-Lipschitz on D.*
4. *Kantorovich condition: $x_0 \in D$ is such that $\beta\gamma\|x_0 - x_*\| \leq \frac{1}{2}$.*

Then Newton's method starting from x_0 converges quadratically to x_* in the sense that $\|e_{n+1}\| \leq \beta\gamma\|e_n\|^2$ for all n.

Proof. We only consider the case $n = 0$, the rest follows by induction. We split the proof into two parts. First, we need to prepare an estimate of $\|F'(x_0)^{-1}\|$:
1. Estimate of $\|F'(x_0)^{-1}\|$. We have an estimate for $\|F'(x_*)^{-1}\|$ (assumption 2) We produce the desired estimate by Banach's perturbation Lemma 40. We have

$$\begin{aligned} \|F'(x_0)^{-1}\| &= \big\|\big(F'(x_*) + F'(x_0) - F'(x_*)\big)^{-1}\big\| \\ &= \big\|\big(F'(x_*)\big(\mathbb{I} + F'(x_*)^{-1}(F'(x_0) - F'(x_*))\big)\big)^{-1}\big\| \\ &\leq \underbrace{\big\|F'(x_*)^{-1}\big\|}_{\leq\beta}\,\big\|\big(\mathbb{I} + \underbrace{F'(x_*)^{-1}(F'(x_0) - F'(x_*))}_{=:\mathbb{A}}\big)^{-1}\big\|. \end{aligned} \tag{3.12}$$

Now we need to estimate $\|\mathbb{A}\|$ in order to apply Lemma 40"

$$\begin{aligned}\|\mathbb{A}\| &= \|F'(x_*)^{-1}(F'(x_0) - F'(x_*))\| \\ &\leq \|F'(x_*)^{-1}\|\|F'(x_0) - F'(x_*)\| \\ &\leq \beta\gamma\|x_0 - x_*\| \leq \tfrac{1}{2},\end{aligned} \tag{3.13}$$

due to assumptions 2, 3, and 4 (in that order). If we substitute (3.13) into (3.12) via Lemma 40, we get

$$\|F'(x_0)^{-1}\| \leq \beta\|(\mathbb{I} + \mathbb{A})^{-1}\| \leq \beta\frac{1}{1 - \|\mathbb{A}\|} \leq \beta\frac{1}{1 - \frac{1}{2}} = 2\beta. \tag{3.14}$$

We note that the estimate (3.14) makes intuitive sense: If, by assumption 2, we estimate $F'(x_*)^{-1}$ by β, then at a nearby point $F'(x_0)^{-1}$ can be estimated a little bit worse, by 2β.

2. Error estimate. Now we can estimate $e_1 = x_1 - x_*$. By expressing x_1 from Newton's method and adding the 'smart zero' $F(x_*)$, we get

$$\begin{aligned}x_1 - x_* &= x_0 - x_* - F'(x_0)^{-1}\big(F(x_0) - F(x_*)\big) \\ &= \underbrace{F'(x_0)^{-1}}_{\|\cdot\| \leq 2\beta}\big(\underbrace{F'(x_0)(x_0 - x_*) - F(x_0) + F(x_*)}_{\|\cdot\| \leq \frac{1}{2}\gamma\|x_0 - x_*\|^2}\big),\end{aligned}$$

due to estimate (3.14) and Lemma 41. Thus we have proved that

$$\|e_1\| \leq \beta\gamma\|e_0\|^2.$$

The estimate of e_n for general n can be obtained similarly, by induction. □

3.3.3 Variations on Newton's method

Inexact Newton

In practice, we do not solve the linear systems for the Newton update $F'(x_n)\Delta x_n = -F(x_n)$ from (3.5) exactly, but we apply some iterative method (preconditioned conjugate gradients, GMRES, etc.). Thus we solve the linear systems with some error. Specifically, we seek Δx_n such that the relative residual of the linear system is small in some sense, for example such that

$$\|F'(x_n)\Delta x_n + F(x_n)\| \leq \eta_n\|F(x_n)\|$$

with some tolerance η_n. If one analyzes the error of the resulting ***inexact Newton method***, one obtains the error inequality

$$\|e_{n+1}\| \leq C(\|e_n\| + \eta_n)\|e_n\|. \tag{3.15}$$

We note that this is essentially the same estimate as (2.21) in the proof of Theorem 25 (here η_n plays a similar role as $|h_n|$) and the proof of these two estimates follows essentially the same lines. The inequality (3.15) also has similar implications as in Theorem 25: If $\eta_n \to 0$ then we get superlinear convergence, etc. We note that assumption (3.15) and (3.15) are a 'black box' result independent of the iterative method used for the linear systems. Of coarse there are many papers written on the analysis of specific combinations such as Newton-CG, applied to specific problems.

Chord method

The idea of the chord method is to reuse the matrix in the linear system for the Newton update (3.5) for all or several iterations. So instead of solving (3.5), we solve

$$F'(x_0)\Delta x_n = -F(x_n) \tag{3.16}$$

in each Newton iteration. This can be advantageous from several points of view: we save computational time evaluating the matrix $F'(x_n)$ and if we spend more time on pre-computing a more sophisticated expensive preconditioner for $F'(x_0)$, we can solve systems (3.16) much more efficiently. Perhaps, we could even go as far as pre-computing an LU or Cholesky factorization of $F'(x_0)$. The price we pay is that Newton with the update given by (3.16) converges only linearly and is suitably only for 'mildly' nonlinear problems. One can then balance the mentioned advantages and disadvantages by updating the system matrix F' every K iterations for some K. Such methods can be analyzed and, for example, if we take $K = 2$, the resulting convergence rate is $\sqrt{3} \approx 1.732$.

Newton with differences

We discussed the use of difference approximations for the derivative in Newton's method is Section 2.4.1. Here the situation is more complicated, since we would have to approximate the whole Jacobi matrix F' using differences, i.e. approximate each partial derivative $\frac{\partial f_i}{\partial x_j}$ for all $i, j = 1, \ldots, N$, which is clearly impractical for large N.

However, there is an elegant approach to reducing the workload. As mentioned earlier, in practical applications, one solves the linear system for the Newton update by some numerical method. Often this will be some Krylov subspace method, e.g. conjugate gradients, etc. Such methods do not need the system matrix $\mathbb{A}$ to be explicitly given. Instead they require the repeated evaluation of $\mathbb{A}p$ for some given vector p. In our case, such a method will need the evaluation of $F'(x_n)p$ for a given $p \in \mathbb{R}^N$. But $F'(x_n)p$ is the *directional derivative* of F in the direction p, which can be approximated straightforwardly as

$$F'(x_n)p \approx \frac{1}{h}\big(F(x_n + hp) - F(x_n)\big)$$

for h small. This requires only one extra evaluation of $F(x_n + hp)$ and is therefore cheap compared to approximating the whole Jacobi matrix F'. In the end we do not even need a data structure to store matrices, only vectors, and such methods are called ***matrix-free algorithms***. These can be advantageous for example in the setting of finite element methods for partial differential equations, where the unknowns are often stored not in classical vector format (arrays), but as degrees of freedom in some strange data structure corresponding to the (unstructured) computational mesh. Creating a data structure for a matrix based on such a data mesh-based structure for vectors is sometimes rather technical and tedious implementational work and can be avoided altogether with matrix-free methods.

3.4 Quasi-Newton methods

In this section we will present a generalization of the one-dimensional secant method to systems of equations. These methods replace the matrix F', by matrices that are more easily evaluated and/or inverted. The method is derived under a series of simplifying assumptions, which we shall motivate and discuss.

The basic idea is to take the following modification of Newton's method:

$$x_{n+1} = x_n - \mathbb{B}_n^{-1} F(x_n), \tag{3.17}$$

where the matrix B_n will (a) be additively updated in each iteration and (b) should somehow correspond to the secant method in 1D. Concerning the secant method, we can write it in the form

$$b_n = \frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}, \quad x_{n+1} = x_n - b_n^{-1} f(x_n). \tag{3.18}$$

The method (3.17) is clearly a generalization of the second equation in (3.18) to $\mathbb{R}^N$. However, it is unclear how to directly generalize the first equation in (3.18) to the case when $x, F \in \mathbb{R}^N$ and $\mathbb{B} \in \mathbb{R}^{N,N}$. The key is to reformulate the relation for b_n to the form $b_n(x_n - x_{n-1}) = f(x_n) - f(x_{n-1})$. This form can be directly generalized for matrices and vectors to get the so-called ***quasi-Newton condition*** (or secant condition)

$$\mathbb{B}_n(x_n - x_{n-1}) = F(x_n) - F(x_{n-1}). \tag{3.19}$$

We note that this condition does not determine the matrix $\mathbb{B}_n$ uniquely, since (3.19) represents N equations for the N^2 unknown entries of $\mathbb{B}_n$. It is therefore necessary to add additional conditions on B_n to obtain a uniquely determined $\mathbb{B}_n$.

Remark 43. *We note that condition (3.19) is natural for a matrix which should play a similar role as $F'(x_n)$. The Mean value theorem (Lemma 30) states that*

$$\left[\int_0^1 F'(x_{n-1} + t(x_n - x_{n-1}))\,\mathrm{d}t\right](x_n - x_{n-1}) = F(x_n) - F(x_{n-1}), \tag{3.20}$$

which has a similar form as (3.19). Therefore, $\mathbb{B}_n$ plays a similar role as the Jacobi matrix averaged along the line between x_n, x_{n-1}. This is not claiming that the resulting $\mathbb{B}_n$ will be an approximation of F' in some sense, only that it satisfies similar requirements.

Broyden's method

Now we must supply additional requirements on $\mathbb{B}_n$ to get a uniquely determined method. The first such method was ***Broyden's method*** published in 1965. The basic additional assumptions are such: Let $\mathbb{B}_n$ be given, then $\mathbb{B}_{n+1}$ is constructed as follows:

1. The matrices $\mathbb{B}_n$ are constructed by additive updates by some matrices $\mathbb{C}_n$:

$$\mathbb{B}_{n+1} = \mathbb{B}_n + \mathbb{C}_n. \tag{3.21}$$

2. $\mathbb{C}_n$ is a rank-1 matrix:

$$\mathbb{C}_n = a_n b_n^T, \quad a_n, b_n \in \mathbb{R}^N. \tag{3.22}$$

3. $\mathbb{B}_{n+1}$ also satisfies the quasi-Newton condition (3.19)

$$\mathbb{B}_{n+1}\Delta x_n = \Delta F_n, \tag{3.23}$$

where $\Delta x_n = x_{n+1} - x_n$ and $\Delta F_n = F(x_{n+1}) - F(x_n)$.

While assumptions 1 and 3 seem natural, assumption 2 seems arbitrary. The idea is to take $\mathbb{C}_n$ 'as simple as possible', however that is a very vague notion. A less arbitrary explanation of the choice of rank-1 updates is Lemma 45, as we shall see.

If we substitute (3.21) and (3.22) into (3.23), we get the quasi-Newton condition written in terms of the yet unspecified vectors a_n, b_n:

$$a_n b_n^T \Delta x_n = \Delta F_n - \mathbb{B}_n \Delta x_n. \tag{3.24}$$

If we notice that $b_n^T \Delta x_n$ is a number, we can divide (3.24) by it to obtain

$$a_n = \frac{1}{b_n^T \Delta x_n}\big(\Delta F_n - \mathbb{B}_n \Delta x_n\big). \tag{3.25}$$

Therefore, it is sufficient to only specify the vector b_n, because a_n will then be given by formula (3.25).

So it remains to choose a suitable vector b_n. There are many possibilities, here we present two of them. The basic idea is that in the current iteration, the update $\mathbb{C}_n$ should be based on the current information, which is the points x_n, x_{n+1} and the values of F at these points. Therefore, under ideal circumstances, we are building a good approximation along the line joining x_n, x_{n+1} but have no information about the problem in other directions. In other words, $\mathbb{C}_n$ should work as expected 'along the line joining x_n, x_{n+1}', i.e. when applied to the vector Δx_n, but it should do nothing for other vectors, especially vectors in the perpendicular direction. Altogether, the effect of $\mathbb{C}_n$ on vectors $y \in \mathbb{R}^N$ such that $y^T \Delta x_n = 0$ should be zero (we try not to 'invent' some information about the behavior of the problem where we do not know anything). Written mathematically, we require

$$\mathbb{C}_n y = a_n b_n^T y = 0 \text{ for all } y \text{ such that } y^T \Delta x_n = 0.$$

The simplest natural choice satisfying the requirement is $b_n = \Delta x_n$, since then

$$\mathbb{C}_n y = a_n \Delta x_n^T y = a_n y^T \Delta x_n = 0,$$

since $y^T \Delta x_n = 0$. This choice of $b_n = \Delta x_n$ leads to the so-called ***Good Broyden method***. A more general possibility to ensure (3.25) is the choice $b_n = \mathbb{D}\Delta x_n$ for some matrix $\mathbb{D}$ – a specific choice of $\mathbb{D}$ will lead to the so-called ***bad Broyden method***.

If we take $b_n = \Delta x_n$ in (3.25), we get

$$\mathbb{C}_n = a_n b_n^T = \frac{1}{\|\Delta x_n\|^2}\big(\Delta F_n - \mathbb{B}_n \Delta x_n\big)\Delta x_n^T. \tag{3.26}$$

Thus we can write down the final algorithm:

Definition 44 (***'Good' Broyden method***). *Given $x_0, \mathbb{B}_0$, compute for $n = 0, 1, \ldots$*

$$x_{n+1} = x_n - \mathbb{B}_n^{-1} F(x_n)$$
$$\mathbb{B}_{n+1} = \mathbb{B}_n + \frac{1}{\|\Delta x_n\|^2}\big(\Delta F_n - \mathbb{B}_n \Delta x_n\big)\Delta x_n^T,$$

where $\Delta x_n = x_{n+1} - x_n$ and $\Delta F_n = F(x_{n+1}) - F(x_n)$.

Remarks:

- The algorithm needs only one evaluation of F per iteration, similarly as the secant method in 1D.
- The choice $b_n = \mathbb{B}_n^T \Delta x_n$ gives the so-called ***'bad' Broyden method***.
- Broyden's method has ***superlinear convergence***, however, unlike Newton's method, it does not have a uniquely defined convergence rate for all functions F with given regularity. One can show that the best convergence rate in $\mathbb{R}^N$ is the unique positive root of the equation $\rho^{N+1} - \rho^N - 1 = 0$ (compare with the secant method with $N = 1$), which can be shown to be approximately $\sqrt[N]{N}$ as $N \to \infty$. This implies that as $N \to \infty$, the best convergence rate tends to 1 from above. The decrease of the convergence rate with growing dimension is intuitive: The secant equation determines $\mathbb{B}_n$ uniquely only for $N = 1$. As N grows, so does the gap between N (number of equations in the quasi-Newton condition) and N^2 (components of $\mathbb{B}_n$). To bridge this larger and larger gap, we needed to 'make up' simplifying assumptions ($\mathbb{C}_n$ is rank 1, etc.) which are however somewhat artificial, thus decreasing the convergence rate. The highest convergence rate is in 1D, where these additional artificial assumptions are not needed.
- If $F(x) = \mathbb{A}x - b$ is a linear system, then Broyden is a finite method – for any x_0 we get the exact solution in at most N iterations. This is worse than Newton (exact solution in the first iteration), however it is a finite method nonetheless.
- 'Good' Broyden is affine invariant but not contravariant. For 'Bad' Broyden it is exactly opposite.
- The terminology 'good' and 'bad' Broyden method is misleading. It originates from Broyden's original paper, where several numerical experiments are considered. The 'good' version worked better then the 'bad' one on these examples, so Broyden introduced these names. As it turned out later, the situation is not so clear and for other cases the 'bad' variant can actually be better. There are papers on this subject.

Now we return to the question of why we chose the update to be rank 1. We motivated this by simplicity, but this is a relative concept- why not diagonal or unitary or circulant, etc.? A possible answer is given by the following theorem: The rank 1 update turns out to be the smallest possible update such that the updated

matrix satisfies the quasi-Newton condition. Here 'smallest' means 'smallest in the Frobenius norm'. Since the Frobenius norm is the L^2 norm of all components of the matrix, one can say that the rank 1 update is the smallest elementwise update satisfying the Quasi-Newton condition. This means that the rank 1 update is the most conservative – make the smallest possible change to $\mathbb{B}_n$ so that we satisfy the basic assumption, the Quasi-Newton condition. This is reasonable, because, as noted earlier, all the other assumptions are to some extent artificial and we want to minimize their influence to some extent.

Lemma 45. *Let $\Delta x_n, \Delta F_n \in \mathbb{R}^N$ be given. Then the matrix $\mathbb{C}_n$ given by (3.26) is the smallest update of $\mathbb{B}_n$ in the Frobenius norm, such that $\mathbb{B}_{n+1} = \mathbb{B}_n + \mathbb{C}_n$ satisfies the quasi-Newton condition $\mathbb{B}_{n+1}\Delta x_n = \Delta F_n$.*

Proof. Let $\tilde{\mathbb{C}}_n$ be another update such that $\tilde{\mathbb{B}}_{n+1} = \mathbb{B}_n + \tilde{C}_n$ also satisfies $\tilde{\mathbb{B}}_{n+1}\Delta x_n = \Delta F_n$. Then

$$\begin{aligned}
\|\mathbb{C}_n\|_F &= \left\| \frac{1}{\|\Delta x_n\|^2} \big(\Delta F_n - \mathbb{B}_n \Delta x_n\big) \Delta x_n^T \right\|_F \\
&= \left\| \frac{1}{\|\Delta x_n\|^2} \big(\tilde{\mathbb{B}}_{n+1}\Delta x_n - \mathbb{B}_n \Delta x_n\big) \Delta x_n^T \right\|_F \\
&= \frac{1}{\|\Delta x_n\|^2} \big\| \big(\tilde{\mathbb{B}}_{n+1} - \mathbb{B}_n\big) \Delta x_n \Delta x_n^T \big\|_F \\
&\leq \big\| \underbrace{\tilde{\mathbb{B}}_{n+1} - \mathbb{B}_n}_{\tilde{C}_n} \big\|_F \underbrace{\frac{1}{\|\Delta x_n\|^2} \|\Delta x_n \Delta x_n^T\|_F}_{=1} \\
&= \|\tilde{C}_n\|_F,
\end{aligned}$$

which we wanted to prove. Here we have used the property of the Frobenius norm that $\|u\|^2 = \|uu^T\|_F$ for any $u \in \mathbb{R}^N$, which can be proved directly:

$$\|u\|^2 = \sum_{i=1}^N u_i^2 = \sqrt{\Big(\sum_{i=1}^N u_i^2\Big)^2} = \sqrt{\sum_{i,j=1}^N u_i^2 u_j^2} = \sqrt{\sum_{i,j=1}^N (u_i u_j)^2} = \|uu^T\|_F.$$

□

Finally, we come to the last reason, why a rank 1 update should be a wise choice. Using the so-called Sherman-Morrison formula, one can directly compute the inverse $\mathbb{B}_{n+1}^{-1}$ as an update of $\mathbb{B}_n^{-1}$.

Lemma 46 (Sherman-Morrison formula)**.** *Let $\mathbb{A} \in \mathbb{R}^N$ be an invertible matrix and let $u, v \in \mathbb{R}^N$. Then $\mathbb{A} + uv^T$ is invertible iff $1 + v^T\mathbb{A}^{-1}u \neq 0$ and*

$$(\mathbb{A} + uv^T)^{-1} = \mathbb{A}^{-1} - \frac{\mathbb{A}^{-1}uv^T\mathbb{A}^{-1}}{1 + v^T\mathbb{A}^{-1}u}. \tag{3.27}$$

Lemma 46 allows us to simplify Broyden's method significantly, as we can now directly compute $\mathbb{B}_{n+1}$ using updates. A direct application of (3.27) to the 'good'

Broyden method with $u = a_n, v = b_n$ gives us the following algorithm:

$$\begin{aligned} x_{n+1} &= x_n - \mathbb{B}_n^{-1} F(x_n) \\ \mathbb{B}_{n+1}^{-1} &= \mathbb{B}_n^{-1} + \frac{\Delta x_n - \mathbb{B}_n^{-1}\Delta F_n}{\Delta x_n^T \mathbb{B}_n^{-1} \Delta F_n} \Delta x_n^T \mathbb{B}_n^{-1}, \end{aligned} \tag{3.28}$$

which requires only $O(N^2)$ operations per iteration without the necessity to solve linear systems with the matrix $\mathbb{B}_n$.
Remarks:

- In the algorithm (3.28), we never need the matrices $\mathbb{B}_n$, we only use $\mathbb{B}_n^{-1}$.
- The general recommendation is to take $\mathbb{B}_0 = \mathbb{I}$, which is somewhat surprising, since in general $\mathbb{I}$ has nothing in common with F'. However, Broyden very quickly starts accumulating information about F. Alternatively, one can take $\mathbb{B}_0$ as some easily invertible approximation of F', for example the diagonal part of F'.
- Unfortunately, the update of $\mathbb{B}_n$ does not preserve sparsity, so we end up with full matrices. This can be a problem in e.g. finite element methods. There exist sparse versions of Broyden's method which preserve the sparsity structure of F'.
- Unfortunately, the Broyden update of $\mathbb{B}_n$ does not preserve symmetry or positive definiteness: If $\mathbb{B}_n$ is symmetric and/or positive definite, then $\mathbb{B}_{n+1}$ is not.
- The BFGS (Broyden, Fletcher, Goldfarb, Shanno) method uses the update $\mathbb{B}_{n+1} = \mathbb{B}_n + \alpha a a^T + \beta b b^T$ which automatically preserves symmetry, and after choosing the parameters α, β, it also preserves positive definiteness. The BFGS method is one of the most popular methods in optimization, where the role of the Jacobi matrix F' is taken by the Hessian matrix $\mathbb{H}$ of the minimized functional. The point is that Hessian matrices are naturally symmetric, while there is no reason for symmetry of Jacobi matrices. Thus the BFGS is more popular in the optimization community.

3.5 Continuation methods

As we have seen throughout this text, the main disadvantage of more sophisticated methods is their local convergence. Here we present one possible 'globalization' strategy how to try to enlarge the region of convergence. We will describe the basic strategy of ***continuation*** (or homotopy) methods.

The basic idea is to find or introduce an auxiliary parameter on which the problem depends and which controls how 'hard' the problem is. Sometimes this is naturally contained in our equations – a parameter which for some values gives a much simpler problem, while we want to compute the solution to a hard problem for some other value of the parameter. For example, in computational fluid dynamics, the viscosity (or Reynolds number) is such a parameter: for Reynolds number equal to

zero, we get the Stokes problem which is linear and thus much simpler than high Reynolds flows with turbulence etc. Perhaps our specific problem contains such a parameter naturally. If not, we will introduce such a parameter artificially.

In any case, we will call the auxiliary parameter t and we assume that for $t = 0$ we get a simpler (e.g. linear) problem $F_0(x) = 0$ and for $t = 1$, we get the original problem $F(x) = 0$ which was to be solved. So instead of a single function $F : D \subset \mathbb{R}^N \to \mathbb{R}^N$ we have a class of problems $H : D \times [0,1] \subset \mathbb{R}^{N+1} \to \mathbb{R}^N$ such that

$$\begin{aligned} &H(x,0) = F_0(x), \text{ which is a simple problem,} \\ &H(x,1) = F(x), \quad \text{which is the original problem.} \end{aligned} \tag{3.29}$$

If F does not naturally contain such a tuning parameter t, we can take for example the convex combination

$$H(x,t) = tF(x) + (1-t)F_0(x), \tag{3.30}$$

where F_0 is some 'simple' mapping, for which we ideally know the solution to $F_0(x) = 0$. However, F_0 should not be chosen arbitrarily, as the resulting method would have problems. The function F_0 should somehow be related to F. One possible choice is to choose some x_0 and take $F_0(x) = F(x) - F(x_0)$. Then x_0 is trivially the solution to $F_0(x) = 0$. Taking such an F_0 in (3.30) results in

$$H(x,t) = F(x) + (t-1)F(x_0). \tag{3.31}$$

This definition of H clearly satisfies the basic assumption (3.29), since we know the solution x_0 to $H(x,0) = 0$. In the following we will always keep (3.31) in mind as a basic choice of H.

Now instead of the single problem $F(x) = 0$ we will be solving a whole class of problems

$$H(x,t) = 0, \quad \forall t \in [0,1]. \tag{3.32}$$

We assume that (3.32) has a solution $x(t)$ for any $t \in [0,1]$ and that it depends continuously on t. In other words, we assume the existence of a continuous mapping $x : [0,1] \to \mathbb{R}^N$ such that

$$H(x(t),t) = 0, \quad \forall t \in [0,1].$$

In this case, $x(\cdot)$ defines a curve in $\mathbb{R}^N$ such that $x(0) = x_0$ is prescribed and $x(1) = x_*$ is the desired solution to our original problem $F(x_*) = 0$.

Remark 47. *The standard implicit function theorem gives local existence and uniqueness of $x(\cdot)$. The fundamental assumption (apart from regularity) is the regularity of $\partial_x H$ – the Jacobi matrix of H with respect to the x-variables. This can be improved to get a global existence result under additional assumptions. For example, we get the following.*

Lemma 48. *Let $F : \mathbb{R}^N \to \mathbb{R}^N$ be continuously differentiable and let $\|F'(x)^{-1}\| \leq \beta < \infty$ for all $x \in \mathbb{R}^N$. Then for all $x_0 \in \mathbb{R}^N$ there exists a continuously differentiable mapping $x : [0,1] \to \mathbb{R}^N$ such that $H(x(t),t) = 0$ for all $t \in [0,1]$, where H is defined by (3.31).*

Now we come to the basic idea of continuation methods. We want to approximate the curve $x(\cdot)$ from its known starting point x_0 to its unknown endpoint x_*. To this end we choose a partition $0 = t_0 < t_1 < \ldots < t_M = 1$ of $[0, 1]$ and we solve the sequence of intermediate problems

$$H(x, t_i) = 0, \quad i = 1, \ldots, M.$$

We will denote the solution of the i-th problem as x_i, so that $H(x_i, t_i) = 0$. Now since $x(\cdot)$ is continuous, if $t_i - t_{i-1}$ is sufficiently small then x_{i-1} is close to x_i. So close in fact that x_{i-1} can be used as an initial point in a locally convergent method for x_i. In other words, each x_i has a neighborhood where e.g. Newton's method will converge. If these neighborhoods overlap sufficiently, then x_{i-1} (or its approximation) can be used as a good initial guess in Newton's method for solving $H(x, t_i) = 0$ with the solution x_i. The whole process then goes as follows: start with x_0 and do several iterations of Newton for x_1. Then use the current approximation of x_1 and do several iterations of Newton for x_2. And so on. In the end x_{M-1} is a good initial approximation for $x_M = x_*$. The point is that x_0 is a bad initial guess in Newton for x_M, but we use the intermediate solutions x_i to bridge this gap via (possibly very small) neighborhoods of convergence of all the individual x_i. Of coarse, in practice we only do a finite number N_i of iterations of Newton for each x_i. The result is that we need two indexes: $x_{i,n}$, where the first index denotes the problem solved and the second one denotes the Newton iteration index. The resulting algorithm is this:

General continuation algorithm with Newton's method (3.33)

Given x_0, set $x_{1,0} = x_0$.
For $i = 1, \ldots, M-1$:
  For $n = 0, \ldots, N_i - 1$:
    $x_{i,n+1} = x_{i,n} - [\partial_x H(x_{i,n}, t_i)]^{-1} H(x_{i,n}, t_i)$, (Newton for $H(x, t_i) = 0$)
  $x_{i+1,0} = x_{i,N_i}$
For $n = 0, \ldots$ (until stopping criterion is satisfied) do
  $x_{M,n+1} = x_{M,n} - [\partial_x H(x_{M,n}, 1)]^{-1} H(x_{M,n}, 1)$. (Newton for $H(x, 1) = 0$)

Here the last line is simply just Newton's method for $F(x) = 0$. Since we do not need to compute the auxiliary x_i to huge precision, one could simply take $N_i = 1$, i.e. do just one iteration of Newton for each x_i, while taking a finer partition of $[0, 1]$. In this case we can drop the second iteration index and the previous algorithm (3.33) with the choice of H given by (3.31) reduces to

$$\begin{aligned} x_{n+1} &= x_n - F'(x_n)^{-1}\big[F(x_n) + (t_n - 1)F(x_0)\big], && n = 0, 1, \ldots, M-1, \\ x_{n+1} &= x_n - F'(x_n)^{-1}F(x_n), && n = M, M+1 \ldots. \end{aligned}$$

We note that this resulting algorithm is very much similar to Newton's method with a slight change of the Newton formula in the first M iterations. This will also be the case in other similar methods, which result in some strange simple modification of Newton with the ultimate goal of making it converge more globally.

Concerning the "General continuation algorithm with Newton's method" above, we have the following convergence result.

Theorem 49. *Let $H : D \times [0,1] \subset \mathbb{R}^{N+1} \to \mathbb{R}^N$ be continuously differentiable with respect to x. Let there exist a continuous mapping $x : [0,1] \to D$ solving $H(x(t),t) = 0$ for all $t \in [0,1]$. Let $\partial_x H(x(t),t)$ be regular for all $t \in [0,1]$. Then there exists a partition $\{t_i\}_{i=0}^M$ of $[0,1]$ and numbers $\{N_i\}_{i=1}^{M-1}$ such that the sequence $\{x_{i,n}\}$ defined by algorithm (3.33) is well defined and $\lim_{n\to\infty} x_{M,n} = x_*$.*

Predictor-corrector approach

Up to now we have simply taken the current approximation of x_i as an initial guess in Newton's method for x_{i+1}. The question is whether we can do better. What we want to do is take x_i, somehow produce a better initial guess $\hat{x}_{i+1}$ (*predictor* phase) and use that as an initial guess for Newton (*corrector* phase):

1. **Predictor:** Given x_i, find $\hat{x}_{i+1}$ close to x_{i+1}.
2. **Corrector:** Given $\hat{x}_{i+1}$, use a numerical method to find a better approximation of x_{i+1}.

What we have done up to now is called ***classical continuation***, where the predictor phase is simply $\hat{x}_{i+1} = x_i$ and the corrector phase consists of Newton's method. Another possibility is the following.

Tangent continuation method

The idea is to locally approximate the curve $x(\cdot)$ by its tangent and find $\hat{x}_{i+1}$ on the tangent:

$$\hat{x}_{i+1} = x_i + x'(t_i)(t_{i+1} - t_i). \tag{3.34}$$

Now the question is how to obtain $x'(\cdot)$? To this end we simply take the basic relation $H(x(t),t) = 0$ and differentiate with respect to t. We get the so-called ***Davidenko differential equation***

$$\partial_x H(x(t),t)\, x'(t) + \partial_t H(x(t),t) = 0,$$

where $\partial_t H$ is the partial derivative of H with respect to the second variable t. If we assume regularity of the Jacobi matrix $\partial_x H(x(t),t)$, we can express

$$x'(t) = -\partial_x H(x(t),t)^{-1} \partial_t H(x(t),t). \tag{3.35}$$

In the case of H defined by (3.31), we get

$$x'(t) = -F'(x(t))^{-1} F(x_0),$$

therefore the predictor phase (3.34) reduces to

$$\hat{x}_{i+1} = x_i - F'(x_i)^{-1} F(x_0)(t_{i+1} - t_i),$$

which again resembles some simple variation on Newton's formula.

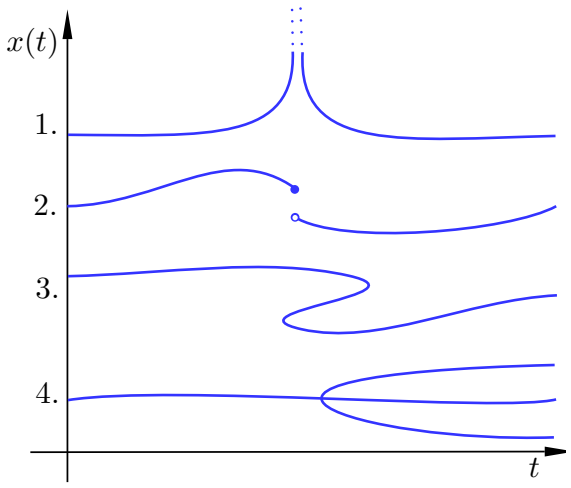

Figure 3.1: Various problems with the continuation path $x(\cdot)$: 1. Escape to infinity, 2. Discontinuity, 3. Turning point, 4. Bifurcation.

Methods based on the Davidenko equation

The original goal was to approximate the endpoint x_* of the curve $x(\cdot)$ given its known initial point x_0. Since we now have a differential equation for $x(\cdot)$, we can discretize the differential equation directly. For example, if we use the forward (explicit) Euler method to discretize (3.35) on the partition $\{t_i\}_{i=0}^M$, we get the scheme

$$\frac{x_{i+1} - x_i}{t_{i+1} - t_i} = -\partial_x H(x_i, t_i)^{-1} \partial_t H(x_i, t_i), \quad i = 0, \dots M-1,$$

or after reformulation as

$$x_{i+1} = x_i - (t_{i+1} - t_i) \partial_x H(x_i, t_i)^{-1} \partial_t H(x_i, t_i).$$

Again, in the case of H given by (3.31), this reduces to

$$x_{i+1} = x_i - (t_{i+1} - t_i) F'(x_i)^{-1} F(x_0).$$

Again this looks like some variation on Newton's method with a damping factor $(t_{i+1} - t_i)$ and $F(x_0)$ instead of $F(x_n)$.

Of coarse one can use more sophisticated methods than Euler's method, for example Runge-Kutta, etc. Such methods appear in the literature and can be analyzed.

We conclude this section by noting what are the basic obstacles in the continuation–based approaches described above. The main problem is the nonexistence of a continuous curve $x(\cdot)$ satisfying the basic assumptions used above. If $H(\cdot, \cdot)$ is chosen poorly, one of several things can go wrong, which prevent the successful application of the methods above:

1. $x(\cdot)$ escapes to infinity and returns back.

2. $x(\cdot)$ is discontinuous.

3. $x(\cdot)$ undergoes a turning point.

4. $x(\cdot)$ undergoes a bifurcation.

These situations are depicted in 1D Figure 3.1 in the presented order. As was mentioned earlier, there are no universal recipes how to treat such problems and there is an extensive literature dealing with these issues.

Exercises

Exercise 15 (Globalization of arctan).

As we have seen in Exercise 9, Newton's method converges only locally to the solution of $\arctan(x) = 0$. Apply any of the continuation strategies described in this section to get a globally convergent method. Try it out.

Bibliography

[OR70] J. M. Ortega and W. C. Rheinboldt, ***Iterative solution of nonlinear equations in several variables***, Computer science and applied mathematics, Academic Press, New York, 1970.

[RR78] A. Ralston and P. Rabinowitz, ***A first course in numerical analysis***, second ed., McGraw-Hill, 1978.

[Seg00] J. Segethová, ***Základy numerické matematiky***, Karolinum, 2000.

[YG88] D.M. Young and R.T. Gregory, ***A survey of numerical mathematics***, vol. 1, Dover Publications, 1988.