Applied Numerical Methods with Matlab (BSc)

Róbert Horváth

Budapest University of Technology and Economics
Faculty of Natural Sciences
Institute of Mathematics
Department of Analysis

spring, 2020

Introduction

Course description

Course description

- Contact: e-mail: rhorvath@math.bme.hu, Office: H.24/b
- Course webpage: anal.math.bme.hu/appnum
- Consultations: office hours: Thursdays 16-17, or by appointment via e-mail
- Course requirements: see the course webpage.
- Lecture notes:
  - slides of the lecture
  - assignments for homework
  - Books:

  Laurene V. Fausett, Applied Numerical Analysis Using Matlab, Pearson Prentice Hall, 2008

  W. Cheney, D. Kincaid, Numerical Mathematics and Computing, Brooks/Cole, Cangage learning, 2013

  Steven C. Chapra, Applied Numerical Methods with MATLAB - for engineers and scientists, McGraw Hill, 2008
  - Catch up with Matlab:

  `https://www.mathworks.com/moler/chapters.html`

  `https://web.stanford.edu/class/ee254/software/using_ml.pdf`

Introduction to numerical analysis

Introduction

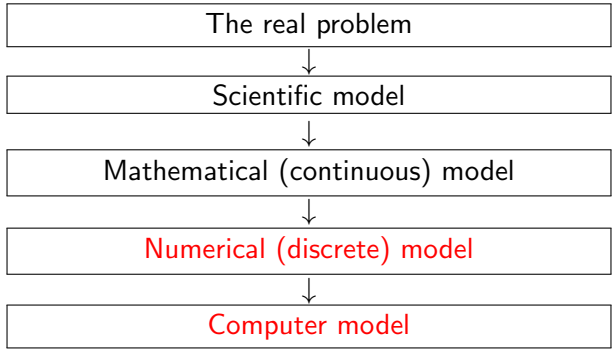
"Numerical analysis is the study of algorithms for the problems of continuous mathematics." (Lloyd N. Trefethen, 1992)

It constructs algorithms and analyses them from the point of view of accuracy, efficiency and its behavior during computer realization.

Problems of continuous mathematics come from different disciplines. They are the mathematical models of e.g. physical, biological, chemical or economical problems.

Introduction

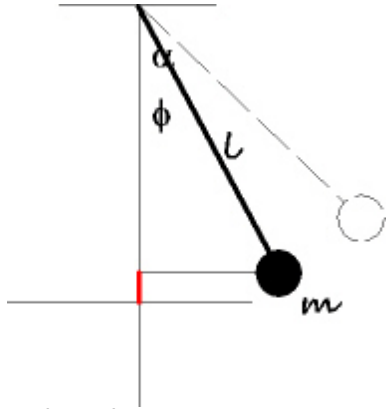
Model construction:



Example of the pendulum motion

Problem: Compute the period of a pendulum.

Sci. mod.: Neglect the weight of the string and the drag. Apply the energy conservation principle: $\frac{1}{2}ml^2(\phi'(t))^2 + mgl(1 - \cos\phi(t)) = mgl(1 - \cos\alpha)$.



Math. mod.: The differential equation for the angular velocity:

$$\phi'(t) = \pm\sqrt{\frac{2g}{l}}\sqrt{\cos\phi(t) - \cos\alpha}$$

The period must be computed from this equation.

Example of the pendulum motion

$$\int_0^{T/4} \frac{\phi'(t)}{-\sqrt{\frac{2g}{l}}\sqrt{\cos\phi(t)-\cos\alpha}}\,\mathrm{d}t = T/4.$$

Changing the variable:

$$T = 2\sqrt{2}\sqrt{\frac{l}{g}}\int_0^{\alpha} \frac{1}{\sqrt{\cos\phi-\cos\alpha}}\,\mathrm{d}\phi$$

$$= 4\sqrt{\frac{l}{g}}\int_0^{\pi/2} \frac{1}{\sqrt{1-\sin^2(\alpha/2)\sin^2\vartheta}}\,\mathrm{d}\vartheta.$$

The value of the integral cannot be given in closed form ($\sin\vartheta = \sin(\phi/2)/\sin(\alpha/2)$).

Num. mod.: Let us use numerical integration formulas (see later).

Comp. mod.: $l = 1m$, $g = 9.8m/s^2$
$T = 2.008035541s$ ($\alpha = 5^\circ$), $T = 2.369049722s$ ($\alpha = 90^\circ$).

Example of the pendulum motion

Other approach: Let us develop the Taylor series of the function $1/\sqrt{1-x}$ about $x = 0$, and let us apply the series at the point $\sin^2(\alpha/2)\sin^2\vartheta$, then let us integrate the formula:

$$T = 2\pi\sqrt{\frac{l}{g}}\left(1 + \frac{1}{4}\sin^2\frac{\alpha}{2} + \dots\right).$$

If we suppose that the initial angular displacement is small, then we obtain the period formula

$$T \approx 2\pi\sqrt{\frac{l}{g}}.$$

This is independent of α. In the example we obtain $T = 2.007089923s$.

Possible error sources

Possible error sources

The real problem

↓ model error, measurement (inherited) error

Scientific model

↓ expression error

Mathematical model

↓ discretization error

Numerical model

↓ rounding error, truncation error

Computer model

Measuring the error with norms

Vector, matrix and function norms

It is highly recommended here to review the summary section about normed spaces → page 359.

If x, y are two elements in a normed space V, then their distance can be measured with the number $\|x - y\|$.

In $\mathbb{R}^n$ we use the following vector norms ($\overline{\mathbf{x}} = [x_1, \ldots, x_n]^T$):

- $\|\overline{\mathbf{x}}\|_1 = |x_1| + \cdots + |x_n|$ (octahedron norm),
- $\|\overline{\mathbf{x}}\|_2 = \sqrt{x_1^2 + \cdots + x_n^2}$ (Euclidean norm),
- $\|\overline{\mathbf{x}}\|_\infty = \max\{|x_1|, \ldots, |x_n|\}$ (maximum norm, $p \to \infty$).

Norms on $\mathbb{R}^{n \times n}$ are called matrix norms. (For the special properties of matrices see the summary section → page 372) Matrix norms can be defined from vector norms with the expression

$$\|\mathbf{A}\| := \sup_{\overline{\mathbf{x}} \neq \overline{\mathbf{0}}} \frac{\|\mathbf{A}\overline{\mathbf{x}}\|}{\|\overline{\mathbf{x}}\|}. \tag{1}$$

This is the so-called induced matrix norm

Vector, matrix and function norms

Thm. 1. Suppose that the matrix norm $\|.\|$ was induced by the vector norm $\|.\|$. Then

- $\|\mathbf{Ax}\| \leq \|\mathbf{A}\| \cdot \|\mathbf{x}\|$, $\forall \overline{\mathbf{x}} \in \mathbb{R}^n$ (consistency),
- $\|\mathbf{I}\| = 1$ ($\mathbf{I}$ is the identity matrix),
- $\|\mathbf{AB}\| \leq \|\mathbf{A}\| \cdot \|\mathbf{B}\|$ (submultiplicity).

Proof. It follows directly from the definition of an induced matrix norm. ■

Thm. 2. The vector norms induce the following matrix norms:

- $p = 1$: $\|\mathbf{A}\|_1 = \max_{j=1,\ldots,n} \sum_{i=1}^{m} |a_{ij}|$,
- $p = \infty$: $\|\mathbf{A}\|_\infty = \max_{i=1,\ldots,m} \sum_{j=1}^{n} |a_{ij}|$,
- $p = 2$: $\|\mathbf{A}\|_2 = \sqrt{\varrho(\mathbf{A}^T\mathbf{A})}$ (ϱ: spectral radius).

Proof. The 1-norm case is proven as an exercise.

Vector, matrix and function norms

Rmk. In the case of symmetric matrices $\mathbf{A} \in \mathbb{R}^{n \times n}$, we have $\|\mathbf{A}\|_2 = \varrho(\mathbf{A})$.

Rmk. The matrix norm $\|\mathbf{A}\| = \max_{i,j}\{|a_{ij}|\}$ is not an induced norm. The so-called Frobenius norm $\|\mathbf{A}\|_F = \sqrt{\sum_{i,j} a_{ij}^2}$ is not an induced norm, too.

The space of the continuous functions defined on $[a, b]$ is denoted with $C[a, b]$. The usual norm of this space, the maximum norm, is defined as follows

$$\|f\|_{C[a,b]} = \max_{x \in [a,b]} \{|f(x)|\}.$$

Thm. 3. For quadratic matrices, the estimation $\varrho(\mathbf{A}) \leq \|\mathbf{A}\|$ is satisfied in any induced norm.

Proof.: Let $\overline{\mathbf{x}} \neq \mathbf{0}$ be an eigenvector of $\mathbf{A}$ and λ be the corresponding eigenvaluue. Then $|\lambda| \cdot \|\overline{\mathbf{x}}\| = \|\lambda\overline{\mathbf{x}}\| = \|\mathbf{A}\overline{\mathbf{x}}\| \leq \|\mathbf{A}\| \cdot \|\overline{\mathbf{x}}\|$. ■

Thm. 4. Let $\mathbf{A} \in \mathbb{R}^{n\times n}$ be a given matrix. Then for any positive $\varepsilon > 0$, there exists an induced norm $\|.\|$, such that $\|\mathbf{A}\| \leq \varrho(\mathbf{A}) + \varepsilon$.

Thm. 5. Let $\mathbf{A} \in \mathbb{R}^{n\times n}$ be a given matrix. $\mathbf{A}^k$ tends to $\mathbf{0}$ elementwise if and only if $\varrho(\mathbf{A}) < 1$. Exactly in the same case, the series

$$\sum_{k=0}^{\infty} \mathbf{A}^k$$

converges, moreover its sum is $(\mathbf{I} - \mathbf{A})^{-1}$.

Thm. 6. If the relation $\|\mathbf{A}\| < 1$ is valid for the matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$ in some induced norm, then the following estimation holds

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

Proof: It follows from the previous theorem that the matrix $\mathbf{I}-\mathbf{A}$ is non-singular.

$$\mathbf{I} = (\mathbf{I}-\mathbf{A})(\mathbf{I}-\mathbf{A})^{-1} \Rightarrow 1 \leq \|\mathbf{I}-\mathbf{A}\|\|(\mathbf{I}-\mathbf{A})^{-1}\|$$
$$\leq (1+\|\mathbf{A}\|)\|(\mathbf{I}-\mathbf{A})^{-1}\| \Rightarrow \text{estimation on the left hand side.}$$

Let us multiply both sides of the equality $\mathbf{I} = \mathbf{I}-\mathbf{A}+\mathbf{A}$ with the inverse of $\mathbf{I}-\mathbf{A}$, then take the norms on both sides.

$$\|(\mathbf{I}-\mathbf{A})^{-1}\| \leq 1 + \|(\mathbf{I}-\mathbf{A})^{-1}\|\,\|\mathbf{A}\|,$$

and after reordering we obtain the inequality on the right hand side. ■

Speed of convergence

Speed of convergence

In iterative methods, the solution is the limit of a specially constructed sequence. Nonlinear equations cannot be solve with direct methods in general. In this case we use iterative methods, that is we generate a sequence that is convergent and its limit is the solution of the equation.

Let us consider the sequence $x_k \to x^\star$. Let $e_k = x_k - x^\star$ be the error of the kth element.

Def. 7. We say that the order of the convergence of the sequence $\{x_k\}$ is the positive real number p if the limit

$$\lim_{k\to\infty} \frac{\|e_{k+1}\|}{\|e_k\|^p} = C \neq 0$$

exists, it is finite and non-zero.

Rmk. If the order of convergence can be defined for a sequence, then it is unique.

Speed of convergence

Rmk. If $p = 1$, then the convergence is linear. If $1 < p < 2$, then the convergence is superlinear. The case $p = 2$ means second order of convergence.

Rmk. If we have a sequence with convergence order p, then for large k values we have the approximation

$$\|e_{k+1}\| \approx C\|e_k\|^p.$$

The logarithm of the equation is

$$\log \|e_{k+1}\| \approx \log C + p \log \|e_k\|.$$

If we graph $\log \|e_{k+1}\|$ against $\log \|e_k\|$, the points falls on a line with slope p that intersects the vertical axis at $\log C$.

This method can be used to check the order of convergence of a sequence (or a method that produces the sequence) empirically.

Example. Both $x_{k+1} = x_k - (2/5)(x_k^2 - 2)$ and $y_{k+1} = y_k - (y_k^2 - 2)/2/y_k$ ($x_0 = y_0 = 3$) tend to $\sqrt{2}$. The first one is order 1 and the second one is order 2.

Machine number format and its corollaries

Some simple examples

MATLAB results:

- $\tan(\pi/2) = 1.6331e+016$
- $2^{-1074}/2 = 0$
- $2^{-1074} = 4.94066e-324$; $2^{-1074} \cdot 1.2 = 4.94066e-324$
- $10^{310} =$Inf
- Let y_k denote the semiperimeter of a regular polygon with 2^k edges inscribed into a circle with radius 1. Then $y_k \to \pi$, if $k \to \infty$. Moreover we have the recursion

$$y_{k+1} = 2^{k+1}\sqrt{\frac{1}{2}\left(1 - \sqrt{1-(2^{-k}y_k)^2}\right)},$$

where $y_1 = 2$, $y_2 = 2\sqrt{2}$, ... , $y_{10} = 3.14158627$, $y_{12} = 3.14166137$, ..., $y_{19} = 3.70727600$, ... Does not tend to π!

Some simple examples

MATLAB results:

- Calculate the following expression in different ways!

$$y = 333.75b^6 + a^2(11a^2b^2 - b^6 - 121b^4 - 2) + 5.5b^8 + \frac{a}{2b},$$

with $a = 77617$ és $b = 33096$.

- Matlab double precision: $y = -1.1806e + 21$
- Matlab double precision without exponents ($a^2 = a * a$, etc.): $y = 1.1726$
- Matlab single precision: $y = -6.3383e + 29$
- Matlab single precision without exponents ($a^2 = a * a$, etc.): $y = 6.3383e + 29$
- Correct answer:

$$\begin{aligned} z &= 333.75b^6 + a^2(11a^2b^2 - b^6 - 121b^4 - 2) \\ &= -79171113406689613611011347015249428 50 \\ x &= 5.5b^8 = 791711134066896136110113470152494284 8 \\ y &= z + x + \frac{a}{2b} = -2 + \frac{77617}{2 \cdot 33096} = -0.827396059946821 \end{aligned}$$

Representation of real numbers in floating point systems

(Konrad Zuse, Berlin, 1930s)

$$\pm b^k \left(\frac{a_0}{b^0} + \frac{a_1}{b^1} + \frac{a_2}{b^2} + \cdots + \frac{a_{p-1}}{b^{p-1}} \right) \equiv a_0.a_1a_2 \ldots a_{p-1} \times b^k$$

- b: base of the representation
- p: the number of the digits in the mantissa
- k: exponent or characteristic
- $0 \leq a_i < b$ integers, $(i = 0, \ldots, p-1)$
- If $a_0 \neq 0$ then the number is in normal form. This is a unique representation.

Illustrative example

`http://www.binaryconvert.com/result_double.html?decimal=048046049`

Representation of real numbers in floating point systems

In the floating point number system we have:

- Only finite number of rational numbers.
- The numbers do not form a field (e.g. the addition is not associative). (Ex.: $123.4 + 0.04 + 0.03 + 0.02 + 0.01$ in different orders in the case $p = 4$, $b = 10$, $k_{max} = 2$)
- The numbers form a bounded set. In the previous example, the largest number is 999.9 (overflow)
- Around zero, there is a relatively large space. The smallest positive representable number in normal form is 0.01. Without the normal form restriction: 0.00001 (underflow).
- The smallest number that is larger then 1 is denoted by $1 + \varepsilon_m$, where ε_m is the so-called machine epsilon. In the example: 0.001.

Double precision floating point numbers

64 bits, binary number system

- The 1. bit stores the sign of the number ($0 = +, 1 = -$).
- The bits 2-12. store the characteristic such that we add 1023 to the exponent and we store the binary version of that number (from -1022 to 1023). The characteristic -1023 stores the 0 (if the mantissa is zero) or indicates that the number is not in normal form ($0.a_1 \dots a_{52} \times 2^{-1022}$). The characteristic coded with all 1s is used for special purposes (mantissa is not zero - NaN, mantissa is zero - $\pm$Inf (depending on the sign bit)).
- The bits 13-64. store the mantissa (the part after the binary point).

Double precision floating point numbers

The largest exactly representable positive number

$$M = 1.\underbrace{111\ldots111}_{52\text{ numbers}} \times 2^{1023} = 1.79769 \times 10^{308}$$

and the smallest positive exactly representable number

$$m = 0.\underbrace{000\ldots000}_{51\text{ numbers}} 1 \times 2^{-1022} = 4.94066 \times 10^{-324}.$$

The smallest positive exactly representable number in normal form

$$\varepsilon_0 = 1.\underbrace{000\ldots000}_{52\text{ numbers}} \times 2^{-1022} = 2.22507 \times 10^{-308}.$$

The smallest exactly representable number next to 1

$$1.\underbrace{000\ldots000}_{51\text{ numbers}} 1 \times 2^{0},$$

which is greater than 1 with $\varepsilon_m = 2^{-52} = 2.22 \times 10^{-16}$

Rounding to floating points

Thm. 8. Let $0 < x \leq M$. Then

$$|fl(x) - x| \leq \begin{cases} m/2, & \text{if } x < m/2, \\ \frac{\varepsilon_m |x|}{2}, & \text{if } m/2 \leq x \leq M. \end{cases}$$

Proof: The first part is trivial. Let us suppose that x is between the floating point numbers x_i and x_j. Let the number of the digits of the mantissa of x_i be p and the characteristic k. Then

$$|fl(x) - x| \leq \frac{x_j - x_i}{2} = \frac{b^{-p+1} b^k}{2} \leq \frac{\varepsilon_m |x|}{2}. \blacksquare$$

The relative error if $m/2 \leq x \leq M$ is

$$\frac{|fl(x) - x|}{|x|} \leq \frac{\varepsilon_m}{2} =: \mathtt{u} \text{ machine precision.}$$

Catastrophic cancellation

This happens by the subtraction of two numbers that are close to each other:

Example. The case of the sequence that should tend to π. The problem can be eliminated with the following reformulation of the iteration:

$$y_{k+1} = y_k \sqrt{\frac{2}{1+\sqrt{1-(2^{-k}y_k)^2}}}.$$

Example.

$$\sqrt{9876} = 9.937806599 \times 10^1, \qquad \sqrt{9875} = 9.937303457 \times 10^1, \quad \text{error} = 10^{-8}\%$$

$$\downarrow$$

$$\sqrt{9876} - \sqrt{9875} = 0.000503142 \times 10^1 = 5.03142 \underbrace{\mathbf{0000}}_{\text{no information}} \times 10^{-3}$$

error=$10^{-4}\%$

Catastrophic cancellation

Better solution:

$$\sqrt{9876} - \sqrt{9875} = \frac{1}{\sqrt{9876} + \sqrt{9875}}$$
$$= 0.005031418679 = 5.031418679 \times 10^{-3}$$

Catastrophic cancellation can occur in those cases when the result is much smaller than the absolute values of the terms summed up.

Example.

$$e^x = \lim_{n \to \infty} \sum_{i=0}^{n} \frac{x^i}{i!}$$

Let $x = -25$. Then $e^{-25} \approx 1.388794 \times 10^{-11}$. The limit of the above sequence according to Matlab is 8.086559×10^{-7}.

Operation count

If floating point operations are the dominant cost then the computation time is proportional to the number of mathematical operations. This is measured in $flop$s. 1 $flop$ is one floating point operation $(-, +, *, /)$.

Def. 9. We say that the sequence $\{a_n\}$ is of order $O(n^\alpha)$ $(\alpha > 0)$ $(n \to \infty)$, if there are constants $n_0 > 0$ and $K > 0$ such that $|a_n| \leq K n^\alpha$ if $n \geq n_0$. Notation: $a_n = O(n^\alpha)$.

Introduction to the solution of systems of linear algebraic equations

Systems of linear algebraic equations

Systems of linear algebraic equations (SLAEs)

- General form ($a_{ij},\ b_i$ are known, find the values x_j)

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

- Vector form

$$x_1\overline{\mathbf{a}}_1 + \cdots + x_n\overline{\mathbf{a}}_n = \overline{\mathbf{b}}$$

- Matrix form

$$\mathbf{A}\overline{\mathbf{x}} = \overline{\mathbf{b}}$$

Thm. 10. A SLAE is solvable iff $r(\mathbf{A}) = r(\mathbf{A}|\overline{\mathbf{b}})$. If it is solvable and $r(\mathbf{A}) < n$, then it has infinitely many solutions, if $r(\mathbf{A}) = n$, then the solution is unique.

Sensibility of the solution

The relative error of the solution

Thm. 11. Let us suppose that, instead of the system $\mathbf{A}\overline{\mathbf{x}} = \overline{\mathbf{b}}$, we solve the system $(\mathbf{A} + \boldsymbol{\delta}\mathbf{A})\overline{\mathbf{y}} = \overline{\mathbf{b}} + \boldsymbol{\delta}\overline{\mathbf{b}}$. The solution is written in the form $\overline{\mathbf{y}} = \overline{\mathbf{x}} + \boldsymbol{\delta}\overline{\mathbf{x}}$. Moreover, let us suppose that $\|\boldsymbol{\delta}\mathbf{A}\| < 1/\|\mathbf{A}^{-1}\|$ in some induced norm. Then the following estimation is true

$$\frac{\|\boldsymbol{\delta}\overline{\mathbf{x}}\|}{\|\overline{\mathbf{x}}\|} \leq \frac{\kappa(\mathbf{A})}{1 - \kappa(\mathbf{A})\|\boldsymbol{\delta}\mathbf{A}\|/\|\mathbf{A}\|} \cdot \left(\frac{\|\boldsymbol{\delta}\overline{\mathbf{b}}\|}{\|\overline{\mathbf{b}}\|} + \frac{\|\boldsymbol{\delta}\mathbf{A}\|}{\|\mathbf{A}\|} \right)$$

where $\kappa(\mathbf{A}) = \|\mathbf{A}\|\|\mathbf{A}^{-1}\|$.

Proof. Since $\|\boldsymbol{\delta}\mathbf{A}\| < 1/\|\mathbf{A}^{-1}\|$, the estimation $\|\mathbf{A}^{-1}\boldsymbol{\delta}\mathbf{A}\| < 1$ holds. Thus, in view of the equality $\mathbf{A} + \boldsymbol{\delta}\mathbf{A} = \mathbf{A}(\mathbf{I} - \mathbf{A}^{-1}\boldsymbol{\delta}\mathbf{A})$ the matrix $\mathbf{A} + \boldsymbol{\delta}\mathbf{A}$ is regular (Theorem 5.). Moreover,

$$\boldsymbol{\delta}\overline{\mathbf{x}} = (\mathbf{A} + \boldsymbol{\delta}\mathbf{A})^{-1}(\overline{\mathbf{b}} + \boldsymbol{\delta}\overline{\mathbf{b}}) - \overline{\mathbf{x}} = (\mathbf{A} + \boldsymbol{\delta}\mathbf{A})^{-1}(\overline{\mathbf{b}} + \boldsymbol{\delta}\overline{\mathbf{b}} - (\mathbf{A} + \boldsymbol{\delta}\mathbf{A})\overline{\mathbf{x}})$$

$$= (\mathbf{A} + \boldsymbol{\delta}\mathbf{A})^{-1}(\boldsymbol{\delta}\overline{\mathbf{b}} - \boldsymbol{\delta}\mathbf{A}\overline{\mathbf{x}}) = (\mathbf{I} + \mathbf{A}^{-1}\boldsymbol{\delta}\mathbf{A})^{-1}\mathbf{A}^{-1}(\boldsymbol{\delta}\overline{\mathbf{b}} - \boldsymbol{\delta}\mathbf{A}\overline{\mathbf{x}}).$$

The relative error of the solution

Let us apply Theorem 6.

$$\|\boldsymbol{\delta}\overline{\mathbf{x}}\| \leq \frac{\|\mathbf{A}^{-1}\|}{1-\|\mathbf{A}^{-1}\boldsymbol{\delta}\mathbf{A}\|}(\|\boldsymbol{\delta}\overline{\mathbf{b}}\| + \|\boldsymbol{\delta}\mathbf{A}\| \cdot \|\overline{\mathbf{x}}\|)$$

$$= \frac{\|\mathbf{A}^{-1}\| \cdot \|\mathbf{A}\|}{1-\|\mathbf{A}^{-1}\boldsymbol{\delta}\mathbf{A}\|}\left(\frac{\|\boldsymbol{\delta}\overline{\mathbf{b}}\|}{\|\mathbf{A}\|} + \frac{\|\boldsymbol{\delta}\mathbf{A}\| \cdot \|\overline{\mathbf{x}}\|}{\|\mathbf{A}\|}\right).$$

We obtain

$$\frac{\|\boldsymbol{\delta}\overline{\mathbf{x}}\|}{\|\overline{\mathbf{x}}\|} \leq \frac{\|\mathbf{A}^{-1}\| \cdot \|\mathbf{A}\|}{1-\|\mathbf{A}^{-1}\boldsymbol{\delta}\mathbf{A}\|}\left(\frac{\|\boldsymbol{\delta}\overline{\mathbf{b}}\|}{\|\mathbf{A}\| \cdot \|\overline{\mathbf{x}}\|} + \frac{\|\boldsymbol{\delta}\mathbf{A}\|}{\|\mathbf{A}\|}\right)$$

$$\leq \frac{\kappa(\mathbf{A})}{1-\kappa(\mathbf{A})\|\boldsymbol{\delta}\mathbf{A}\|/\|\mathbf{A}\|} \cdot \left(\frac{\|\boldsymbol{\delta}\overline{\mathbf{b}}\|}{\|\overline{\mathbf{b}}\|} + \frac{\|\boldsymbol{\delta}\mathbf{A}\|}{\|\mathbf{A}\|}\right). \blacksquare$$

Condition number of matrices

Condition number of matrices

Let us notice that if the coefficients of a SLAE are changed with a small amount, then the solution can change with a relatively large amount if the parameter $\kappa(\mathbf{A})$ is large.

Def. 12. Let $\mathbf{A} \in \mathbb{R}^{n\times n}$ be a regular matrix. Then the number $\kappa(\mathbf{A}) = \|\mathbf{A}\| \cdot \|\mathbf{A}^{-1}\|$ is called the condition number of the matrix. (Its value depends also on the norm!)

The properties of the condition number in induced norm:

- $\kappa(\mathbf{A}) \geq 1$ $(1 = \|\mathbf{I}\| = \|\mathbf{A}\mathbf{A}^{-1}\| \leq \|\mathbf{A}\| \cdot \|\mathbf{A}^{-1}\|)$,
- $\kappa(\mathbf{A}) = \kappa(\mathbf{A}^{-1})$,
- $\kappa(\alpha\mathbf{A}) = \kappa(\mathbf{A})$, $\alpha \neq 0$,
- For orthogonal matrices: $\kappa_2(\mathbf{A}) = 1$ $(\|\mathbf{A}\|_2 = \|\mathbf{A}^{-1}\|_2 = 1)$,
- For symmetric matrices: $\kappa(\mathbf{A}) \geq |\lambda_{\max}/\lambda_{\min}|$, moreover $\kappa_2(\mathbf{A}) = |\lambda_{\max}/\lambda_{\min}|$ ($\lambda_{\max}$, $\lambda_{\min}$: eigenvalues with the maximal and minimal absolute value).

Hilbert matrix

This is an example for a very badly conditioned matrix:

Hilbert matrix: $\mathbf{H}_n \in \mathbb{R}^{n\times n}$, $(\mathbf{H}_n)_{i,j} = 1/(i+j-1)$.

$$\mathbf{H}_6 = \begin{bmatrix} 1 & 1/2 & 1/3 & 1/4 & 1/5 & 1/6 \\ 1/2 & 1/3 & 1/4 & 1/5 & 1/6 & 1/7 \\ 1/3 & 1/4 & 1/5 & 1/6 & 1/7 & 1/8 \\ 1/4 & 1/5 & 1/6 & 1/7 & 1/8 & 1/9 \\ 1/5 & 1/6 & 1/7 & 1/8 & 1/9 & 1/10 \\ 1/6 & 1/7 & 1/8 & 1/9 & 1/10 & 1/11 \end{bmatrix}$$

Example. $\kappa_2(\mathbf{H}_6) \approx 1.6 \times 10^7$, $\kappa_2(\mathbf{H}_{10}) \approx 3.5 \times 10^{13}$.

Solution methods of SLAEs

Solution methods of SLAEs

- Direct methods: They give exact solutions in finitely many steps. (Cramer's rule $x_i = \det\mathbf{A}_i/\det\mathbf{A}$ ($\mathbf{A}_i$-t can be obtained by changing the ith column of $\mathbf{A}$ to $\overline{\mathbf{b}}$), $\overline{\mathbf{x}} = \mathbf{A}^{-1}\overline{\mathbf{b}}$, Gaussian method and its variants)
- Iterative methods: they form a vector sequence that tends to the solution of the system (Gauss–Seidel, Jacobi, SOR). Important question is that when to step the iteration process.

Direct methods of SLAEs

Gaussian method

Gaussian method

$$
\begin{aligned}
a_{11}x_1 + \cdots + a_{1n}x_n &= b_1 \\
a_{21}x_1 + \cdots + a_{2n}x_n &= b_2 \\
&\vdots \\
a_{n1}x_1 + \cdots + a_{nn}x_n &= b_n
\end{aligned}
$$

Carl Friedrich Gauss
(1777-1855)

Possible transformations that do not alter the solution:

- Multiplication of one equation with a constant ($\neq 0$).
- Addition of one equation to another one.
- Interchange of two equations.
- Interchange of two unknowns.

Gaussian method

The coefficient matrix and the right hand side of the system:

$$\left(\begin{array}{cccc|c} a_{11} & a_{12} & \dots & a_{1n} & b_1 \\ a_{21} & a_{22} & \dots & a_{2n} & b_2 \\ a_{31} & a_{32} & \dots & a_{3n} & b_3 \\ \vdots & & & & \\ a_{n1} & a_{n2} & \dots & a_{nn} & b_n \end{array}\right)$$

Gaussian method

The initial matrix of the elimination $[\mathbf{A}^{(1)}|\overline{\mathbf{b}}^{(1)}]$:

$$\left.\begin{array}{cccc|c} a_{11}^{(1)} & a_{12}^{(1)} & \dots & a_{1n}^{(1)} & b_1^{(1)} \\ a_{21}^{(1)} & a_{22}^{(1)} & \dots & a_{2n}^{(1)} & b_2^{(1)} \\ a_{31}^{(1)} & a_{32}^{(1)} & \dots & a_{3n}^{(1)} & b_3^{(1)} \\ \vdots & & & & \\ a_{n1}^{(1)} & a_{n2}^{(1)} & \dots & a_{nn}^{(1)} & b_n^{(1)} \end{array}\right.$$

Gaussian method

The elimination of the first column:

$$\left(\begin{array}{cccc|c} a_{11}^{(1)} & a_{12}^{(1)} & \dots & a_{1n}^{(1)} & b_1^{(1)} \\ a_{21}^{(1)} & a_{22}^{(1)} & \dots & a_{2n}^{(1)} & b_2^{(1)} \\ a_{31}^{(1)} & a_{32}^{(1)} & \dots & a_{3n}^{(1)} & b_3^{(1)} \\ \vdots & & & & \\ a_{n1}^{(1)} & a_{n2}^{(1)} & \dots & a_{nn}^{(1)} & b_n^{(1)} \end{array}\right)$$

$l_{21} = a_{21}^{(1)}/a_{11}^{(1)}, \dots, l_{n1} = a_{n1}^{(1)}/a_{11}^{(1)}$

Gaussian method

The elimination of the first column:

$$\left(\begin{array}{cccc|c} a_{11}^{(1)} & a_{12}^{(1)} & \dots & a_{1n}^{(1)} & b_1^{(1)} \\ 0 & a_{22}^{(1)} - l_{21}a_{12}^{(1)} & \dots & a_{2n}^{(1)} - l_{21}a_{1n}^{(1)} & b_2^{(1)} - l_{21}b_1 \\ 0 & a_{32}^{(1)} - l_{31}a_{12}^{(1)} & \dots & a_{3n}^{(1)} - l_{31}a_{1n}^{(1)} & b_3^{(1)} - l_{31}b_1 \\ \vdots & & & & \\ 0 & a_{n2}^{(1)} - l_{n1}a_{12}^{(1)} & \dots & a_{nn}^{(1)} - l_{n1}a_{1n}^{(1)} & b_n^{(1)} - l_{n1}b_1 \end{array}\right)$$

Gaussian method

The elimination of the first column $[\mathbf{A}^{(2)}|\overline{\mathbf{b}}^{(2)}]$:

$$\left[\begin{array}{cccc|c} a_{11}^{(1)} & a_{12}^{(1)} & \dots & a_{1n}^{(1)} & b_1^{(1)} \\ 0 & a_{22}^{(2)} & \dots & a_{2n}^{(2)} & b_2^{(2)} \\ 0 & a_{32}^{(2)} & \dots & a_{3n}^{(2)} & b_3^{(2)} \\ \vdots & & & & \\ 0 & a_{n2}^{(2)} & \dots & a_{nn}^{(2)} & b_n^{(2)} \end{array}\right]$$

Gaussian method

The elimination of the second column:

$$\left(\begin{array}{cccc|c} a_{11}^{(1)} & a_{12}^{(1)} & \dots & a_{1n}^{(1)} & b_1^{(1)} \\ 0 & a_{22}^{(2)} & \dots & a_{2n}^{(2)} & b_2^{(2)} \\ 0 & a_{32}^{(2)} & \dots & a_{3n}^{(2)} & b_3^{(2)} \\ \vdots & & & & \\ 0 & a_{n2}^{(2)} & \dots & a_{nn}^{(2)} & b_n^{(2)} \end{array}\right)$$

$l_{32} = a_{32}^{(2)}/a_{22}^{(2)}, \dots, l_{n2} = a_{n2}^{(2)}/a_{22}^{(2)}$

Gaussian method

The elimination of the second column:

$$\left(\begin{array}{cccc|c} a_{11}^{(1)} & a_{12}^{(1)} & \dots & a_{1n}^{(1)} & b_1^{(1)} \\ 0 & a_{22}^{(2)} & \dots & a_{2n}^{(2)} & b_2^{(2)} \\ 0 & 0 & \dots & a_{3n}^{(2)} - l_{32}a_{2n}^{(2)} & b_3^{(2)} - l_{32}b_2^{(2)} \\ \vdots & & & & \\ 0 & 0 & \dots & a_{nn}^{(2)} - l_{n2}a_{2n}^{(2)} & b_n^{(2)} - l_{n2}b_2^{(2)} \end{array}\right)$$

Gaussian method

The elimination of the second column $[\mathbf{A}^{(3)}|\overline{\mathbf{b}}^{(3)}]$:

$$\left[\begin{array}{cccc|c} a_{11}^{(1)} & a_{12}^{(1)} & \dots & a_{1n}^{(1)} & b_1^{(1)} \\ 0 & a_{22}^{(2)} & \dots & a_{2n}^{(2)} & b_2^{(2)} \\ 0 & 0 & \dots & a_{3n}^{(3)} & b_3^{(3)} \\ \vdots & & & & \\ 0 & 0 & \dots & a_{nn}^{(3)} & b_n^{(3)} \end{array}\right]$$

Gaussian method

After the elimination of the $(n-1)$st column, we obtain the form $[\mathbf{A}^{(n)}|\overline{\mathbf{b}}^{(n)}]$:

$$\left(\begin{array}{cccc|c} a_{11}^{(1)} & a_{12}^{(1)} & \dots & a_{1n}^{(1)} & b_1^{(1)} \\ 0 & a_{22}^{(2)} & \dots & a_{2n}^{(2)} & b_2^{(2)} \\ 0 & 0 & \dots & a_{3n}^{(3)} & b_3^{(3)} \\ \vdots & & & & \\ 0 & 0 & \dots & a_{nn}^{(n)} & b_n^{(n)} \end{array}\right)$$

Gaussian method

Back substitution:

$$\begin{aligned} a_{11}^{(1)}x_1 + a_{12}^{(1)}x_2 + \cdots + a_{1n}^{(1)}x_n &= b_1^{(1)} \\ a_{22}^{(2)}x_2 + \cdots + a_{2n}^{(2)}x_n &= b_2^{(2)} \\ &\vdots \\ a_{nn}^{(n)}x_n &= b_n^{(n)} \end{aligned}$$

Gaussian method

Back substitution:

$$\begin{aligned} a_{11}^{(1)}x_1 + a_{12}^{(1)}x_2 + \cdots + a_{1n}^{(1)}x_n &= b_1^{(1)} \\ a_{22}^{(2)}x_2 + \cdots + a_{2n}^{(2)}x_n &= b_2^{(2)} \\ &\vdots \\ a_{nn}^{(n)}x_n &= b_n^{(n)} \\ \rightarrow x_n = b_n^{(n)}/a_{nn}^{(n)} & \end{aligned}$$

Gaussian method

Back substitution:

$$
\begin{aligned}
a_{11}^{(1)}x_1 + a_{12}^{(1)}x_2 + \cdots + a_{1n}^{(1)}x_n &= b_1^{(1)} \\
a_{22}^{(2)}x_2 + \cdots + a_{2n}^{(2)}x_n &= b_2^{(2)} \\
\rightarrow x_2 = (b_2^{(2)} - x_n a_{2n}^{(2)} - \cdots - x_3 a_{23}^{(2)})/a_{22}^{(2)} & \\
\vdots & \\
a_{nn}^{(n)}x_n &= b_n^{(n)} \\
\rightarrow x_n = b_n^{(n)}/a_{nn}^{(n)} &
\end{aligned}
$$

Gaussian method

Back substitution:

$$
\begin{aligned}
a_{11}^{(1)}x_1 + a_{12}^{(1)}x_2 + \cdots + a_{1n}^{(1)}x_n &= b_1^{(1)} \\
\rightarrow x_1 = (b_1^{(1)} - x_n a_{1n}^{(1)} - \cdots - x_2 a_{12}^{(1)})/a_{11}^{(1)} & \\
a_{22}^{(2)}x_2 + \cdots + a_{2n}^{(2)}x_n &= b_2^{(2)} \\
\rightarrow x_2 = (b_2^{(2)} - x_n a_{2n}^{(2)} - \cdots - x_3 a_{23}^{(2)})/a_{22}^{(2)} & \\
\vdots & \\
a_{nn}^{(n)}x_n &= b_n^{(n)} \\
\rightarrow x_n = b_n^{(n)}/a_{nn}^{(n)} &
\end{aligned}
$$

Gaussian method

The procedure can be carried out in the present form only if the constants $a_{11}^{(1)}, \ldots, a_{nn}^{(n)}$, the so-called pivot elements are not zeros.

The two phases of the algorithm:

- Elimination process
- Back substitution (solution of a SLAE with a triangular coefficient matrix)

Example. Solve the SLAE.

$$\begin{aligned} x_1 + 1/2x_2 + 1/3x_3 &= 11/6 \\ 1/2x_1 + 1/3x_2 + 1/4x_3 &= 13/12 \\ 1/3x_1 + 1/4x_2 + 1/5x_3 &= 47/60 \end{aligned}$$

Solution: $x_1 = x_2 = x_3 = 1$.

Investigation of the Gaussian method

The algorithm of the Gaussian method

Gaussian method, SLAE given with the matrix $[\mathbf{A}|\overline{\mathbf{b}}] = [\bar{a}_{ij}]_{n\times(n+1)}$.

for k:=1:n-1 **do**
  for i:=k+1:n **do**
    $l_{ik} := \bar{a}_{ik}/\bar{a}_{kk}$
    for j:=k+1:n+1 **do**
      $\bar{a}_{ij} := \bar{a}_{ij} - l_{ik} \cdot \bar{a}_{kj}$
    end for
  end for
end for
$x_n := \bar{a}_{n,n+1}/\bar{a}_{nn}$
for k:=n-1:-1:1 **do**
  $x_k := \bar{a}_{k,n+1}$
  for j:=k+1:n **do**
    $x_k := x_k - \bar{a}_{kj} \cdot x_j$
  end for
  $x_k := x_k/\bar{a}_{kk}$
end for

Gaussian transformation

Let $\bar{\mathbf{l}}_k = [0, \ldots, 0, l_{k+1,k}, \ldots, l_{n,k}]^T \in \mathbb{R}^n$ $(k = 1, \ldots, n-1)$. Then the kth step of the Gaussian elimination can be written as the matrix multiplication from left with the matrix $\mathbf{L}_k := \mathbf{I} - \bar{\mathbf{l}}_k \bar{\mathbf{e}}_k^T$.

It is easy to see that $(\mathbf{I} - \bar{\mathbf{l}}_k \bar{\mathbf{e}}_k^T)^{-1} = \mathbf{I} + \bar{\mathbf{l}}_k \bar{\mathbf{e}}_k^T$.

The performance of the Gaussian method

Thm. 13. The Gaussian method can be performed with the previous algorithm iff all leading principal minors of $\mathbf{A}$ are non-zero, that is $\det(\mathbf{A}(1:k,1:k)) \neq 0$ $(k = 1, \ldots, n)$.

Proof: During the Gaussian elimination process we add some rows of the matrix to other rows. This procedure does not modify the determinant of the matrix. Thus

$$\begin{aligned}
\det(\mathbf{A}(1:1,1:1)) &= \det(\mathbf{A}^{(1)}(1:1,1:1)) = a_{11}^{(1)} \neq 0, \\
\det(\mathbf{A}(1:2,1:2)) &= \det(\mathbf{A}^{(2)}(1:2,1:2)) = a_{11}^{(1)} a_{22}^{(2)} \neq 0, \\
&\vdots \\
\det(\mathbf{A}(1:n,1:n)) &= \det(\mathbf{A}^{(n)}(1:n,1:n)) = a_{11}^{(1)} a_{22}^{(2)} \ldots a_{nn}^{(n)} \neq 0.
\end{aligned}$$

(We need the last condition because of the back substitution.)
This implies the statement. ■.

Performance of the Gaussian method

Thm. 14. If the coefficient matrix $\mathbf{A}$ of the SLAE

- has a strictly dominant diagonal,
- is symmetric positive definite,
- M-matrix,

then the Gaussian method can be realized with the previous algorithm.

We introduce M-matrices.

Def. 15. We call a matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$ to be an M-matrix if all its offdiagonal elements are nonpositive, it is regular and $\mathbf{A}^{-1} \geq \mathbf{0}$.

Example.

$$\mathbf{A} = \begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix}, \quad \mathbf{A}^{-1} = \begin{bmatrix} 3/4 & 1/2 & 1/4 \\ 1/2 & 1 & 1/2 \\ 1/4 & 1/2 & 3/4 \end{bmatrix}.$$

Performance of the Gaussian method - M-matrices

Thm. 16. The elements of the main diagonal of an M-matrix are positive.

Proof: If $a_{ii} \leq 0$, then $\mathbf{A}\overline{\mathbf{e}}_i \leq \mathbf{0}$. In this case $\overline{\mathbf{e}}_i \leq \mathbf{0}$, because $\mathbf{A}^{-1} \geq \mathbf{0}$, which is a contradiction. ■

Thm. 17. If $\mathbf{A}$ is an M-matrix, then there is a positive vector $\overline{\mathbf{g}} > \mathbf{0}$ such that $\mathbf{A}\overline{\mathbf{g}} > \mathbf{0}$.

Proof: Let $\overline{\mathbf{e}} = [1, \ldots, 1]^T$. Then $\overline{\mathbf{g}} = \mathbf{A}^{-1}\overline{\mathbf{e}}$ is a good choice because all elements are positive and $\mathbf{A}\overline{\mathbf{g}} = \mathbf{A}\mathbf{A}^{-1}\overline{\mathbf{e}} = \overline{\mathbf{e}} > \mathbf{0}$. ■

The converse of the theorem is also true in the following form.

Thm. 18. If a vector $\overline{\mathbf{g}} > 0$ exists with the property $\mathbf{A}\overline{\mathbf{g}} > 0$ and the offdiagonal of $\mathbf{A}$ is non-positive, then $\mathbf{A}$ is an M-matrix.

Performance of the Gaussian method - M-matrices

Thm. 19. Let $\mathbf{A}$ be an M-matrix and $\overline{\mathbf{g}}$ a vector for which the condition of the above theorem is valid. Then

$$\|\mathbf{A}^{-1}\|_\infty \leq \frac{\|\overline{\mathbf{g}}\|_\infty}{\min_i(\mathbf{A}\overline{\mathbf{g}})_i}.$$

Proof: Let $\mathbf{A}\overline{\mathbf{g}} = \overline{\mathbf{y}} > \mathbf{0}$. Then

$$(\min_i y_i)\|\mathbf{A}^{-1}\|_\infty \leq \|\mathbf{A}^{-1}\overline{\mathbf{y}}\|_\infty = \|\overline{\mathbf{g}}\|_\infty,$$

from which the statement follows directly. ■

Operation count for the Gaussian method

Operation count

Operation count for the elimination:

$$\frac{2(n-1)n(2n-1)}{6} + \frac{3(n-1)n}{2}$$

$$= \frac{4n^3 + 3n^2 - 7n}{6} = \frac{2}{3}n^3 + O(n^2)\ flop$$

Operation count for the back substitution: $1 + 3 + \cdots + 2n - 1 = n^2\ flop$

Altogether:

$$\frac{2}{3}n^3 + O(n^2)$$

For large matrices the number of operations for the back substitution is negligible compared to that for the elimination.

Operation count

For triangular matrices: n^2 (only back substitution).

For tridiagonal matrices: $8n - 7$.

Rmk. If we computed the solution $\overline{\mathbf{x}}$ with the formula $\overline{\mathbf{x}} = \mathbf{A}^{-1}\overline{\mathbf{b}}$ (suppose that we know the inverse somehow), then the number of operations would be $2n^2 - n$.

LU decomposition

LU decomposition

Thm. 20. Let us suppose that for the matrix $\mathbf{A}$ the condition $\det(\mathbf{A}(1:k,1:k)) \neq 0$ $(k = 1,\ldots,n-1)$ is fulfilled, that is the Gaussian elimination method can be performed for this matrix. Then there exist a normed lower triangular matrix $\mathbf{L}$ (lower) (1s are in the main diagonal) and an upper triangular matrix $\mathbf{U}$ such that $\mathbf{A} = \mathbf{LU}$ (LU decomposition). If the regular matrix $\mathbf{A}$ has an LU decomposition, then the decomposition is unique, moreover $\det(\mathbf{A}) = u_{11} \ldots u_{nn}$.

Proof: During the Gaussian elimination process the Gaussian transformations change the matrix $\mathbf{A}$ as follows:

$$\mathbf{L}_{n-1}\mathbf{L}_{n-2}\ldots\mathbf{L}_1\mathbf{A} = \mathbf{U},$$

where $\mathbf{U}$ is the upper triangular matrix obtained after the elimination process.

LU decomposition

Because $(\mathbf{I} - \bar{\mathbf{l}}_k \bar{\mathbf{e}}_k^T)^{-1} = \mathbf{I} + \bar{\mathbf{l}}_k \bar{\mathbf{e}}_k^T$ and $\bar{\mathbf{l}}_k \bar{\mathbf{e}}_k^T \bar{\mathbf{l}}_l \bar{\mathbf{e}}_l^T = \mathbf{0}$ if $l > k$, the matrix $\mathbf{A}$ can be written in the form

$$\mathbf{A} = \mathbf{L}_1^{-1} \dots \mathbf{L}_{n-2}^{-1} \mathbf{L}_{n-1}^{-1} \mathbf{U} = \left(\prod_{k=1}^{n-1} (\mathbf{I} + \bar{\mathbf{l}}_k \bar{\mathbf{e}}_k^T) \right) \mathbf{U}$$

$$= \underbrace{\left(\mathbf{I} + \sum_{k=1}^{n-1} \bar{\mathbf{l}}_k \bar{\mathbf{e}}_k^T \right)}_{\text{lower normed triang. matrix}} \mathbf{U} = \mathbf{L}\mathbf{U}.$$

The calculation of the determinant of the matrix $\mathbf{A}$:

$$\det(\mathbf{A}) = \det(\mathbf{L}) \det(\mathbf{U}) = u_{11} \dots u_{nn}.$$

LU decomposition

Uniqueness:
Let us suppose that there are two different decompositions: $\mathbf{A} = \tilde{\mathbf{L}}\tilde{\mathbf{U}} = \mathbf{L}\mathbf{U}$. Then

$$\tilde{\mathbf{L}}^{-1}\mathbf{L} = \tilde{\mathbf{U}}\mathbf{U}^{-1} = \mathbf{I},$$

because the product of normed lower triangular matrices is normed lower triangular and similar statement is true for upper triangular matrices. ■

Rmk. The matrix $\mathbf{U}$ is the upper triangular matrix that is formed during the elimination process, matrix $\mathbf{L}$ is the matrix of the l_{ij} coefficients

$$\mathbf{L} = \begin{bmatrix} 1 & 0 & \dots & 0 \\ l_{21} & 1 & \dots & 0 \\ l_{31} & l_{32} & \dots & 0 \\ \vdots & & & \\ l_{n1} & l_{n2} & \dots & 1 \end{bmatrix}.$$

Corollary: If one of the leading principal minors of a regular matrix is zero, then the matrix does not have LU decomposition.

LU decomposition

Example.

$$\begin{bmatrix} 1 & 1/2 & 1/3 \\ 1/2 & 1/3 & 1/4 \\ 1/3 & 1/4 & 1/5 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 1/2 & 1 & 0 \\ 1/3 & 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1/2 & 1/3 \\ 0 & 1/12 & 1/12 \\ 0 & 0 & \frac{1}{180} \end{bmatrix}$$

Remarks:

- If we have computed the LU decomposition of $\mathbf{A}$, then the matrices $\mathbf{L}$ and $\mathbf{U}$ can be stored in the computer memory in the place of $\mathbf{A}$. The SLAE $\mathbf{A}\overline{\mathbf{x}} = \overline{\mathbf{b}}$ can be solved with the solution of two SLAEs with triangular coefficient matrices. Operation: $2n^2 << 2n^3/3$.
- We generally do not calculate the inverse of matrices! If we need to do this, then we can perform this task with the expression $\mathbf{U}^{-1}\mathbf{L}^{-1}$ or using the Gauss–Jordan method. The number of operations is $2n^3 + O(h^2)$ in both cases.

Pivoting

Pivoting

The Gauss method can be performed only if the pivot elements are not zero. What should we do if $a_{kk}^{(k)}$ is zero?

- Let us choose a non-zero element from the column $\mathbf{A}(k+1:n,k)$. Let us denote the row index of this element by s. Let us swap the kth and the sth rows (change of indexes), then let us continue the elimination.
- If there is no non-zero element in the column $\mathbf{A}(k+1:n,k)$, then the first k columns are linearly dependent, thus $\det(\mathbf{A}) = 0$. In this case there is no unique solution.
- Partial pivoting: It can be a good idea to decrease the elements of $\mathbf{L}$ in absolute value. In view of the form $l_{sk} = a_{sk}^{(k)}/a_{kk}^{(k)}$, the error can be decreased by choosing the largest element in absolute value to be the pivot element. The number of the required operations is $(n^2-n)/2$ comparisons.

Pivoting

- Full pivoting: In the kth step we choose the greatest element in absolute value from the sub-matrix $\mathbf{A}(k:n, k:n)$. This is $n(n+1)(2n+1)/6 - 1 = n^3/3 + O(n^2)$ comparisons.

Let us consider the problem, and let us round to 4 significant digits.

$$\begin{aligned} 0.003x_1 + 59.14x_2 &= 59.17 \\ 5.291x_1 - 6.13x_2 &= 46.78 \end{aligned}$$

Exact solution $x_1 = 10.00$, $x_2 = 1.000$. Without pivoting, we obtain $x_1 = -10$, $x_2 = 1.001$ (cancellation), with partial pivoting we obtain the exact solution.

$\mathbf{LDM}^T$ decomposition

$\mathbf{LDM}^T$ decomposition

Thm. 21. Let us suppose that all leading principal minors of $\mathbf{A}$ are non-zero. Then there exist the unique normed lower triangular matrices $\mathbf{L}$ and $\mathbf{M}$ and the diagonal matrix $\mathbf{D}$ such that $\mathbf{A} = \mathbf{LDM}^T$.

Proof: The LU decomposition is performable. Let $\mathbf{D}$ be such that $d_{ii} = u_{ii} (\neq 0)$. Then the matrix $\mathbf{M} = (\mathbf{D}^{-1}\mathbf{U})^T$ is a normed lower triangular matrix. Moreover $\mathbf{LD}(\mathbf{D}^{-1}\mathbf{U}) = \mathbf{LU} = \mathbf{A}$. The uniqueness follows from the uniqueness of the LU decomposition. ■

Thm. 22. For symmetric matrices $\mathbf{A}$, there exists a unique normed lower triangular matrix $\mathbf{L}$ and a diagonal matrix $\mathbf{D}$ such that $\mathbf{A} = \mathbf{LDL}^T$.

Proof: The matrix $\mathbf{M}^{-1}\mathbf{A}\mathbf{M}^{-\top} = \mathbf{M}^{-1}\mathbf{LD}$ is symmetric and lower triangular $\Rightarrow$ diagonal. $\det(\mathbf{D}) \neq 0 \Rightarrow \mathbf{M}^{-1}\mathbf{L}$ is also diagonal but also normed lower triangular. That is $\mathbf{M}^{-1}\mathbf{L} = \mathbf{I}$, and $\mathbf{M} = \mathbf{L}$. ■

Cholesky decomposition

Cholesky decomposition

Thm. 23. Let us suppose that $\mathbf{A}$ is a symmetric and positive definite matrix. Then there exist a unique lower triangular matrix $\mathbf{G}$ with positive diagonal such that $\mathbf{A} = \mathbf{G}\mathbf{G}^T$.

Proof: The matrix $\mathbf{A}$ can be written uniquely in the form $\mathbf{A} = \mathbf{L}\mathbf{D}\mathbf{L}^T$. The diagonal matrix $\mathbf{D}$ has positive diagonal. Let $\mathbf{G} = \mathbf{L} \cdot \mathrm{diag}(\sqrt{d_{11}}, \ldots, \sqrt{d_{nn}})$, which is a lower triangular matrix with positive diagonal. Moreover $\mathbf{G}\mathbf{G}^T = \mathbf{A}$. ■

Rmk. In practice, the Cholesky decomposition is not calculated with the above expression but the elements of $\mathbf{G}$ are calculated directly from above and from left by the help of the expression $\mathbf{A} = \mathbf{G}\mathbf{G}^T$. The number of operations is $n^3/3 + O(n^2)$ flop $+ n$ square root.

André-Louis Cholesky, 1875–1918, French

Iterative solutions of SLAEs

Linear iterative methods

When do we use iterative methods?

We would like to define a linear iteration

$$\overline{\mathbf{x}}_{k+1} = \mathbf{B}\overline{\mathbf{x}}_k + \overline{\mathbf{f}},\ k = 0, 1, \ldots$$

such that the limit of the vector sequence is the solution of the system $\mathbf{A}\overline{\mathbf{x}} = \overline{\mathbf{b}}$.

The number of operations in one iteration step is $2n^2\ flop$. Thus, we can perform $n/3$ iteration steps in order to not to exceed the number of operations of the Gaussian method. The method is mainly used for sparse matrices, when the number of nonzero elements is $O(n)$ (e.g. band matrices).

Questions:

- When does the sequence converge to the solution?
- How fast is the convergence?
- How to choose the matrix $\mathbf{B}$ and the vectors $\overline{\mathbf{f}}$, $\overline{\mathbf{x}}_0$?
- When to stop the iteration?

Convergence of iterative methods

Because of the inequality

$$\|\mathbf{B}\overline{\mathbf{x}}' - \overline{\mathbf{f}} - (\mathbf{B}\overline{\mathbf{x}}'' - \overline{\mathbf{f}})\| \leq \|\mathbf{B}\| \cdot \|\overline{\mathbf{x}}' - \overline{\mathbf{x}}'\|$$

and the Banach fixed point theorem (page 364), if $\|\mathbf{B}\| < 1$ in some induced norm ($\Leftrightarrow$ $\varrho(\mathbf{B}) < 1$), and the solution $\overline{\mathbf{x}}^\star$ of the system is a fixed point of the map $\overline{\mathbf{x}} \mapsto \mathbf{B}\overline{\mathbf{x}} + \overline{\mathbf{f}}$ then starting the iteration from an arbitrary vector, it will tend to the solution of the system. Moreover

$$\|\overline{\mathbf{x}}_k - \overline{\mathbf{x}}^\star\| \leq \frac{\|\mathbf{B}\|^k}{1 - \|\mathbf{B}\|}\|\overline{\mathbf{x}}_1 - \overline{\mathbf{x}}_0\|.$$

Rmk. The smaller the spectral radius the faster the convergence.

The construction of the iteration

The iteration can be constructed as follows. Let $\mathbf{A} = \mathbf{S} - \mathbf{T}$ and let $\mathbf{S}$ be nonsingular. Then

$$\mathbf{A}\overline{\mathbf{x}} = \overline{\mathbf{b}} \;\rightarrow\; (\mathbf{S} - \mathbf{T})\overline{\mathbf{x}} = \overline{\mathbf{b}} \;\rightarrow\; \overline{\mathbf{x}} = \mathbf{S}^{-1}\mathbf{T}\overline{\mathbf{x}} + \mathbf{S}^{-1}\overline{\mathbf{b}}.$$

$$\overline{\mathbf{x}}_{k+1} = \underbrace{\mathbf{S}^{-1}\mathbf{T}}_{\mathbf{B}}\overline{\mathbf{x}}_k + \underbrace{\mathbf{S}^{-1}\overline{\mathbf{b}}}_{\overline{\mathbf{f}}}.$$

The matrix $\mathbf{S}$ is called preconditioner. Because $\mathbf{B} = \mathbf{I} - \mathbf{S}^{-1}\mathbf{A}$, a good preconditioner must be

- close to $\mathbf{A}$, hence the norm of $\mathbf{B}$ can be small in this case (see later).
- and easily invertible.

Example.

- $\mathbf{S} = \mathbf{A}$: it is close to $\mathbf{A}$ but the computation of its inverse is as difficult as that of $\mathbf{A}$. The method converges in one step.
- $\mathbf{S} = \mathbf{I}$: inverse is easy, but it has nothing to do with $\mathbf{A}$.

Jacobi iteration

Jacobi iteration

Let $\mathbf{A} = \mathbf{D} - \mathbf{L} - \mathbf{R}$, where $\mathbf{D}$ is the diagonal matrix of $\mathbf{A}$ (suppose that there are no zeros in the diagonal). $\mathbf{L}$ is the matrix of the elements below the diagonal, while $\mathbf{R}$ is constructed from the elements above the diagonal, and both multiplied by -1. Let $\mathbf{S} = \mathbf{D}$ and $\mathbf{T} = \mathbf{R} + \mathbf{L}$.

Def. 24. The iteration

$$\overline{\mathbf{x}}_{k+1} = \underbrace{\mathbf{D}^{-1}(\mathbf{L} + \mathbf{R})}_{:=\mathbf{B}_J} \overline{\mathbf{x}}_k + \mathbf{D}^{-1}\overline{\mathbf{b}}$$

constructed with the above splitting ($\overline{\mathbf{x}}_0$ is arbitrary) is called Jacobi iteration.

Jacobi iteration

Carl Gustav Jacob Jacobi (1804-1851, German)

Componentwise:

$$(\overline{\mathbf{x}}_{k+1})_i = -\frac{1}{a_{ii}} \left(\sum_{j=1,\neq i}^{n} a_{ij} (\overline{\mathbf{x}}_k)_j - b_i \right), \quad i = 1, \ldots, n.$$

Gauss–Seidel iteration

Gauss–Seidel iteration

Let us modify the previous iteration! Let us use the newly computed components!

$$(\overline{\mathbf{x}}_{k+1})_i = -\frac{1}{a_{ii}} \left(\sum_{j=1}^{i-1} a_{ij} (\overline{\mathbf{x}}_{k+1})_j + \sum_{j=i+1}^{n} a_{ij} (\overline{\mathbf{x}}_k)_j - b_i \right).$$

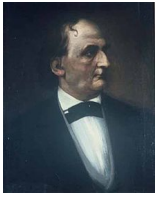
Matrix form:

$$\overline{\mathbf{x}}_{k+1} = \mathbf{D}^{-1}(\mathbf{L}\overline{\mathbf{x}}_{k+1} + \mathbf{R}\overline{\mathbf{x}}_k + \overline{\mathbf{b}}),$$

that is

$$\overline{\mathbf{x}}_{k+1} = \underbrace{(\mathbf{D} - \mathbf{L})^{-1}\mathbf{R}}_{\mathbf{B}_{GS}} \overline{\mathbf{x}}_k + (\mathbf{D} - \mathbf{L})^{-1}\overline{\mathbf{b}}.$$

Def. 25. The iteration constructed with the splitting $\mathbf{S} = \mathbf{D} - \mathbf{L}$, $\mathbf{T} = \mathbf{R}$ ($\overline{\mathbf{x}}_0$ is arbitrary) is called Gauss–Seidel iteration.



Philipp Ludwig von Seidel (1821-1896, German)

Comparison of the Jacobi and Gauss–Seidel iterations

The Gauss–Seidel seems to be better, because we always use the updated components, but if

$$\mathbf{A} = \begin{bmatrix} 1 & 1/2 & 1 \\ 1/2 & 1 & 1 \\ -2 & 2 & 1 \end{bmatrix}$$

then

$$\mathbf{B}_J = \begin{bmatrix} 0 & -1/2 & -1 \\ -1/2 & 0 & -1 \\ 2 & -2 & 0 \end{bmatrix}, \quad \mathbf{B}_{GS} \begin{bmatrix} 0 & -1/2 & -1 \\ 0 & 1/4 & -1/2 \\ 0 & -3/2 & -1 \end{bmatrix}.$$

Thus $\varrho(\mathbf{B}_J) = 1/2 < 1$ and $\varrho(\mathbf{B}_{GS}) = |-3/8 - \sqrt{73}/8| \approx 1.443 > 1$.

Relaxation methods

Relaxation methods

The Jacobi method fulfills the equality:

$$(\overline{\mathbf{x}}_{k+1})_i = (\overline{\mathbf{x}}_k)_i + (\overline{\mathbf{x}}_{k+1})_i - (\overline{\mathbf{x}}_k)_i.$$

The main idea of the relaxation for the Jacobi method:

$$(\tilde{\overline{\mathbf{x}}}_{k+1})_i = (\tilde{\overline{\mathbf{x}}}_k)_i + \omega((\tilde{\overline{\mathbf{x}}}_{k+1})_{i,J} - (\tilde{\overline{\mathbf{x}}}_k)_i),\ 0 \neq \omega \in \mathbb{R},$$

where $(\tilde{\overline{\mathbf{x}}}_0)_i = (\overline{\mathbf{x}}_0)_i$, $(\tilde{\overline{\mathbf{x}}}_{k+1})_{i,J}$ is the value where the Jacobi method would step from $(\tilde{\overline{\mathbf{x}}}^k)_i$ $(i = 1, \ldots, n)$, and ω is a so-called relaxation parameter.

Main goal: how to choose ω in order to make the convergence faster?

- $\omega = 1$: we get back the Jacobi iteration.
- $0 < \omega < 1$: under-relaxation.
- $\omega > 1$: over-relaxation.

JOR method (Jacobi over-relaxation, $J(\omega)$)

The componentwise form of the JOR method (without $\tilde{}$):

$$(\overline{\mathbf{x}}_{k+1})_i = (\overline{\mathbf{x}}_k)_i + \omega \left(-\frac{1}{a_{ii}} \left(\sum_{j=1,\neq i}^{n} a_{ij}(\overline{\mathbf{x}}_k)_j - b_i \right) - (\overline{\mathbf{x}}_k)_i \right)$$

$$= (1-\omega)(\overline{\mathbf{x}}_k)_i - \frac{\omega}{a_{ii}} \left[\sum_{j=1,j\neq i}^{n} a_{ij}(\overline{\mathbf{x}}_k)_j - b_i \right].$$

Thus we arrive at the vector form

$$\overline{\mathbf{x}}_{k+1} = \underbrace{((1-\omega)\mathbf{I} + \omega\mathbf{D}^{-1}(\mathbf{L}+\mathbf{R}))}_{\mathbf{B}_{J(\omega)}} \overline{\mathbf{x}}_k + \omega\mathbf{D}^{-1}\overline{\mathbf{b}},$$

where the iteration matrix is

$$\mathbf{B}_{J(\omega)} = \omega\mathbf{B}_J + (1-\omega)\mathbf{I}. \tag{2}$$

SOR method (Successive over-relaxation, $GS(\omega)$)

This method is the relaxation of the Gauss–Seidel method:

We apply the relaxation elementwise:

$$(\overline{\mathbf{x}}_{k+1})_i = (1-\omega)(\overline{\mathbf{x}}_k)_i - \frac{\omega}{a_{ii}}\left[\sum_{j=1}^{i-1} a_{ij}(\overline{\mathbf{x}}_{k+1})_j + \sum_{j=i+1}^{n} a_{ij}(\overline{\mathbf{x}}_k)_j - b_i\right].$$

In matrix form:

$$\overline{\mathbf{x}}_{k+1} = \underbrace{(\mathbf{D}-\omega\mathbf{L})^{-1}((1-\omega)\mathbf{D}+\omega\mathbf{R})}_{\mathbf{B}_{GS(\omega)}}\overline{\mathbf{x}}_k + \omega(\mathbf{D}-\omega\mathbf{L})^{-1}\overline{\mathbf{b}}.$$

Convergence

Convergence of regular splitting

Def. 26. The splitting $\mathbf{A} = \mathbf{S} - \mathbf{T}$ of the matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ is called regular splitting, if $\mathbf{S}$ is non-singular, $\mathbf{S}^{-1} \geq \mathbf{0}$ and $\mathbf{T} \geq \mathbf{0}$.

Thm. 27. If $\mathbf{A} = \mathbf{S} - \mathbf{T}$ is a regular splitting of a non-singular matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ with the property $\mathbf{A}^{-1} \geq \mathbf{0}$ then $\varrho(\mathbf{S}^{-1}\mathbf{T}) < 1$.

Thm. 28. Let $\mathbf{A} = \mathbf{D} - \mathbf{L} - \mathbf{R}$ (with the previous splitting), where we have $\mathbf{L} + \mathbf{R} \geq \mathbf{0}$. Then the matrix $\mathbf{A}$ has a regular splitting $\mathbf{A} = \mathbf{S} - \mathbf{T}$ with the property $\varrho(\mathbf{S}^{-1}\mathbf{T}) < 1$ iff $\mathbf{A}$ is an M-matrix.

Convergence of the Jacobi and Gauss–Seidel iterations

Thm. 29. For M-matrices, the J, J(ω), GS and GS(ω) ($\omega \in (0,1]$) methods are all convergent.

Proof. If $\mathbf{A}$ is an M-matrix then $\mathbf{A}^{-1} \geq \mathbf{0}$. In the case of the JOR method, the choice

$$\mathbf{S} = \frac{1}{\omega}\mathbf{D},\ \mathbf{T} = \frac{1-\omega}{\omega}\mathbf{D} + \mathbf{L} + \mathbf{R}$$

gives a regular splitting for $\omega \in (0,1]$. Thus the iteration is convergent based on the previous theorem.
In the case of the SOR method, the choice

$$\mathbf{S} = \frac{1}{\omega}\mathbf{D} - \mathbf{L},\ \mathbf{T} = \frac{1-\omega}{\omega}\mathbf{D} + \mathbf{R}$$

gives regular splitting for all $\omega \in (0,1]$. The case $\omega = 1$ gives back the Jacobi and Gauss–Seidel methods. ■

Convergence of the Jacobi and Gauss–Seidel iterations

Thm. 30. For matrices with strictly dominant diagonal, the Jacobi iteration is convergent. (Similar theorem is true for the Gauss–Seidel iteration.)

Proof.

$$\varrho(\mathbf{B}_J) \leq \|\mathbf{B}_J\|_\infty = \max_{i=1,\ldots,n} \sum_{j=1, j\neq i}^{n} \frac{|a_{ij}|}{|a_{ii}|} < 1. \blacksquare$$

Thm. 31. If $\mathbf{A}$ is symmetric and positive definite then the Gauss–Seidel iteration is convergent.

Thm. 32. [Ostrowski, Reich] If $\mathbf{A}$ is symmetric, positive definite and $\omega \in (0, 2)$ then

$$\varrho(\mathbf{B}_{GS(\omega)}) < 1,$$

that is the SOR method is convergent.

Thm. 33. [Kahan] For the SOR method we have

$$\varrho(\mathbf{B}_{GS(\omega)}) \geq |1 - \omega|,$$

that is the necessary condition of the convergence is $\omega \in (0, 2)$.

Stopping conditions

Stopping conditions

When to stop the iteration?

- If $\|\mathbf{B}\| < 1$ in some norm, then based on the Banach fixed point theorem we have

$$\|\overline{\mathbf{x}}_k - \overline{\mathbf{x}}^\star\| \leq \frac{\|\mathbf{B}\|^k}{1 - \|\mathbf{B}\|} \|\overline{\mathbf{x}}_1 - \overline{\mathbf{x}}_0\|.$$

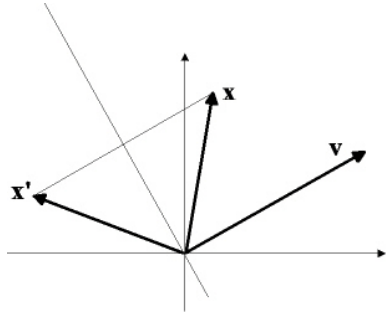
  From the value $\|\mathbf{B}\|$ and the result of the first iteration step, we can calculate that how many iteration we need to achieve a prescribed accuracy in a certain norm.
- Consider the results of two consecutive iterations. If $\|\overline{\mathbf{x}}_{k+1} - \overline{\mathbf{x}}_k\|$ is sufficiently small then we stop the iteration.
- We compute the so-called residuals: $\overline{\mathbf{r}}_k = \overline{\mathbf{b}} - \mathbf{A}\overline{\mathbf{x}}_k$. If $\|\overline{\mathbf{r}}_k\|/\|\overline{\mathbf{r}}_0\|$ is sufficiently small then we stop the iteration.
- We fix a value $k_{\max}$ where we stop the iteration at all events.

QR decomposition

Householder reflection

Householder reflection

How can we give the reflection image of a vector $\overline{\mathbf{x}}$ across a line through the origin that is perpendicular to the vector $\overline{\mathbf{v}}$ in $\mathbb{R}^2$?



$$\overline{\mathbf{x}}' = \overline{\mathbf{x}} - \frac{2\overline{\mathbf{v}}^T\overline{\mathbf{x}}}{\overline{\mathbf{v}}^T\overline{\mathbf{v}}}\overline{\mathbf{v}} = \overline{\mathbf{x}} - \frac{2\overline{\mathbf{v}}\overline{\mathbf{v}}^T\overline{\mathbf{x}}}{\overline{\mathbf{v}}^T\overline{\mathbf{v}}} = (\mathbf{I} - \frac{2\overline{\mathbf{v}}\overline{\mathbf{v}}^T}{\overline{\mathbf{v}}^T\overline{\mathbf{v}}})\overline{\mathbf{x}}.$$

Householder reflection

Let $\overline{\mathbf{v}} \in \mathbb{R}^n$ be an arbitrary nonzero vector. Then the multiplication with the matrix

$$\mathbf{H} = \mathbf{I} - \frac{2\overline{\mathbf{v}}\overline{\mathbf{v}}^T}{\overline{\mathbf{v}}^T\overline{\mathbf{v}}}$$

reflects each vector $\overline{\mathbf{x}}$ to the plain that goes through the origin and perpendicular to the vector $\overline{\mathbf{v}}$.

Thm. 34. $\mathbf{H}$ is a symmetric and orthogonal matrix.

Proof. The symmetry is trivial.

$$\left(\mathbf{I} - \frac{2\overline{\mathbf{v}}\overline{\mathbf{v}}^T}{\overline{\mathbf{v}}^T\overline{\mathbf{v}}}\right)\left(\mathbf{I} - \frac{2\overline{\mathbf{v}}\overline{\mathbf{v}}^T}{\overline{\mathbf{v}}^T\overline{\mathbf{v}}}\right) = \mathbf{I} - 4\frac{\overline{\mathbf{v}}\overline{\mathbf{v}}^T}{\overline{\mathbf{v}}^T\overline{\mathbf{v}}} + 4\frac{\overline{\mathbf{v}}\overline{\mathbf{v}}^T}{\overline{\mathbf{v}}^T\overline{\mathbf{v}}}\frac{\overline{\mathbf{v}}\overline{\mathbf{v}}^T}{\overline{\mathbf{v}}^T\overline{\mathbf{v}}} = \mathbf{I}. \blacksquare$$

Householder reflection

Question: How to choose the vector $\overline{\mathbf{v}}$ to reflect the vector $\overline{\mathbf{x}}$ to the axis x_1, that is parallel to the vector $\overline{\mathbf{e}}_1$?

$$\underbrace{\mathbf{H}\overline{\mathbf{x}}}_{\in lin(\overline{\mathbf{e}}_1)} = \overline{\mathbf{x}} - \frac{2\overline{\mathbf{v}}^T\overline{\mathbf{x}}}{\overline{\mathbf{v}}^T\overline{\mathbf{v}}}\overline{\mathbf{v}},$$

thus $\overline{\mathbf{v}} \in lin(\overline{\mathbf{x}}, \overline{\mathbf{e}}_1)$. Let $\overline{\mathbf{v}} = \overline{\mathbf{x}} + \alpha\overline{\mathbf{e}}_1$.
Then

$$\mathbf{H}\overline{\mathbf{x}} = \overline{\mathbf{x}} - \frac{2(\overline{\mathbf{x}}^T + \alpha\overline{\mathbf{e}}_1^T)\overline{\mathbf{x}}}{(\overline{\mathbf{x}} + \alpha\overline{\mathbf{e}}_1)^T(\overline{\mathbf{x}} + \alpha\overline{\mathbf{e}}_1)}(\overline{\mathbf{x}} + \alpha\overline{\mathbf{e}}_1)$$

$$= \overline{\mathbf{x}} - 2\frac{\overline{\mathbf{x}}^T\overline{\mathbf{x}} + \alpha x_1}{\overline{\mathbf{x}}^T\overline{\mathbf{x}} + 2\alpha x_1 + \alpha^2}\overline{\mathbf{x}} - \alpha\frac{2\overline{\mathbf{v}}^T\overline{\mathbf{x}}}{\overline{\mathbf{v}}^T\overline{\mathbf{v}}}\overline{\mathbf{e}}_1$$

$$= \left(1 - 2\frac{\|\overline{\mathbf{x}}\|_2^2 + \alpha x_1}{\|\overline{\mathbf{x}}\|_2^2 + 2\alpha x_1 + \alpha^2}\right)\overline{\mathbf{x}} - \alpha\frac{2\overline{\mathbf{v}}^T\overline{\mathbf{x}}}{\overline{\mathbf{v}}^T\overline{\mathbf{v}}}\overline{\mathbf{e}}_1.$$

If $\alpha = \pm\|\overline{\mathbf{x}}\|_2$ then the coefficient of $\overline{\mathbf{x}}$ is zero.

Householder reflection

Thus, if a vector $\overline{\mathbf{x}} \neq \mathbf{0}$ is given then $\overline{\mathbf{v}} = \overline{\mathbf{x}} \pm \|\overline{\mathbf{x}}\|_2 \overline{\mathbf{e}}_1$ is a good choice. Then

$$\mathbf{H}\overline{\mathbf{x}} = \mp\|\overline{\mathbf{x}}\|_2 \frac{2(\overline{\mathbf{x}} \pm \|\overline{\mathbf{x}}\|_2\overline{\mathbf{e}}_1)^T\overline{\mathbf{x}}}{(\overline{\mathbf{x}} \pm \|\overline{\mathbf{x}}\|_2\overline{\mathbf{e}}_1)^T(\overline{\mathbf{x}} \pm \|\overline{\mathbf{x}}\|_2\overline{\mathbf{e}}_1)}\overline{\mathbf{e}}_1$$

$$= \mp\|\overline{\mathbf{x}}\|_2 \frac{2\|\overline{\mathbf{x}}\|_2^2 \pm 2\|\overline{\mathbf{x}}\|_2 x_1}{2\|\overline{\mathbf{x}}\|_2^2 \pm 2\|\overline{\mathbf{x}}\|_2 x_1}\overline{\mathbf{e}}_1 = \mp\|\overline{\mathbf{x}}\|_2\overline{\mathbf{e}}_1.$$

Def. 35. The reflection matrix $\mathbf{H}$ that reflects a given vector $\overline{\mathbf{x}}$ through a plane that goes through the origin such a way that the reflection is on the first coordinate axis, is called Householder reflection (that belongs to the vector $\overline{\mathbf{x}}$).

Application: Based on the above considerations, the Householder reflection that belongs to the vector $\overline{\mathbf{x}}$ can be determined as follows:
- We determine the normal vector of the plane of reflection: $\overline{\mathbf{v}} = \overline{\mathbf{x}} \pm \|\overline{\mathbf{x}}\|_2\overline{\mathbf{e}}_1$,
- then we construct the reflection matrix with the vector $\overline{\mathbf{v}}$:

$$\mathbf{H} = \mathbf{I} - \frac{2\overline{\mathbf{v}}\overline{\mathbf{v}}^T}{\overline{\mathbf{v}}^T\overline{\mathbf{v}}}.$$

Householder reflection

$$\mathbf{H}\overline{\mathbf{x}} = \mathbf{H}\begin{bmatrix} * \\ * \\ \vdots \\ * \end{bmatrix} = \begin{bmatrix} * \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

Rmk. If $x_1 \neq 0$ then it is practical to choose the normal vector as $\overline{\mathbf{v}} = \overline{\mathbf{x}} + \operatorname{sgn}(x_1)\|\overline{\mathbf{x}}\|_2\overline{\mathbf{e}}_1$.

Rmk. It is practical to norm the vector $\overline{\mathbf{v}}$ such that the first element of the vector will be 1. Then $\overline{\mathbf{v}}$ can be stored in the place of the eliminated elements of $\overline{\mathbf{x}}$.

Rmk. Let $\mathbf{C}$ be an arbitrary matrix. Then the calculation of $\mathbf{HC}$ can be performed as follows:

$$\mathbf{HC} = \left(\mathbf{I} - \frac{2\overline{\mathbf{v}}\overline{\mathbf{v}}^T}{\overline{\mathbf{v}}^T\overline{\mathbf{v}}}\right)\mathbf{C} = \mathbf{C} - \frac{2\overline{\mathbf{v}}\overline{\mathbf{v}}^T}{\overline{\mathbf{v}}^T\overline{\mathbf{v}}}\mathbf{C}$$

$$= \mathbf{C} + \overline{\mathbf{v}}\underbrace{\left(-\frac{2\overline{\mathbf{v}}^T\mathbf{C}}{\overline{\mathbf{v}}^T\overline{\mathbf{v}}}\right)}_{=:\overline{\mathbf{w}}^T} = \mathbf{C} + \overline{\mathbf{v}}\overline{\mathbf{w}}^T.$$

QR decomposition

QR decomposition

Thm. 36. Let $\mathbf{A} \in \mathbb{R}^{m \times n}$ $(m \geq n)$ be a full rank matrix. Then there exists an orthogonal matrix $\mathbf{Q} \in \mathbb{R}^{m \times m}$ and an upper triangular matrix $\mathbf{R} \in \mathbb{R}^{m \times n}$ such that $\mathbf{A} = \mathbf{QR}$.

Proof. Let $\mathbf{H}_1$ be the Householder reflection that belongs to the column $\mathbf{A}(1:m, 1)$. Then the $2:m$ elements of the first column of $\mathbf{A}^{(2)} := \mathbf{H}_1\mathbf{A}$ are zero. Let $\tilde{\mathbf{H}}_2$ be the Householder reflection that belongs to the column $\mathbf{A}^{(2)}(2:m, 2)$. Moreover, let $\mathbf{H}_2 = \mathrm{diag}(1, \tilde{\mathbf{H}}_2)$. Then the $2:m$ elements of the first column of $\mathbf{A}^{(3)} := \mathbf{H}_2\mathbf{A}^{(2)}$ and the $3:m$ elements of the second column are zero, etc. Based on the full rank, this procedure can be continued further. We obtain the representation

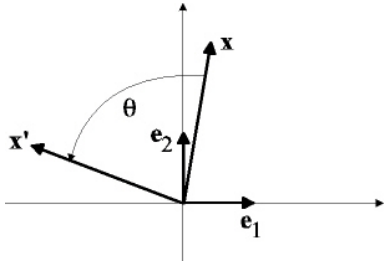
$$\mathbf{H}_n \cdot \cdots \cdot \mathbf{H}_1 \cdot \mathbf{A} = \mathbf{R},$$

where $\mathbf{R}$ is an upper triangular matrix. The matrix $\mathbf{Q}^T := \mathbf{H}_n \cdot \cdots \cdot \mathbf{H}_1$ is orthogonal, so with the above notations we have $\mathbf{A} = \mathbf{QR}$. ■

Givens rotation

Givens rotation

Rotation with angle θ in $\mathbb{R}^2$.



$$\overline{\mathbf{x}}' = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \overline{\mathbf{x}}.$$

This matrix is orthogonal. Moreover with the choice $s = \sin\theta$ and $c = \cos\theta$, the vector $[x_1, x_2]^T$ $(x_1 \neq 0)$ is transformed to the form $[*, 0]^T$.

Givens rotation

- If $x_2 = 0$ then $s = 0$, $c = 1$ is a good choice.
- If $x_2 \neq 0$ then from the solution of the SLAE $sx_1 + cx_2 = 0$, $s^2 + c^2 = 1$ we obtain the parameters

$$s = \frac{\pm x_2}{\sqrt{x_1^2 + x_2^2}}, \quad c = \frac{\mp x_1}{\sqrt{x_1^2 + x_2^2}}.$$

Generally: rotation in the hyperplane (i, j) with angle θ

$$\mathbf{G}(i, j, \theta) = \begin{bmatrix} 1 & & & & & & & \\ & \ddots & & & & & & \\ & & c & & & & -s & \\ & & & 1 & & & & \\ & & & & \ddots & & & \\ & & & & & 1 & & \\ & & s & & & & c & \\ & & & & & & & \ddots \\ & & & & & & & & 1 \end{bmatrix}$$

Application of the Givens rotation

QR decomposition (schematically):

$$\begin{bmatrix} * & * & * \\ * & * & * \\ * & * & * \\ * & * & * \end{bmatrix} \to \begin{bmatrix} * & * & * \\ * & * & * \\ * & * & * \\ 0 & * & * \end{bmatrix} \to \begin{bmatrix} * & * & * \\ * & * & * \\ 0 & * & * \\ 0 & * & * \end{bmatrix} \to \begin{bmatrix} * & * & * \\ 0 & * & * \\ 0 & * & * \\ 0 & * & * \end{bmatrix}$$

$$\begin{bmatrix} * & * & * \\ 0 & * & * \\ 0 & * & * \\ 0 & 0 & * \end{bmatrix} \to \begin{bmatrix} * & * & * \\ 0 & * & * \\ 0 & 0 & * \\ 0 & 0 & * \end{bmatrix} \to \begin{bmatrix} * & * & * \\ 0 & * & * \\ 0 & 0 & * \\ 0 & 0 & 0 \end{bmatrix}$$

Rmk. The number of operations of the Householder QR decomposition is $2n^2(m - n/3)$, while for the Givens QR decomposition we have $3n^2(m - n/3)$.

Application of Givens rotation

The QR decomposition of an upper Hessenberg matrix (schematically):

$$\begin{bmatrix} * & * & * \\ * & * & * \\ 0 & * & * \\ 0 & 0 & * \end{bmatrix} \rightarrow \begin{bmatrix} * & * & * \\ 0 & * & * \\ 0 & * & * \\ 0 & 0 & * \end{bmatrix} \rightarrow \begin{bmatrix} * & * & * \\ 0 & * & * \\ 0 & 0 & * \\ 0 & 0 & * \end{bmatrix} \rightarrow \begin{bmatrix} * & * & * \\ 0 & * & * \\ 0 & 0 & * \\ 0 & 0 & 0 \end{bmatrix}$$

Householder and Givens


,


Alston Scott Householder, 1904-1993 (USA), Wallace Givens, 1910-1993 (USA)

Solution of full rank over-determined systems

Solution of over-determined systems

Over-determined systems

$$\mathbf{A}\overline{\mathbf{x}} = \overline{\mathbf{b}}, \quad \mathbf{A} \in \mathbb{R}^{m\times n},\ m \geq n,\ r(\mathbf{A}) = n$$

The above system generally does not have solution (or only one). Then we can search for the vector $\overline{\mathbf{x}}$ (denoted by $\overline{\mathbf{x}}_{LS}$) that minimizes the norm $\|\mathbf{A}\overline{\mathbf{x}} - \overline{\mathbf{b}}\|_2^2$.
Let

$$\phi(\overline{\mathbf{x}}) = \|\mathbf{A}\overline{\mathbf{x}} - \overline{\mathbf{b}}\|_2^2,$$

and let $\overline{\mathbf{z}} \in \mathbb{R}^n$ be an arbitrary vector. Because of the full column rank, $\|\mathbf{A}\overline{\mathbf{z}}\|_2 = 0$ can hold only if $\overline{\mathbf{z}} = \mathbf{0}$. Then

$$\phi(\overline{\mathbf{x}} + \overline{\mathbf{z}}) = \|\mathbf{A}(\overline{\mathbf{x}} + \overline{\mathbf{z}}) - \overline{\mathbf{b}}\|_2^2$$

$$= \|\mathbf{A}\overline{\mathbf{x}} - \overline{\mathbf{b}}\|_2^2 + \|\mathbf{A}\overline{\mathbf{z}}\|_2^2 + 2\overline{\mathbf{z}}^T\mathbf{A}^T(\mathbf{A}\overline{\mathbf{x}} - \overline{\mathbf{b}}).$$

Let $\overline{\mathbf{x}}_{LS}$ be the solution of the SLAE $\mathbf{A}^T\mathbf{A}\overline{\mathbf{x}} = \mathbf{A}^T\overline{\mathbf{b}}$ ($\overline{\mathbf{z}}^T\mathbf{A}^T\mathbf{A}\overline{\mathbf{z}} = \|\mathbf{A}\overline{\mathbf{z}}\|_2^2 \neq 0$ provided that $\overline{\mathbf{z}} \neq \mathbf{0}$, thus $\mathbf{A}^T\mathbf{A}$ is SPD, thus it is non-singular). Then

$$\phi(\overline{\mathbf{x}}_{LS} + \overline{\mathbf{z}}) = \|\mathbf{A}\overline{\mathbf{x}}_{LS} - \overline{\mathbf{b}}\|_2^2 + \|\mathbf{A}\overline{\mathbf{z}}\|_2^2 = \phi(\overline{\mathbf{x}}_{LS}) + \|\mathbf{A}\overline{\mathbf{z}}\|_2^2,$$

that shows that $\overline{\mathbf{x}}_{LS}$ uniquely minimizes ϕ indeed.

Over-determined systems

We have to solve the so-called normal equation

$$\mathbf{A}^T\mathbf{A}\overline{\mathbf{x}} = \mathbf{A}^T\overline{\mathbf{b}}.$$

It has unique solution due to the full rank, thus the solution can be written in the form $\overline{\mathbf{x}}_{LS} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\overline{\mathbf{b}}$. This is not efficient in practice.

Computation of $\overline{\mathbf{x}}_{LS}$ with the normal equation

- $\mathbf{A}^T\mathbf{A}$ is SPD.
- Let us compute its Cholesky decomposition $\mathbf{L}\mathbf{L}^T$.
- Let us solve the system $\mathbf{L}\overline{\mathbf{y}} = \mathbf{A}^T\overline{\mathbf{b}}$.
- We get $\overline{\mathbf{x}}_{LS}$ as the solution of $\mathbf{L}^T\overline{\mathbf{x}} = \overline{\mathbf{y}}$.

Number of operations: $(m + n/3)n^2$ flop

Computation of $\overline{\mathbf{x}}_{LS}$ with QR decomposition

$$\|\mathbf{A}\overline{\mathbf{x}} - \overline{\mathbf{b}}\|_2^2 = \|\mathbf{Q}\mathbf{R}\overline{\mathbf{x}} - \overline{\mathbf{b}}\|_2^2 = \|\mathbf{Q}^T(\mathbf{Q}\mathbf{R}\overline{\mathbf{x}} - \overline{\mathbf{b}})\|_2^2$$
$$= \|\mathbf{R}\overline{\mathbf{x}} - \mathbf{Q}^T\overline{\mathbf{b}}\|_2^2 = \|\mathbf{R}_1\overline{\mathbf{x}} - \overline{\mathbf{c}}\|_2^2 + \|\overline{\mathbf{d}}\|_2^2,$$

where $\mathbf{R}_1 = \mathbf{R}(1:n, 1:n)$, $\overline{\mathbf{c}} = (\mathbf{Q}^T\overline{\mathbf{b}})(1:n, :)$, $\overline{\mathbf{d}} = (\mathbf{Q}^T\overline{\mathbf{b}})(n+1:m, :)$.

- Compute the QR decomposition of $\mathbf{A}$.
- Determine the matrix $\mathbf{R}_1 = \mathbf{R}(1:n, 1:n)$.
- Determine the vector $\overline{\mathbf{c}} = (\mathbf{Q}^T\overline{\mathbf{b}})(1:n, :)$.
- $\overline{\mathbf{x}}_{LS}$ is the solution of the SLAE $\mathbf{R}_1\overline{\mathbf{x}} = \overline{\mathbf{c}}$.

Number of operations: $2(m - n/3)n^2$ flop

Over-determined systems

Rmk.

- If $m >> n$ then the number of operations of the solution with the QR decomposition is approximately the double of that of the other.
- For quadratic full rank matrices, the number of operations is the same in both cases: $4n^3/3$, which is the double of that of the Gaussian method. When we take into the account also the memory usage, then the total solution time may be comparable with that of the Gaussian method, moreover, in this case there is no growth factor, that is the method is stable.
- We cannot use these methods for (nearly) rank deficient matrices.
- For the normal equation, we can use the CG method but the condition number of the new system will be the square of that of the original system.

Eigenvalue problems

The power method

The idea of the power method

Let $\mathbf{A} \in \mathbb{R}^{n \times n}$ be a normal matrix, and let us suppose that $\mathbf{A}$ has a strictly dominant eigenvalue, that is

$$|\lambda_1| > |\lambda_2| \geq \dots |\lambda_n|.$$

Then the eigenvalue $\lambda_1 \in \mathbb{R}$ and the corresponding eigenvector $\overline{\mathbf{v}}_1$ can be chosen to be real. Let $\overline{\mathbf{v}}_1, \dots, \overline{\mathbf{v}}_n$ be the normed eigenvectors, and because $\mathbf{A}$ is normal, they form an orthonormal basis. Let $\overline{\mathbf{x}} \in \mathbb{R}^n$ be such that $\alpha_1 \neq 0$ ($\alpha_1 \in \mathbb{R}$) is not zero in the form $\overline{\mathbf{x}} = \alpha_1 \overline{\mathbf{v}}_1 + \alpha_2 \overline{\mathbf{v}}_2 + \dots + \alpha_n \overline{\mathbf{v}}_n$.

Then

$$\mathbf{A}^k \overline{\mathbf{x}} = \alpha_1 \lambda_1^k \overline{\mathbf{v}}_1 + \alpha_2 \lambda_2^k \overline{\mathbf{v}}_2 + \dots + \alpha_n \lambda_n^k \overline{\mathbf{v}}_n$$

$$= \lambda_1^k \left(\alpha_1 \overline{\mathbf{v}}_1 + \underbrace{\alpha_2 \left(\frac{\lambda_2}{\lambda_1} \right)^k \overline{\mathbf{v}}_2}_{\to 0} + \dots + \underbrace{\alpha_n \left(\frac{\lambda_n}{\lambda_1} \right)^k \overline{\mathbf{v}}_n}_{\to 0} \right).$$

The power method

The power method, $\overline{\mathbf{v}}_1^T \overline{\mathbf{y}}^{(0)} \neq 0$, $\|\overline{\mathbf{y}}^{(0)}\|_2 = 1$

for $k := 1 : k_{\max}$ **do**
$\overline{\mathbf{x}}^{(k)} := \mathbf{A}\overline{\mathbf{y}}^{(k-1)}$
$\overline{\mathbf{y}}^{(k)} := \overline{\mathbf{x}}^{(k)} / \|\overline{\mathbf{x}}^{(k)}\|_2$
$\nu^{(k)} := (\overline{\mathbf{y}}^{(k)})^T \mathbf{A}\overline{\mathbf{y}}^{(k)}$
end for

Thm. 37.

$$\overline{\mathbf{y}}^{(k)} = \frac{\mathbf{A}^k \overline{\mathbf{y}}^{(0)}}{\|\mathbf{A}^k \overline{\mathbf{y}}^{(0)}\|_2},$$

$\nu^{(k)} \to \lambda_1$, moreover there exists a sequence $\{\gamma_k\} \subset \mathbb{R}$ such that $|\gamma_k| = 1$ $(k = 1, \dots)$ and

$$\gamma_k \overline{\mathbf{y}}^{(k)} \to \overline{\mathbf{v}}_1.$$

Proof.

The first part can be proven with induction.

Parseval's equality: $\|\overline{\mathbf{x}}\|_2 = \sqrt{\sum_{i=1}^{n} |\alpha_i|^2}$.

Namely:

$$\overline{\mathbf{x}}^H \overline{\mathbf{x}} = \left(\sum_{i=1}^{n} \overline{\alpha}_i \overline{\mathbf{v}}_i^H \right) \left(\sum_{i=1}^{n} \alpha_i \overline{\mathbf{v}}_i \right) = \sum_{i=1}^{n} |\alpha_i|^2.$$

Let $\overline{\mathbf{y}}^{(0)} = \alpha_1 \overline{\mathbf{v}}_1 + \alpha_2 \overline{\mathbf{v}}_2 + \cdots + \alpha_n \overline{\mathbf{v}}_n$ and we know that $\alpha_1 \neq 0$. Hence

$$\overline{\mathbf{y}}^{(k)} = \frac{\lambda_1^k \left(\alpha_1 \overline{\mathbf{v}}_1 + \alpha_2 \left(\frac{\lambda_2}{\lambda_1} \right)^k \overline{\mathbf{v}}_2 + \cdots + \alpha_n \left(\frac{\lambda_n}{\lambda_1} \right)^k \overline{\mathbf{v}}_n \right)}{\sqrt{\sum_{i=1}^{n} |\alpha_i|^2 |\lambda_i|^{2k}}}$$

$$= \frac{\lambda_1^k \alpha_1 \left(\overline{\mathbf{v}}_1 + \frac{\alpha_2}{\alpha_1} \left(\frac{\lambda_2}{\lambda_1} \right)^k \overline{\mathbf{v}}_2 + \cdots + \frac{\alpha_n}{\alpha_1} \left(\frac{\lambda_n}{\lambda_1} \right)^k \overline{\mathbf{v}}_n \right)}{|\lambda_1|^k |\alpha_1| \sqrt{1 + \sum_{i=2}^{n} |\frac{\alpha_i}{\alpha_1}|^2 |\frac{\lambda_i}{\lambda_1}|^{2k}}}.$$

The power method

Thus

$$\overbrace{\frac{|\lambda_1|^k|\alpha_1|}{\lambda_1^k\alpha_1}}^{=:\gamma_k}\overline{\mathbf{y}}^{(k)}$$

$$= \frac{\left(\overline{\mathbf{v}}_1 + \frac{\alpha_2}{\alpha_1}\left(\frac{\lambda_2}{\lambda_1}\right)^k\overline{\mathbf{v}}_2 + \cdots + \frac{\alpha_n}{\alpha_1}\left(\frac{\lambda_n}{\lambda_1}\right)^k\overline{\mathbf{v}}_n\right)}{\sqrt{1+\sum_{i=2}^n|\frac{\alpha_i}{\alpha_1}|^2|\frac{\lambda_i}{\lambda_1}|^{2k}}} \to \overline{\mathbf{v}}_1,$$

where $|\gamma_k| = 1$ $(k = 1, \dots)$.

$$0 \leftarrow (\gamma_k\overline{\mathbf{y}}^{(k)})^T\mathbf{A}(\gamma_k\overline{\mathbf{y}}^{(k)}) - \overline{\mathbf{v}}_1^T\mathbf{A}\overline{\mathbf{v}}_1 = |\gamma_k|^2(\overline{\mathbf{y}}^{(k)})^T\mathbf{A}\overline{\mathbf{y}}^{(k)} - \lambda_1$$
$$= (\overline{\mathbf{y}}^{(k)})^T\mathbf{A}\overline{\mathbf{y}}^{(k)} - \lambda_1 = \nu^{(k)} - \lambda_1. \blacksquare$$

The power method

Rmk.

- If $\lambda_1, \alpha_1 > 0$, then $\overline{\mathbf{y}}^{(k)} \to \overline{\mathbf{v}}_1$.
- If $\lambda_1 > 0, \alpha_1 < 0$, then $-\overline{\mathbf{y}}^{(k)} \to \overline{\mathbf{v}}_1$.
- If $\lambda_1 < 0, \alpha_1 > 0$, then $(-1)^k \overline{\mathbf{y}}^{(k)} \to \overline{\mathbf{v}}_1$.
- If $\lambda_1 < 0, \alpha_1 < 0$, then $(-1)^{k+1} \overline{\mathbf{y}}^{(k)} \to \overline{\mathbf{v}}_1$.

Rmk. Let $\overline{\mathbf{e}}^{(k)} = \overline{\mathbf{y}}^{(k)} - \overline{\mathbf{v}}_1$ be the error of the kth iteration vector. Then, for sufficiently large values k we have $\|\overline{\mathbf{e}}^{(k+1)}\|_2 \approx |\lambda_2/\lambda_1| \|\overline{\mathbf{e}}^{(k)}\|_2$ (linear convergence).

Rmk. If $\overline{\mathbf{x}}$ is an approximation of the eigenvector that belongs to the dominant eigenvalue of $\mathbf{A}$, then we have $\overline{\mathbf{x}}^T(\mathbf{A}\overline{\mathbf{x}}) \approx \overline{\mathbf{x}}^T(\lambda\overline{\mathbf{x}})$ and

$$\lambda \approx \frac{\overline{\mathbf{x}}^T \mathbf{A} \overline{\mathbf{x}}}{\overline{\mathbf{x}}^T \overline{\mathbf{x}}}$$

is an approximation of the eigenvalue.

Rayleigh's coefficient

Rayleigh's coefficient

Def. 38. Let $\mathbf{0} \neq \overline{\mathbf{x}} \in \mathbb{R}^n$, $\mathbf{A} \in \mathbb{R}^{n \times n}$. The number

$$R(\overline{\mathbf{x}}) = \frac{\overline{\mathbf{x}}^T \mathbf{A} \overline{\mathbf{x}}}{\overline{\mathbf{x}}^T \overline{\mathbf{x}}}$$

is called the Rayleigh's coefficient to the vector $\overline{\mathbf{x}}$.

Thm. 39. Let the $\mathbf{0} \neq \overline{\mathbf{x}} \in \mathbb{R}^n$ be a given vector. Then

$$\min_{\alpha \in \mathbb{R}} \|\mathbf{A}\overline{\mathbf{x}} - \alpha \overline{\mathbf{x}}\|_2 = \|\mathbf{A}\overline{\mathbf{x}} - R(\overline{\mathbf{x}})\overline{\mathbf{x}}\|_2.$$

Proof.

$$\begin{aligned} \|\mathbf{A}\overline{\mathbf{x}} - \alpha \overline{\mathbf{x}}\|_2^2 &= (\overline{\mathbf{x}}^T \mathbf{A}^T - \alpha \overline{\mathbf{x}}^T)(\mathbf{A}\overline{\mathbf{x}} - \alpha \overline{\mathbf{x}}) \\ &= \overline{\mathbf{x}}^T \mathbf{A}^T \mathbf{A} \overline{\mathbf{x}} - 2\alpha \overline{\mathbf{x}}^T \mathbf{A} \overline{\mathbf{x}} + \alpha^2 \overline{\mathbf{x}}^T \overline{\mathbf{x}} \\ &= \alpha^2 \overline{\mathbf{x}}^T \overline{\mathbf{x}} - 2\alpha \overline{\mathbf{x}}^T \mathbf{A} \overline{\mathbf{x}} + \overline{\mathbf{x}}^T \mathbf{A}^T \mathbf{A} \overline{\mathbf{x}}. \end{aligned}$$

Rayleigh's coefficient

Because $\overline{\mathbf{x}}^T\overline{\mathbf{x}} > 0$ if $\overline{\mathbf{x}} \neq \mathbf{0}$, hence the function takes its minimum at

$$\alpha_{\min} = \frac{\overline{\mathbf{x}}^T \mathbf{A}\overline{\mathbf{x}}}{\overline{\mathbf{x}}^T\overline{\mathbf{x}}} = R(\overline{\mathbf{x}}). \blacksquare$$

Rmk. For symmetric matrices

$$\lambda_{\min} \leq R(\overline{\mathbf{x}}) \leq \lambda_{\max}.$$

Rmk. For symmetric matrices

$$\lambda_{\max} = \max_{\overline{\mathbf{x}} \in \mathbb{R}^n \neq 0} R(\overline{\mathbf{x}}), \quad \lambda_{\min} = \min_{\overline{\mathbf{x}} \in \mathbb{R}^n \neq 0} R(\overline{\mathbf{x}})$$

(Courant-Fischer theorem).

From now on, we will consider only symmetric matrices in the eigenvalue problems!

Inverse iteration

Inverse iteration

Let $\mathbf{A} \in \mathbb{R}^{n \times n}$ be a non-singular symmetric matrix with the eigenvalues λ_i and with the eigenvectors $\overline{\mathbf{v}}_i$. Then, if $\mu \neq \lambda_i$, then the matrix $\mathbf{A} - \mu\mathbf{I}$ is invertible and the eigenvectors of $(\mathbf{A} - \mu\mathbf{I})^{-1}$ are identical with those of $\mathbf{A}$, its eigenvalues are $(\lambda_i - \mu)^{-1}$.

If the number μ is sufficiently close to λ_j, then the dominant eigenvalue will be $(\lambda_j - \mu)^{-1}$, thus executing the power method with the matrix $(\mathbf{A} - \mu\mathbf{I})^{-1}$, λ_j and $\overline{\mathbf{v}}_j$ can be approximated.

Inverse iteration

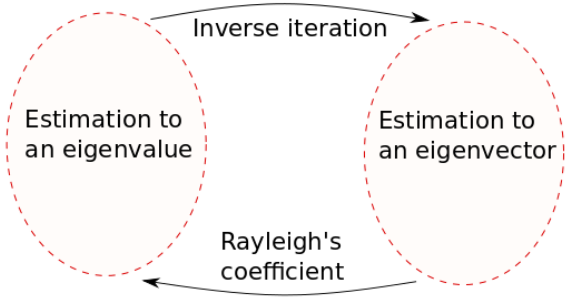
Inverse iteration, $\overline{\mathbf{v}}_1^T \overline{\mathbf{y}}^{(0)} \neq 0$, $\|\overline{\mathbf{y}}^{(0)}\|_2 = 1$

for $k := 1 : k_{\max}$ **do**
$\overline{\mathbf{x}}^{(k)} := (\mathbf{A} - \mu\mathbf{I})^{-1} \overline{\mathbf{y}}^{(k-1)}$
(solution of $(\mathbf{A} - \mu\mathbf{I})\overline{\mathbf{x}}^{(k)} = \overline{\mathbf{y}}^{(k-1)}$)
$\overline{\mathbf{y}}^{(k)} := \overline{\mathbf{x}}^{(k)} / \|\overline{\mathbf{x}}^{(k)}\|_2$
$\nu^{(k)} := (\overline{\mathbf{y}}^{(k)})^T \mathbf{A} \overline{\mathbf{y}}^{(k)}$
end for

Rmk.

- First we compute the LU-decomposition of the matrix $\mathbf{A} - \mu\mathbf{I}$. This makes possible to solve the system with $2n^2$ flops in each iteration.
- Much more expensive than the power method, but it can converge to any eigenvalue.
- The condition $\overline{\mathbf{v}}_1^T \overline{\mathbf{y}}^{(0)} \neq 0$ is not too strict. If it does not hold initially, then it will be satisfied after sufficiently large number of iterations due to the rounding errors. Thus, the method will converge in this case, too.

Approximation of eigenvalues and eigenvectors



Rank deflation

Rank deflation

- Let us suppose that we have computed already the strictly dominant eigenvalue λ_1 and the corresponding eigenvector $\overline{\mathbf{v}}_1$ of the matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$.
- Let us consider the matrix $\mathbf{A} - \lambda_1 \overline{\mathbf{v}}_1 \overline{\mathbf{v}}_1^T$. The eigenvalues of this matrix equal the eigenvalues of $\mathbf{A}$, with the only difference that zero stands instead of λ_1. The eigenvectors are the same.
- When λ_2 is strictly dominant, then executing the power method with the above matrix, we can obtain λ_2 and $\overline{\mathbf{v}}_2$.

QR-iteration

QR-iteration

Main idea: If we could find a matrix $\mathbf{V}$ to the matrix $\mathbf{A}$ such that $\mathbf{V}^{-1}\mathbf{A}\mathbf{V}$ is an upper triangular matrix, then the diagonal of this upper triangular matrix would contain the eigenvalues of the matrix. Unfortunately such a matrix $\mathbf{V}$ cannot be constructed directly.

Let us approximate this matrix with the orthogonal matrices of the QR decomposition.

QR iteration, $\mathbf{A}$ is a given symmetric matrix, $\mathbf{A}^{(0)} := \mathbf{A}$

for $k := 1 : k_{\max}$ **do**
  Construct the QR decomposition of $\mathbf{A}^{(k-1)}$: $\mathbf{A}^{(k-1)} = \mathbf{Q}^{(k-1)}\mathbf{R}^{(k-1)}$
  $\mathbf{A}^{(k)} := (\mathbf{Q}^{(k-1)})^T\mathbf{A}^{(k-1)}\mathbf{Q}^{(k-1)} = \mathbf{R}^{(k-1)}\mathbf{Q}^{(k-1)}$
end for

QR iteration

Thus

$$\mathbf{A}^{(k)} = (\mathbf{Q}^{(k-1)})^T \dots (\mathbf{Q}^{(0)})^T \mathbf{A} \underbrace{\mathbf{Q}^{(0)} \dots \mathbf{Q}^{(k-1)}}_{=:\mathbf{Q}_k} = \mathbf{Q}_k^T \mathbf{A} \mathbf{Q}_k,$$

and the eigenvalues of $\mathbf{A}^{(k)}$ will be the same as the eigenvalues of $\mathbf{A}$.

Thm. 40. a) If all the eigenvalues of $\mathbf{A}$ are real and different in absolute values, then the matrix sequence $\{\mathbf{A}^{(k)}\}$ tends to an upper triangular matrix.
b) If all the eigenvalues of a symmetric matrix $\mathbf{A}$ are different in absolute values, then the matrix sequence $\{\mathbf{A}^{(k)}\}$ tends to a diagonal matrix.

Rmk. In both cases the eigenvalues appear in the diagonal of the limit matrix.

Remarks

Rmk. Let $\mathbf{A} = \mathbf{QR}$ be an upper Hessenberg matrix. Then the matrix

$$\mathbf{A}^{(1)} = \mathbf{Q}^T\mathbf{AQ} = \mathbf{Q}^T\mathbf{QRQ} = \mathbf{RQ} = \mathbf{RQRR}^{-1} = \mathbf{RAR}^{-1}$$

is also upper Hessenberg.

Rmk. Every QR decomposition is $4n^3/3$ flops, thus the method converges very slowly. The solution for this can be the conversion of the original matrix to Hessenberg form, e.g. with Householder reflections ($4n^3/3$ flop, the eigenvalues do not change): $\mathbf{A} \to \mathbf{H}_1\mathbf{AH}_1 \to \mathbf{H}_2\mathbf{H}_1\mathbf{AH}_1\mathbf{H}_2$, etc., schematically

$$\begin{bmatrix} * & * & * & * \\ * & * & * & * \\ * & * & * & * \\ * & * & * & * \end{bmatrix} \to \begin{bmatrix} * & * & * & * \\ * & * & * & * \\ 0 & * & * & * \\ 0 & * & * & * \end{bmatrix} \to \begin{bmatrix} * & * & * & * \\ * & * & * & * \\ 0 & * & * & * \\ 0 & 0 & * & * \end{bmatrix}$$

For Hessenberg matrices, the QR decomposition can be performed with Givens rotations very fast ($3n^2$ flop).

Rmk. For symmetric matrices the Hessenberg form will be tridiagonal.

Solution of nonlinear equations

Nonlinear equations

Nonlinear equations

Example. $x^2 = 4 \sin x$. Find the real solutions.

Example. $x = \cos x$. Find the real solutions.

Example. $x^5 - 4x^4 + x^3 - x^2 + 4x - 4 = 0$. Find the real solutions. There is no solution formula that computes the roots from the coefficients.

Problem: We do not know whether the equation is solvable and how many solution does the equation have.

Separation of the roots

Thm. 41. (Bolzano) If a continuous function satisfies the conditions $f(a) \cdot f(b) < 0$ $(a < b)$, then there exists a constant $c \in (a, b)$ such that $f(c) = 0$.

Rmk. We calculate the function values at certain points, and if the values have different sign at neighbouring points then there is a root between these points.

Rmk. If the function is strictly monotone on a certain interval and there is a root in the interval, then the root is unique.

Rmk. It can be helpful if we draw the graphs of the functions. E.g. we draw the graphs of the left and the right hand side functions, and fix the interval which the intersection located in.

Polynomials

Evaluating polynomials

Horner's scheme (William George Horner (1786-1837, British))

$$a_n x^n + \ldots + a_1 x + a_0 = (\ldots((a_n x + a_{n-1})x + a_{n-2})\ldots)x + a_0$$

Rmk. There are altogether n additions in the formula. In 1954, Ostrowski proved that we need at least n additions to evaluate a polynomial.

Rmk. Victor Pan proved a similar theorem for the number of the multiplications in 1966.

Thm. 42. The roots of the polynomial $p(x) = a_n x^n + \ldots + a_1 x + a_0$ $(a_n, a_0 \neq 0)$ are located in the two circle rings centred in the origin with radius $R = 1 + A/|a_n|$ and $r = 1/(1 + B/|a_0|)$, where

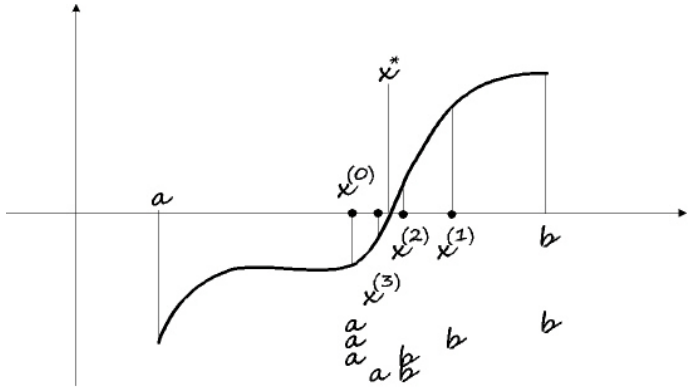
$$A = \max\{|a_{n-1}|, \ldots, |a_0|\},\ B = \max\{|a_n|, \ldots, |a_1|\}.$$

Rmk. In case of $p(x) = x^5 - 4x^4 + x^3 - x^2 + 4x - 4$ we have $1/2 \leq |x_k| \leq 5$.

Bisection method

Bisection method

$f(x) = 0 \longrightarrow$ Find the root $x^\star$.



$x_k \to x^\star$

Bisection method

Bisection method, $a < b$ and f are given, $f(a) < 0 < f(b)$.

for $k := 1 : k_{\max}$ **do**
  $x := a + (b - a)/2$
  $f := f(x)$
  if $f = 0$ **then**
    end
  else
    if $f > 0$ **then**
      $b = x$
    else
      $a = x$
    end if
  end if
end for

Bisection method

Rmk. Convergence order cannot be defined. But it is true the estimation

$$|e_k| \leq \frac{b-a}{2^{k+1}}.$$

This shows that we can expect one digit improvement after 3 steps.

Rmk. When we use only mantissas with two digits then we compute $(0.67 + 0.69)/2 = 1.36/2 \approx 1.4/2 = 0.7$, which is not between the two numbers. But $0.67 + (0.69 - 0.67)/2 = 0.67 + 0.02/2 = 0.67 + 0.01 = 0.68$.

Rmk. If the function has more than one roots then the method will surely find one of them.

Rmk. Other stopping conditions:

$$\frac{|x_k - x_{k-1}|}{|x_{k-1}|} \leq tol., \ |f(x_k)| \leq tol.$$

Newton's method

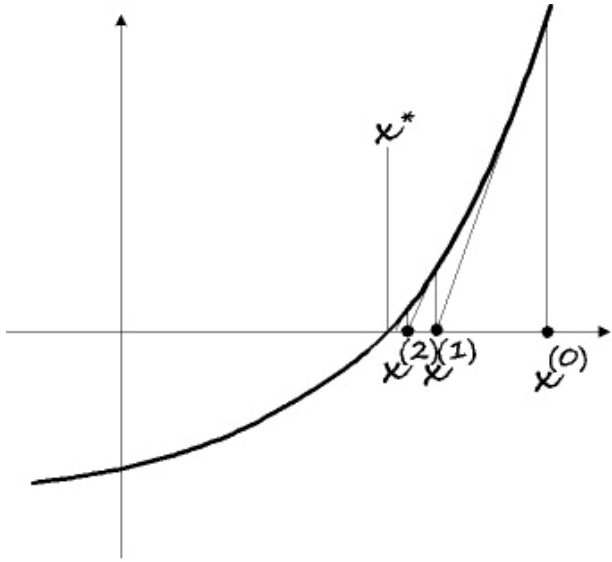
Newton's method

Newton (1669), Raphson (1690)

Newton's method, x_0 and f are given.

$x := x_0$
for $k := 1 : k_{\max}$ **do**
  $x := x - \frac{1}{f'(x)} f(x)$
  if $f(x) = 0$ **then**
    end
  end if
end for

Newton's method



Thm. 43. (Monotone convergence theorem) Let us suppose that $f \in C^2$ and that the first and the second derivatives of the function do not have zeros in the closed interval determined by the points $x^\star$ and x_0, moreover $f(x_0) \cdot f''(x_0) > 0$. Then the sequence $\{x_k\}$ generated by the Newton's method tends to $x^\star$ monotonically.

Proof: Let $x_0 > x^\star$ és $f(x_0) > 0, f''(x_0) > 0$ $(f'(x) > 0)$. We can see from the iteration

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$$

that $x_{k+1} \leq x_k$, that is the sequence is monotonically decreasing. It follows from the strict convexity that $x_k \geq x^\star$. Thus the sequence is convergent. Let us denote the limit with $\bar{x}^\star$.

Then

$$\underbrace{x_{k+1}}_{\to \bar{x}^\star} = \underbrace{x_k}_{\to \bar{x}^\star} - \frac{\overbrace{f(x_k)}^{\to f(\bar{x}^\star)}}{\underbrace{f'(x_k)}_{\to f'(\bar{x}^\star)}},$$

which implies that $\bar{x}^\star = x^\star$. ■

Thm. 44. Under the conditions of the previous theorem, the convergence of $\{x_k\}$ is of second order, moreover if $|f'(x)| \geq m_1 > 0$ and $|f''(x)| \leq M_2 < \infty$ in the interval determined by the points x_0 and $x^\star$ with appropriately chosen constants m_1 and M_2, then it is valid the estimation

$$|e_{k+1}| \leq \frac{M_2}{2m_1}|e_k|^2.$$

Newton's method

Proof: Let us use Taylor's expansion around the point x_k:

$$0 = f(x^\star) = f(x_k) + f'(x_k)(x^\star - x_k) + \frac{1}{2} f''(\xi)(x^\star - x_k)^2,$$

where ξ falls between $x^\star$ and x_k. From the reordering of the Newton's iteration:

$$0 = f(x_k) + f'(x_k)(x_{k+1} - x_k).$$

After subtraction:

$$0 = f'(x_k)(x_{k+1)} - x^\star) - \frac{1}{2} f''(\xi)(x^\star - x_k)^2.$$

Finally

$$|f'(x_k)| \cdot |e_{k+1}| = \frac{1}{2} |f''(\xi)| \cdot |e_k|^2.$$

Newton's method

Thus

$$\lim_{k\to\infty} \frac{|e_{k+1}|}{|e_k|^2} = \frac{|f''(x^\star)|}{2|f'(x^\star)|},$$

which shows that the order of the convergence is second order, indeed. Moreover

$$|e_{k+1}| = \frac{|f''(\xi)|}{2|f'(x_k)|} \cdot |e_k|^2 \leq \frac{M_2}{2m_1}|e_k|^2. \blacksquare$$

Rmk. Newton's method can be applied combined with the bisection method. First we approaches the root with the bisection method in order to fulfil the conditions of the above theorems, then we switch to Newton's method in order to accelerate the convergence.

A simple error estimation

Let us use Taylor's expansion around the point x_k:

$$0 = f(x^\star) = f(x_k) + f'(\xi)(x^\star - x_k),$$

where ξ is between the points x_k and $x^\star$.

Thus

$$|x^\star - x_k| = \frac{|f(x_k)|}{|f'(\xi)|} \leq \frac{|f(x_k)|}{m_1}.$$

Fixed point iterations

Fixed point iterations

Thm. 45. Let us suppose that the zero $x^\star \in [a,b]$ of the function f is a fixed point of the function $F : [a,b] \to [a,b]$. Let us suppose that F is a contraction with contraction coefficient q. Then the iteration $x_{k+1} = F(x_k)$ converges from arbitrary initial point $x_0 \in [a,b]$ to the unique solution of the equation $f(x) = 0$. Moreover

$$|x_k - x^\star| \leq \frac{q^k}{1-q}|x_1 - x_0|$$

Proof: The corollary of Banach's fixed point theorem (see page 364). ■

Rmk. In certain cases F can be given as $F(x) = x - g \cdot f(x)$, where g is a sufficiently chosen number that guarantees the contraction of F.

Rmk. The contraction property can be guaranteed supposing that F is continuous on $[a, b]$ and differentiable in (a, b), moreover there exists a number $0 \leq q < 1$, for which we have $|F'(x)| \leq q$, $\forall x \in (a, b)$ (Lagrange's mean value theorem).

Thm. 46. If, in the previous theorem, F is continuously differentiable at least r times and

$$F'(x^\star) = \ldots = F^{(r-1)}(x^\star) = 0$$

and $F^{(r)}(x^\star) \neq 0$, then the convergence order of the sequence $\{x_k\}$ is r and it is valid the estimation

$$|e_{k+1}| \leq \frac{M_r}{r!}|e_k|^r,$$

where M_r is an upper bound for the absolute value of the rth derivative of the function.

Proof: From the Taylor expansion around the point $x^\star$, we have

$$F(x_k) = F(x^\star) + \frac{F^{(r)}(\xi)}{r!}(x_k - x^\star)^r,$$

where ξ is between the numbers x_k and $x^\star$. That is

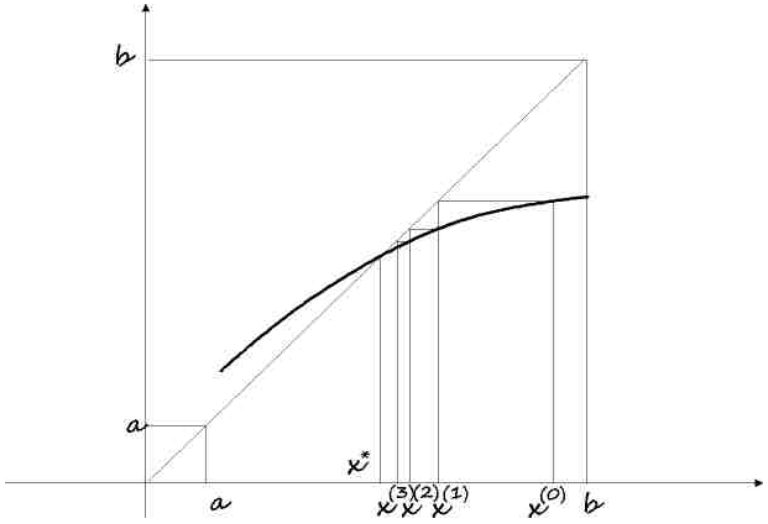
$$\lim_{k\to\infty} \frac{|e_{k+1}|}{|e_k|^r} = \frac{|F^{(r)}(x^\star)|}{r!}$$

that shows the rth order convergence of the method and the required estimation

$$|e_{k+1}| \leq \frac{M_r}{r!}|e_k|^r. \blacksquare$$

Rmk. Newton's method can be written also in a fixed point iteration form with the choice $g(x) = 1/f'(x)$. Its second order convergence could be proven also with the previous theorem.

Fixed point iterations



Systems of nonlinear equations

Newton's method for systems of nonlinear equations

Solve the nonlinear system for the solution $\overline{\mathbf{x}}^\star \in \mathbb{R}^n$

$$\overline{\mathbf{f}}(\overline{\mathbf{x}}) = \mathbf{0},\ \overline{\mathbf{f}} : \mathbb{R}^n \to \mathbb{R}^n.$$

Example. Find the solution of the system

$$\begin{aligned} x^2 + y - 5 &= 0 \\ x + y^2 - 3 &= 0 \end{aligned}$$

Let us approximate $\overline{\mathbf{f}}$ around the point $\overline{\mathbf{x}}_k$ with its first order Taylor expansion

$$\underbrace{\overline{\mathbf{f}}(\overline{\mathbf{x}}^\star)}_{\mathbf{0}} \approx \overline{\mathbf{f}}(\overline{\mathbf{x}}_k) + \overline{\mathbf{f}}'(\overline{\mathbf{x}}_k)(\overline{\mathbf{x}}^\star - \overline{\mathbf{x}}_k),$$

where $\overline{\mathbf{f}}'(\overline{\mathbf{x}}_k)$ is the Jacobian of the function $\overline{\mathbf{f}}$ at the point $\overline{\mathbf{x}}_k$. From this, we can approximate the solution as

$$\overline{\mathbf{x}}^\star \approx \overline{\mathbf{x}}_k - \left[\overline{\mathbf{f}}'(\overline{\mathbf{x}}_k)\right]^{-1} \overline{\mathbf{f}}(\overline{\mathbf{x}}_k).$$

Newton's method for systems of nonlinear equations

Using this approximation recursively, we arrive at an iterative method, the so-called Newton's method

$$\overline{\mathbf{x}}_{k+1} = \overline{\mathbf{x}}_k - \left[\overline{\mathbf{f}}'(\overline{\mathbf{x}}_k)\right]^{-1} \overline{\mathbf{f}}(\overline{\mathbf{x}}_k).$$

(We solve the system $\overline{\mathbf{f}}'(\overline{\mathbf{x}}_k)\overline{\mathbf{y}} = \overline{\mathbf{f}}(\overline{\mathbf{x}}_k)$ for $\overline{\mathbf{y}}$ then we update as $\overline{\mathbf{x}}_{k+1} = \overline{\mathbf{x}}_k - \overline{\mathbf{y}}$.)

Example.

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \end{bmatrix} = \begin{bmatrix} x_k \\ y_k \end{bmatrix} - \begin{bmatrix} 2x_k & 1 \\ 1 & 2y_k \end{bmatrix}^{-1} \begin{bmatrix} x_k^2 + y_k - 5 \\ x_k + y_k^2 - 3 \end{bmatrix}$$

Newton's method for systems of nonlinear equations

Thm. 47. Let us suppose that $\overline{\mathbf{f}}$ is continuously differentiable in a neighbourhood of $\overline{\mathbf{x}}^\star$, moreover let the Jacobians be bounded and Lipschitz continuous here. Then, when we start the Newton's iteration sufficiently close to $\overline{\mathbf{x}}^\star$, it will converge to $\overline{\mathbf{x}}^\star$ quadratically.

Rmk. The solution of a nonlinear system may be obtained also by fixed point iteration. If the equation $\overline{\mathbf{f}}(\overline{\mathbf{x}}) = \mathbf{0}$ is equivalent with the equation $\overline{\mathbf{x}} = \mathbf{F}(\overline{\mathbf{x}})$ with a suitably chosen function $\mathbf{F}$, and the iteration $\overline{\mathbf{x}}_{k+1} = \mathbf{F}(\overline{\mathbf{x}}_k)$ converges to $\overline{\mathbf{x}}^\star$, then $\overline{\mathbf{x}}^\star$ is the solution of $\overline{\mathbf{f}}(\overline{\mathbf{x}}) = \mathbf{0}$.

Relations between root-finding and minimization

Solve $\overline{\mathbf{f}}(\overline{\mathbf{x}}) = \mathbf{0} \Longrightarrow$ find the minimum of the multivariable function $\|\overline{\mathbf{f}}(\overline{\mathbf{x}})\|$

Find the minimum of the multivariable function $f(\overline{\mathbf{x}}) \Longrightarrow$ solve $\nabla f(\overline{\mathbf{x}}) = \mathbf{0}$

Thm. 48. Let us suppose that in a neighbourhood of $\overline{\mathbf{x}}^\star$ the multivariable function $f : \mathbb{R}^n \to \mathbb{R}$ is twice continuously differentiable. If the conditions

$$\nabla f(\overline{\mathbf{x}}^\star) = \mathbf{0}, \ \nabla^2 f(\overline{\mathbf{x}}^\star) \text{ is s.p.d.}$$

are fulfilled, where $\nabla^2 f(\overline{\mathbf{x}}^\star)$ denotes the Hessian of the function f at the point $\overline{\mathbf{x}}^\star$, then $\overline{\mathbf{x}}^\star$ is a local minimizer of the function f.

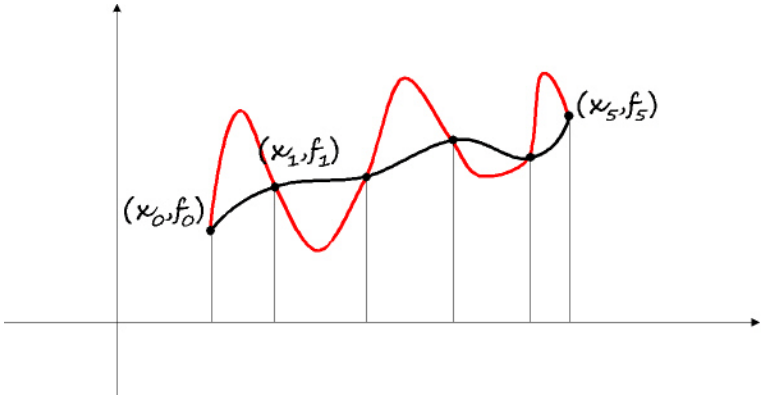
The possible local minimizer $\overline{\mathbf{x}}^\star$ may be found with the Newton's method applied to the equation $\nabla f(\overline{\mathbf{x}}) = \mathbf{0}$ as follows:

$$\overline{\mathbf{x}}_{k+1} = \overline{\mathbf{x}}_k - \left[\nabla^2 f(\overline{\mathbf{x}}_k)\right]^{-1} \nabla f(\overline{\mathbf{x}}_k).$$

Interpolation with polynomials

The interpolation problem

The problem to solve



The problem to solve

Let us suppose that we know the values of a function f only at $n+1$ distinct points (the so-called nodes) ((x_i, f_i) pairs ($i = 0, \ldots, n$), $x_i \neq x_j$, ha $i \neq j$).

Problem:

- Let us calculate the values of the function at other points;
- Let us calculate the derivative of the function;
- Let us calculate the extremizers of the function;
- Let us calculate its definite integral!

Solution: We give a functions ϕ with the properties $\phi(x_i) = f_i$ and we use this function in the calculation instead of the original (unknown) function. The functions ϕ are generally chosen to be polynomials, trigonometric polynomials (sin, cos) or piecewise polynomials.

Lagrange interpolation

Thm. 49. For all fixed $n+1$ nodes, there exists a unique polynomial L_n with degree at most n such that $L_n(x_i) = f_i$.

Proof: Let us choose the required polynomial to be $L_n(x) = \sum_{k=0}^{n} a_k x^k$. In order to satisfy the interpolation property, the following equalities must be valid:

$$L_n(x_i) = \sum_{k=0}^{n} a_k x_i^k = f_i \ (i = 0, \ldots, n).$$

This is a SLAE. Its coefficient matrix is a so-called Vandermonde matrix. Because $x_i \neq x_j$, if $i \neq j$, its determinant is not zero. Thus, the SLAE can be solved uniquely for the coefficients. ■

Interpolation with polynomials – Lagrangian form


Joseph-Louis Lagrange, 1736-1813, Italian (Giuseppe Lodovico Lagrangia)

Def. 50. For the fixed nodes $x_0, \ldots, x_n$, the polynomial

$$l_k(x) = \frac{(x - x_0) \ldots (x - x_{k-1})(x - x_{k+1}) \ldots (x - x_n)}{(x_k - x_0) \ldots (x_k - x_{k-1})(x_k - x_{k+1}) \ldots (x_k - x_n)}$$

$(k = 0, \ldots, n)$ is called the kth (it belongs to the point x_k) characteristic Lagrange polynomial.

Interpolation with polynomials – Lagrangian form

Trivially we have

$$l_k(x_i) = \begin{cases} 1, & \text{if } i = k, \\ 0, & \text{if } i \neq k. \end{cases}$$

Rmk. With the notation $w_{n+1}(x) = (x - x_0) \dots (x - x_n)$ (so-called nodal polynomial) the kth characteristic Lagrange polynomial can be written in the form

$$l_k(x) = \frac{w_{n+1}(x)}{(x - x_k) \cdot w'_{n+1}(x_k)}.$$

Lagrange form of the interpolation polynomial:

$$L_n(x) = \sum_{k=0}^{n} f_k l_k(x).$$

This polynomial trivially has degree at most n and its graph goes through the given points.

Interpolation with polynomials – Lagrangian form

Example. Find the interpolation polynomial to the points $(0, 2)$, $(1, 1)$ and $(3, 5)$!

The characteristic Lagrange polynomials are:

$$l_0(x) = \frac{(x-1)(x-3)}{(0-1)(0-3)} = \frac{1}{3}(x-1)(x-3),$$

$$l_1(x) = \frac{(x-0)(x-3)}{(1-0)(1-3)} = \frac{-1}{2}x(x-3),$$

$$l_2(x) = \frac{(x-0)(x-1)}{(3-0)(3-1)} = \frac{1}{6}x(x-1),$$

thus the interpolation polynomial is

$$p_2(x) = 2l_0(x) + 1l_1(x) + 5l_2(x) = x^2 - 2x + 2.$$

Interpolation error

Interpolation error

Thm. 51. [Cauchy, 1840] Let the function $f \in C^{n+1}$ and the nodal points $x_0, \ldots, x_n$ be given. Let us fix a point x and denote the interval determined by the nodal points and the point x by I_x. Let us denote the interpolation polynomial of f determined by the nodal points by $L_n f$. Then

$$E_n(x) := f(x) - (L_n f)(x) = \frac{f^{(n+1)}(\xi_x)}{(n+1)!} w_{n+1}(x).$$

Proof: If x is a nodal point, then the statement is trivial. Otherwise let

$$G(t) := E_n(t) - \frac{w_{n+1}(t)}{w_{n+1}(x)} E_n(x), \ t \in I_x,$$

which is a C^{n+1} function on the interval I_x. This function has $n+2$ roots.

Interpolation error

Then the function $G'(t)$ has at least $n+1$ roots, etc., and the function $G^{(n+1)}(t)$ has at least one root. Let us denote this root by ξ_x.

$$G^{(n+1)}(t) = f^{(n+1)}(t) - \frac{(n+1)!}{w_{n+1}(x)} E_n(x),$$

thus

$$G^{(n+1)}(\xi_x) = f^{(n+1)}(\xi_x) - \frac{(n+1)!}{w_{n+1}(x)} E_n(x) = 0,$$

hence

$$E_n(x) = \frac{f^{(n+1)}(\xi_x)}{(n+1)!} w_{n+1}(x). \blacksquare$$

Interpolation error

Thm. 52. If $f \in C^{\infty}[a,b]$ and the nodal points $x_0^{(n)}, \ldots, x_n^{(n)}$ are chosen from the interval $[a,b]$ $(n = 1, 2, \ldots)$, moreover, if $\exists M > 0$ such that $\max_{x\in[a,b]}\{|f^{(n)}|\} \leq M^n$, then $\max_{x\in[a,b]}\{|f(x) - (L_n f)(x)|\} \to 0$ if $n \to \infty$.

Proof: We apply the previous theorem:

$$|E_n(x)| = \frac{|f^{(n+1)}(\xi_x)|}{(n+1)!}|w_{n+1}(x)| \leq \frac{M^{n+1}}{(n+1)!}(b-a)^{n+1} \to 0,$$

if $n \to \infty$, even independently of x. ■

Rmk. We will generally use the notation M_n for an upper bound of $\max\{|f^{(n)}|\}$ on a predefined interval. Similarly, m_n will denote a non-negative lower bound for $\min\{|f^{(n)}|\}$.

Runge's example

Pl.: (Carl David Tolmé Runge, German, 1856–1927) Let us choose an equidistant partition of the interval $[-5, 5]$ and let us interpolate the function

$$f(x) = \frac{1}{1 + x^2}$$

in these points! Apparently, the interpolation polynomials do not tend to f. The difference is particularly emphasized at the two ends of the interval.

Interpolation error

Thm. 53. Let x be in the interval determined by the nodal points $x_0, \ldots, x_n$. Then the estimation

$$|w_{n+1}(x)| \leq \frac{n!}{4} h^{n+1}$$

is true for the nodal polynomial, where h is the greatest difference between the adjacent points.

Rmk. The estimations for the inner sub-intervals are less then that for the outer sub-intervals. Thus we can expect that if we choose the nodal point denser close to the ends of the interval, then the interpolation error can be decreased.

Rmk. Independently of x, we have

$$|E_n(x)| \leq \frac{M_{n+1}}{4(n+1)} h^{n+1}.$$

Chebyshev polynomials

Chebyshev polynomials


Pafnuty Lvovich Chebyshev, Russian, 1821-1894

Let us consider the polynomials defined with the recursion

$$T_0(x) = 1,\ T_1(x) = x,\ T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x)$$

on the interval $[-1, 1]$.

Example. $T_2(x) = 2x^2 - 1$, $T_3(x) = 4x^3 - 3x$.

Thm. 54.

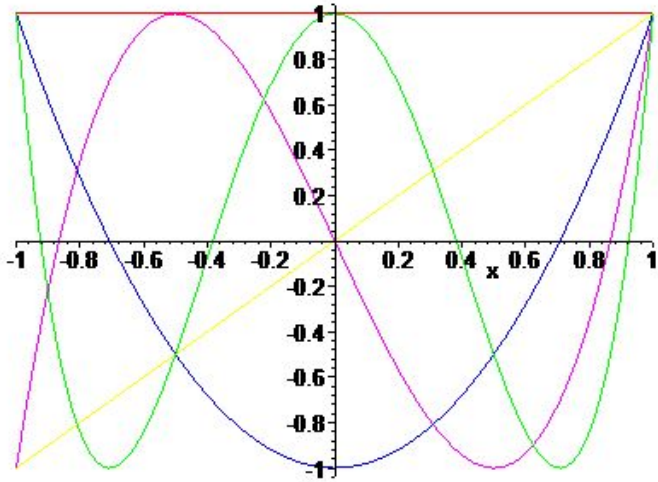
$$T_n(x) = \cos(n \cdot \arccos x).$$

Proof: The statement is trivial for $n = 0$ and $n = 1$. Let us assume that the statement is true for $n = k$. Then

$$\begin{aligned}
&2x\cos(k\arccos x) - \cos((k-1)\arccos x) \\
&= 2x\cos(k\arccos x) \\
&-(\cos(k\arccos x)x + \sin(k\arccos x)\sin(\arccos x)) \\
&= x\cos(k\arccos x) - \sin(k\arccos x)\sin(\arccos x) \\
&= \cos(\arccos x)\cos(k\arccos x) - \sin(k\arccos x)\sin(\arccos x) \\
&= \cos((k+1)\arccos x).
\end{aligned}$$

Thus the statement is true also for $k + 1$. ■

Chebyshev polynomials



Thm. 55.

$$|T_n(x)| \leq 1,$$

moreover the leading coefficient of $T_n(x)$ is 2^{n-1}.

Proof: Trivial. ■

Thm. 56. Let $\tilde{T}_n(x) = T_n(x)/2^{n-1}$, that is we norm the Chebyshev polynomial to leading coefficient 1. Then

$$\|\tilde{T}_n\|_{C[-1,1]} \leq \|p_n^{(1)}\|_{C[-1,1]},$$

where $p_n^{(1)}$ is an arbitrary polynomial with degree at most n and normed to leading coefficient 1.

Chebyshev polynomials

Proof: The extremizers of $T_n(x)$ are the points $t_k^{(n)} = \cos(k\pi/n)$ $(k = 0, \ldots, n)$. Indeed, $T_n(t_k^{(n)}) = \cos(n \arccos(t_k^{(n)})) = \cos(k\pi) = \pm 1$ (alternately). Thus, these points are the extremizers also of $\tilde{T}_n$.

We use reduction to absurdity. Thus, let us suppose that $\exists p_n^{(1)}$, such that

$$\|p_n^{(1)}\|_{C[-1,1]} < \|\tilde{T}_n\|_{C[-1,1]}.$$

Then the polynomial $q(x) = \tilde{T}_n(x) - p_n^{(1)}$ has degree at most $n-1$ and the sign of this polynomial is the same as that of the original polynomial. The polynomial $q(x)$ should change sign n times, which contradicts to the fact that the polynomial has degree at most $n-1$. ■

Chebyshev polynomials

Rmk.

$$|E_n(x)| = \frac{|f^{(n+1)}(\xi_x)|}{(n+1)!} |\underbrace{(x-x_0)(x-x_1)\dots(x-x_n)}_{=\tilde{T}_{n+1}(x)}|$$

Let us choose the nodal points to be the roots of the polynomial $T_{n+1}(x)$, that is the values

$$z_k = \cos\left(\frac{(2k+1)\pi}{2(n+1)}\right),\ k = 0, \dots, n!$$

In this case we have

$$|E_n(x)| \leq \frac{M_{n+1}}{(n+1)!}\frac{1}{2^n}$$

independently of x.

Newton interpolation

Newton form of the interpolation polynomial

Let the nodes (x_i, f_i) $(i = 0, \ldots, n)$ be given. Let us search for the interpolation polynomial in the so-called Newton form:

$$\begin{aligned} p_n(x) = & \, c_0 \\ & + c_1(x - x_0) \\ & + c_2(x - x_0)(x - x_1) \\ & + \ldots \\ & + c_n(x - x_0) \ldots (x - x_{n-1}). \end{aligned}$$

Rmk. This is a polynomial of degree at most n. Because the terms are linearly independent, all polynomials with degree at most n can be uniquely written in this form. Thus the coefficients c_k $(k = 0, \ldots, n)$ are uniquely determined.

Newton's divided differences

Def. 57. Let be given a function f and the nodal points $y_0, \ldots, y_k$. Then the uniquely defined leading coefficient of the interpolation polynomial defined by the points $(y_0, f(y_0)), \ldots, (y_k, f(y_k))$ is called Newton's divided difference of order k. Notation: $[y_0, \ldots, y_k]f$.

Rmk. Trivially, we have $[y_i]f = f(y_i)$.

Rmk. $[y_0, \ldots, y_k]f$ is uniquely defined and does not depend on the order of the nodal points $y_0, \ldots, y_k$.

Thm. 58. If L_{k-1} is the interpolation polynomial defined by the points $(x_0, f_0), \ldots, (x_{k-1}, f_{k-1})$ and L_k is the interpolation polynomial defined by the points $(x_0, f_0), \ldots, (x_k, f_k)$, then the relation

$$L_k(x) = L_{k-1}(x) + [x_0, \ldots, x_k]f \cdot (x - x_0) \ldots (x - x_{k-1})$$

is true.

Proof: $L_k - L_{k-1}$ is a polynomial of degree at most k, and it takes zero value at the points $x_0, \ldots, x_{k-1}$. Moreover its leading coefficient is the same as that of L_k: $[x_0, \ldots, x_k]f$. These conditions determine the polynomial

$$L_k(x) - L_{k-1}(x) = [x_0, \ldots, x_k]f \cdot (x - x_0) \ldots (x - x_{k-1})$$

uniquely, which gives the statement of the theorem. ■

Corollary: Based on the previous theorem, the c_k coefficients of the Newton form of the interpolation polynomial can be calculated as $c_k = [x_0, \ldots, x_k]f$.

Thm. 59. The Newton's divided differences fulfil the recursion formula

$$[x_0, \ldots, x_k]f = \frac{[x_1, \ldots, x_k]f - [x_0, \ldots, x_{k-1}]f}{x_k - x_0}.$$

Newton's divided differences

Proof: Let us denote the interpolation polynomial defined by the points $(x_1, f_1), \ldots, (x_k, f_k)$ by q_{k-1}. Then

$$L_k(x) = \frac{x - x_0}{x_k - x_0} q_{k-1}(x) + \frac{x_k - x}{x_k - x_0} L_{k-1}(x).$$

Indeed, this is a polynomial of degree at most k and $L_k(x_i) = f_i$ $(i = 0, \ldots, k)$. The statement of the theorem follows from the comparison of the leading coefficients. We have

$$\text{leading coef. of } L_k = \frac{\text{leading coef. of } q_{k-1}}{x_k - x_0} - \frac{\text{leading coef. of } L_{k-1}}{x_k - x_0},$$

that is

$$[x_0, \ldots, x_k]f = \frac{[x_1, \ldots, x_k]f - [x_0, \ldots, x_{k-1}]f}{x_k - x_0}.$$

Newton form of the interpolation polynomial

Calculation of the coefficients c_k:

By the definition we have: $[x_i]f = f_i$ $(i = 0, \ldots, n)$. According to the recursion formula:

$$[x_0, x_1]f = \frac{[x_1]f - [x_0]f}{x_1 - x_0}, \; [x_1, x_2]f = \frac{[x_2]f - [x_1]f}{x_2 - x_1},$$

$$[x_0, x_1, x_2]f = \frac{[x_1, x_2]f - [x_0, x_1]f}{x_2 - x_0}, \text{ etc.}$$

Example. Find the interpolation polynomials to the points (0,2), (1,1) és (3,5)! We construct a so-called Newton's divided difference table:

x_i	$f_i = [x_i]f$	$[.,.]f$	$[.,.,.]f$
0	$2 = c_0$		
		$-1 = c_1$	
1	1		$1 = c_2$
		2	
3	5		

Newton form of the interpolation polynomial

Thus the interpolation polynomial has the form:

$$2 + (-1)(x - 0) + 1(x - 0)(x - 1) = x^2 - 2x + 2.$$

For the calculation of the substitution value at a fixed point x we can use a Horner's scheme like rewriting:

$$2 + (-1)(x - 0) + 1(x - 0)(x - 1) = (1(x - 1) + (-1))(x - 0) + 2.$$

Generally:

$$\begin{aligned} L_n(x) = (([x_0, \ldots, x_n]f \cdot (x - x_{n-1}) \\ +[x_0, \ldots, x_{n-1}]f) \cdot (x - x_{n-2}) \\ +[x_0, \ldots, x_{n-2}]f) \cdot (x - x_{n-3}) \ldots + [x_0]f. \end{aligned}$$

Newton form of the interpolation polynomial

Addition of new nodes is easy: the new table:

x_i	$f_i = [x_i]f$	$[.,.]f$	$[.,.,.]f$	$[.,.,.,.]f$
0	$2 = c_0$			
		$-1 = c_1$		
1	1		$1 = c_2$	
		2		$1/2 = c_3$
3	5		$1/2$	
		1		
-1	1			

Thus the interpolation polynomial:

$$2 + (-1)(x-0) + 1(x-0)(x-1) + 1/2(x-0)(x-1)(x-3)$$

$$= x^3/2 - x^2 - x/2 + 2.$$

Comparison of Lagrange's and Newton's formulas

Lagrange

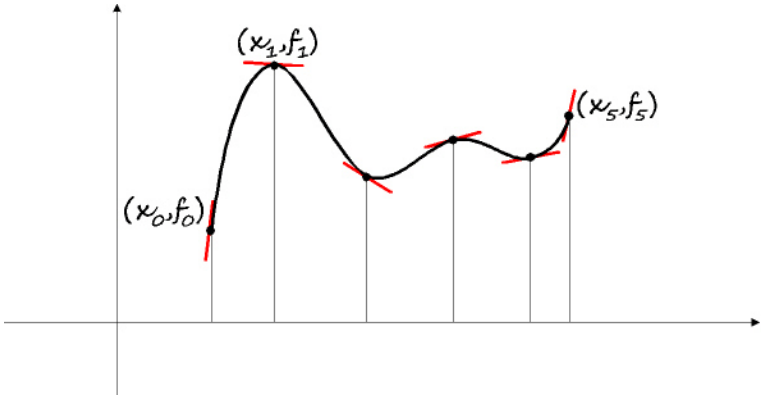
- Less accurate.
- The calculation of $p_n(x)$ for a fixed x costs $4n^2$ flop.
- Addition of new nodes is complicated.
- The characteristic Lagrange polynomials $l_k(x)$ are independent of the values f_k. Thus, if these values change, then the new interpolation polynomial can be obtained easily.

Newton

- More accurate.
- $3n^2/2$ flop is the calculation of the divided differences and additional $3n$ flop is the calculation of the function values..
- Addition of new nodes is easy.
- When the function values change, the polynomial must be newly calculated.

Hermite interpolation

Hermite interpolation



Hermite interpolation

Let the different nodal points $x_0, \ldots, x_n$ be given together with the function and derivative values

$$f_0^{(0)}, f_0^{(1)}, \ldots, f_0^{(m_0)}; \ldots; f_n^{(0)}, f_n^{(1)}, \ldots, f_n^{(m_n)}.$$

We would like to find the polynomial $p(x)$ that satisfies the conditions

$$p^{(i)}(x_k) = f_k^{(i)}, \ k = 0, \ldots, n; \ i = 0, \ldots, m_k.$$

We have altogether $m_0 + 1 + m_1 + 1 + \ldots m_n + 1 = n + 1 + \sum_{k=0}^{n} m_k =: N$ data. Thus, we can expect that a polynomial with degree at most $N - 1$ will be sufficient.

Thm. 60. There exists a unique polynomial H_{N-1} with degree at most $N - 1$ that satisfies the conditions

$$H_{N-1}^{(i)}(x_k) = f_k^{(i)}, \ k = 0, \ldots, n; \ i = 0, \ldots, m_k.$$

Hermite interpolation

Proof: Let $H_{N-1}(x) = a_0 + a_1 x + \ldots + a_{N-1} x^{N-1}$. Then we have to solve the SLAE:

$$\begin{bmatrix} 1 & x_0 & x_0^2 & \ldots & x_0^{N-1} \\ 0 & 1 & 2x_0 & \ldots & (N-1)x_0^{N-2} \\ \vdots & \vdots & \vdots & \ldots & \vdots \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \end{bmatrix} = \begin{bmatrix} f_0^{(0)} \\ f_0^{(1)} \\ f_0^{(2)} \\ \vdots \end{bmatrix}$$

We have here N equations and N unknowns, and the coefficient matrix is non-singular. Indeed, if a non-zero vector existed such that its product with the matrix is a non-zero vector, then the polynomial H_{N-1} would have N roots, which is impossible. ■

Hermite–Fejér interpolation polynomial: At each point only the function value and the derivative are given ($m_k = 1$, $k = 0, \ldots, n$). The the interpolation polynomial has degree at most $2n + 1$.

Hermite–Fejér interpolation

Construction of the interpolation polynomial with divided differences:

$$[x_0, x_1]f = \frac{f(x_0)}{(x_0 - x_1)} + \frac{f(x_1)}{(x_1 - x_0)}$$

Let $x_1 = x_0 + h$ and suppose that $h \to 0$. Then

$$\lim_{h\to 0}[x_0, x_0 + h]f = \lim_{h\to 0}\left(-\frac{f(x_0)}{h} + \frac{f(x_0 + h)}{h}\right) = f'(x_0).$$

Example. $x_0 = 0$, $x_1 = 1$, $f_0^{(0)} = 0$, $f_0^{(1)} = 0$, $f_1^{(0)} = 1$ és $f_1^{(1)} = 3$.

x_i	$f_i = [x_i]f$	$[.,.]f$	$[.,.,.]f$	$[.,.,.,.]f$
0	$0 = c_0$			
		$0 = c_1$		
0	0		$1 = c_2$	
		1		$1 = c_3$
1	1		2	
		3		
1	1			

$$H_3(x) = x^3$$

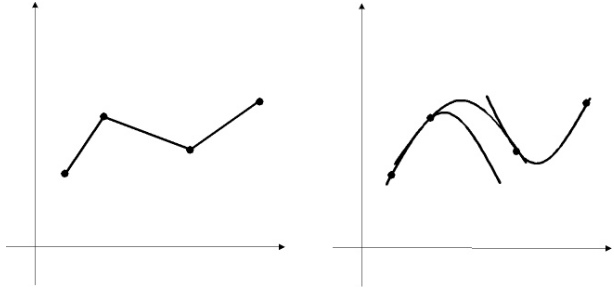
Spline interpolation

Spline interpolation - first and second degree splines

Spline = thin and flat bendable wood or metal strip used to draw curves.

When in an interpolation problem the nodes are given, then Chebyshev nodes cannot be used in order to decrease the interpolation error. In this case we generally interpolate with piecewise polynomials of lower degree. (The points that determine the sub-interals are called knots.)

Example. First and second degree splines



First-degree splines : interpolation error $= M_2 h^2/8$ (h is the maximum step-size).

Spline interpolation - cubic splines

Cubic splines. Let us construct a function s defined on the whole interval $[x_0, x_n]$ that possesses the following properties:

- $s(x_k) = f_k$ $(k = 0, \ldots, n)$,
- g, g', g'' are continuous,
- $s|_{[x_{i-1}, x_i]}$ is an at most cubic polynomial $(i = 1, \ldots, n)$.

The number of data: $4n$.
The number of the conditions: $2n + 2(n-1) = 4n - 2$.

We may choose two parameters arbitrarily:
a) natural cubic spline: $s''(x_0) = s''(x_n) = 0$.
b) clamped cubic spline: the values $s'(x_0)$ and $s'(x_n)$ are fixed.

Thm. 61. There is a unique function s that satisfies the above conditions.

Construction of the natural cubic splines

For the sake of simplicity let $x_k - x_{k-1} = h$ for all $k = 1, \ldots, n$. Let us consider the polynomial s_k that interpolates on the kth sub-interval.
Let

$$s_k(x_{k-1}) = f_{k-1},\ s_k(x_k) = f_k, s_k'(x_{k-1}) = d_{k-1},\ s_k'(x_k) = d_k,$$

where d_{k-1} and d_k are the for now unknown derivatives $s'(x_{k-1})$ and $s'(x_k)$. Let us apply the Hermite–Fejér interpolation:

x_i	$f_i = [x_i]f$	$[.,.]f$	$[.,.,.]f$	$[.,.,.,.]f$
x_{k-1}	f_{k-1}			
		d_{k-1}		
x_{k-1}	f_{k-1}		$\frac{f_k - f_{k-1}}{h^2} - \frac{d_{k-1}}{h}$	
		$\frac{f_k - f_{k-1}}{h}$		$\left(-2\frac{f_k - f_{k-1}}{h^2} + \frac{d_{k-1}+d_k}{h}\right)/h$
x_k	f_k		$\frac{d_k}{h} - \frac{f_k - f_{k-1}}{h^2}$	
		d_k		
x_k	f_k			

Construction of the natural cubic splines

The polynomial s_k and its second derivatives can be obtained. For these we can set $n+1$ equations: $n-1$ equations in the inner points and 2 equations in the end points. In this way we arrive at the SLAE:

$$\frac{h}{3}\begin{bmatrix} 2 & 1 & 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ 1 & 4 & 1 & 0 & 0 & \dots & 0 & 0 & 0 \\ 0 & 1 & 4 & 1 & 0 & \dots & 0 & 0 & 0 \\ \vdots & & & & & & & & \\ 0 & 0 & 0 & 0 & 0 & \dots & 1 & 4 & 1 \\ 0 & 0 & 0 & 0 & 0 & \dots & 0 & 1 & 2 \end{bmatrix} \begin{bmatrix} d_0 \\ d_1 \\ \vdots \\ d_n \end{bmatrix} = \begin{bmatrix} f_1 - f_0 \\ f_2 - f_0 \\ f_3 - f_1 \\ \vdots \\ f_n - f_{n-1} \end{bmatrix}$$

With the solution of the system for the derivatives d_k, the polynomials s_k can be obtained with Hermite–Fejér interpolation.

Construction of the natural cubic splines

Example. Let us determine the natural cubic interplation of the points (0,1), (1,2) and (2,0)! The system of equations

$$\frac{1}{3}\begin{bmatrix} 2 & 1 & 0 \\ 1 & 4 & 1 \\ 0 & 1 & 2 \end{bmatrix}\begin{bmatrix} d_0 \\ d_1 \\ d_2 \end{bmatrix} = \begin{bmatrix} 1 \\ -1 \\ -2 \end{bmatrix}.$$

We obtain that $d_0 = 7/4$, $d_1 = -1/2$, $d_2 = -11/4$ and the cubic polynomials that belong to the sub-intervals:

$$s_1(x) = -\frac{3}{4}x^3 + \frac{7}{4}x + 1,\ s_2(x) = \frac{3}{4}x^3 - \frac{9}{2}x^2 + \frac{25}{4}x - \frac{1}{2}.$$

Construction of the clamped cubic splines

The SLAE can be obtained similarly to the previous case. Now d_0 and d_n are fixed, and modify the system according to this fact.

$$\frac{h}{3}\begin{bmatrix} 4 & 1 & 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ 1 & 4 & 1 & 0 & 0 & \dots & 0 & 0 & 0 \\ 0 & 1 & 4 & 1 & 0 & \dots & 0 & 0 & 0 \\ \vdots & & & & & & & & \\ 0 & 0 & 0 & 0 & 0 & \dots & 1 & 4 & 1 \\ 0 & 0 & 0 & 0 & 0 & \dots & 0 & 1 & 4 \end{bmatrix} \begin{bmatrix} d_1 \\ \vdots \\ d_{n-1} \end{bmatrix} = \begin{bmatrix} f_2 - f_0 - d_0 h/3 \\ f_3 - f_1 \\ \vdots \\ f_{n-1} - f_{n-3} \\ f_n - f_{n-2} - d_n h/3 \end{bmatrix}$$

With the solution of the system for the derivatives d_k, the polynomials s_k can be obtained with Hermite–Fejér interpolation.

Construction of the clamped cubic splines

Example. Let us determine the clamped cubic spline interpolation to the points (0,1), (1,2) and (2,0), if $s'(0) = 0$ and $s'(2) = 1$!
Thus, $d_0 = 0$ and $d_2 = 1$. The "SLAE" simplifies to

$$\frac{1}{3}4d_1 = -1 - \frac{1}{3}0 - \frac{1}{3}1,$$

which gives $d_1 = -1$.

The cubic polynomials that belong to the sub-intervals are:

$$s_1(x) = -3x^3 + 4x^2 + 1, \; s_2(x) = 4x^3 - 17x^2 + 21x - 6.$$

Properties of cubic splines

Error estimate for cubic splines

Thm. 62. Let $f \in C^4[x_0, x_n]$ and let s be the cubic spline interpolating f on an equidistant mesh (with stepsize h) $x_0 < x_1 < \ldots < x_n$. Then

$$\|f^{(r)} - s^{(r)}\|_{C[x_0,x_n]} \leq C_r h^{4-r} \|f^{(4)}\|_{C[x_0,x_n]}, \ r = 0, 1, 2, 3,$$

where $C_0 = 5/384$, $C_1 = 1/24$, $C_2 = 3/8$ and $C_3 = 1$.

Minimum norm property of cubic splines

Thm. 63. Let $f \in C^2[a, b]$ and let s be the cubic spline interpolating f. Then

$$\int_{x_0}^{x_n} |s''(x)|^2 \, \mathrm{d}x \leq \int_{x_0}^{x_n} |f''(x)|^2 \, \mathrm{d}x,$$

where equality holds iff $f = s$.

Trigonometric interpolation

Interpolating trigonometric polynomials

Trigonometric polynomials

When we know that the data are the values of a periodic function, then it is advisable to interpolate with trigonometric functions instead of polynomials. .

Let us suppose that we know the values (f_k) of a 2π periodic function at the points $x_k = 2\pi k/(n+1) \in [0, 2\pi)$ $(k = 0, \ldots, n)$, where n is a positive natural number. Let us search for the interpolating function in the form

$$t_m(x) = a_0 + \sum_{j=1}^{m} (a_j \cos(jx) + b_j \sin(jx)),$$

which has to satisfy the equalities $t_m(x_k) = f_k$ $(k = 0, \ldots, n)$. t_m is called trigonometric polynomial of mth degree.

Trigonometric polynomials

Thus we have $2m+1$ coefficients and $n+1$ equations.

- If n is even, then we can expect that a polynomial with degree $m = n/2$ will be suitable.
- If n is odd, then introduce the notation $m = (n+1)/2$. Then we have $n+2$ coefficients and $n+1$ equations, that is the system is underdetermined. The term with the coefficient b_m has the following values at the nodes:

$$b_m \sin(mx_k) = b_m \sin\left(\frac{n+1}{2}\frac{2\pi k}{n+1}\right) = b_m \sin(\pi k) = 0.$$

Hence, the value of b_m can be chosen to be zero. We say that in this case the trigonometric polynomial (in the case if n is odd) is balanced.

Thm. 64. Let us suppose that the f_k values $(k = 0, \ldots, n)$ are given at the nodes $x_k = 2\pi k/(n+1)$. Let us suppose that n is odd. Then there exists a unique balanced trigonometric polynomial of degree $m = (n+1)/2$ denoted by t_m that satisfies the interpolation condition $t_m(x_k) = f_k$ $(k = 0, \ldots, n)$.

Proof: We will construct the polynomial. We work with complex numbers. Using the equality $e^{\mathtt{i}\phi} = \cos\phi + \mathtt{i}\sin\phi$ we obtain that

$$e^{\mathtt{i}jx} = \cos(jx) + \mathtt{i}\sin(jx), \;\; e^{-\mathtt{i}jx} = \cos(jx) - \mathtt{i}\sin(jx),$$

which results in the formulas

$$\cos(jx) = \frac{e^{\mathtt{i}jx} + e^{-\mathtt{i}jx}}{2}, \quad \sin(jx) = \frac{e^{\mathtt{i}jx} - e^{-\mathtt{i}jx}}{2\mathtt{i}}.$$

Interpolation with trigonometric polynomials

After back substitution to the original polynomial t_m and with the use of the interpolation property we obtain that

$$f_k = t_m(x_k) = a_0 + \sum_{j=1}^{m} \left(a_j \frac{e^{\mathtt{i}jx_k} + e^{-\mathtt{i}jx_k}}{2} + b_j \frac{e^{\mathtt{i}jx_k} - e^{-\mathtt{i}jx_k}}{2\mathtt{i}} \right)$$

$$= \overbrace{a_0}^{=:c_0} + \sum_{j=1}^{m-1} \left(\overbrace{\frac{a_j - b_j\mathtt{i}}{2}}^{=:c_j} e^{\mathtt{i}jx_k} + \overbrace{\frac{a_j + b_j\mathtt{i}}{2}}^{=:c_{2m-j}} e^{-\mathtt{i}jx_k} \right)$$

$$+ \underbrace{\overbrace{\frac{a_m}{2}}^{=:c_m} e^{\mathtt{i}mx_k} + \overbrace{\frac{a_m}{2}}^{=:c_m} e^{-\mathtt{i}mx_k}}_{c_m e^{\mathtt{i}mx_k}}$$

$$= \sum_{j=0}^{n} c_j e^{\mathtt{i}jx_k}, \quad k = 0, \ldots, n.$$

Interpolation with trigonometric polynomials

We applied the equality

$$e^{\mathtt{i}mx_k} = e^{-\mathtt{i}mx_k} = (-1)^k.$$

The original real coefficients can be calculated with the complex coefficients c_j:

$$a_0 = c_0, \quad a_m = c_m, \quad a_j = c_j + c_{2m-j} \ (j = 1, \ldots, m-1),$$

$$b_j = \mathtt{i}(c_j - c_{2m-j}) \ (j = 1, \ldots, m-1).$$

Because $f_k \in \mathbb{R}$, taking the complex conjugate of both sides we arrive at the form

$$f_k = \overline{f}_k = \sum_{j=0}^{n} \overline{c}_j e^{-\mathtt{i}jx_k}, \quad k = 0, \ldots, n,$$

that is $c_0, \ c_m \in \mathbb{R}$ és $c_{2m-j} = \overline{c}_j$, thus $a_j = 2\mathrm{Re}(c_j)$ és $b_j = -2\mathrm{Im}(c_j)$ $(j = 1, \ldots, m-1)$.

Interpolation with trigonometric polynomials

Let us introduce the notation

$$w = e^{-\mathtt{i}2\pi/(n+1)}.$$

w is a $(n+1)$th root of unity, because $w^{n+1} = 1$. Moreover,

$$e^{-\mathtt{i}jx_k} = w^{jk}$$

and using this notation we have to solve the SLAE

$$f_k = \sum_{j=0}^{n} c_j w^{-jk}, \ k = 0, \ldots, n$$

for the coefficients c_j. We show that this SLAE always has a unique solution, which fact shows that the trigonometric interpolation polynomial is unique.

Interpolation with trigonometric polynomials

With the notations

$$\overline{\mathbf{f}}_{n+1} = [f_0, \ldots, f_n]^T,\ \overline{\mathbf{c}}_{n+1} = [c_0, \ldots, c_n]^T,$$

$$\mathbf{F}_{n+1} \in \mathbb{R}^{(n+1)\times(n+1)},\ (\mathbf{F}_{n+1})_{jk} = w^{jk}$$

the SLAE can be written in the form

$$\overline{\mathbf{f}}_{n+1} = \mathbf{F}^H_{n+1}\overline{\mathbf{c}}_{n+1}.$$

Lemma. $\mathbf{F}_{n+1}\mathbf{F}^H_{n+1} = (n+1)\mathbf{I}_{n+1}$

Proof:

$$(\mathbf{F}_{n+1}\mathbf{F}^H_{n+1})_{kj} = \sum_{s=0}^{n} w^{ks}w^{-js} = \sum_{s=0}^{n} w^{s(k-j)} =$$

$$= \begin{cases} \text{n+1}, & \text{if } j = k, \\ \frac{(w^{k-j})^{n+1}-1}{w^{k-j}-1} = 0, & \text{if } j \neq k. \end{cases} \blacksquare$$

Interpolation with trigonometric polynomials

Let us return to the proof of the theorem. Let us multiply both sides of the SLAE

$$\overline{\mathbf{f}}_{n+1} = \mathbf{F}_{n+1}^{H}\overline{\mathbf{c}}_{n+1}$$

by the matrix $\mathbf{F}_{n+1}$. We obtain

$$\mathbf{F}_{n+1}\overline{\mathbf{f}}_{n+1} = \mathbf{F}_{n+1}\mathbf{F}_{n+1}^{H}\overline{\mathbf{c}}_{n+1},$$

that is the coefficients c_j can be calculated uniquely as

$$\overline{\mathbf{c}}_{n+1} = \frac{1}{n+1}\mathbf{F}_{n+1}\overline{\mathbf{f}}_{n+1}.\blacksquare$$

Let us introduce the notation $\hat{\mathbf{f}}_{n+1} := (n+1)\overline{\mathbf{c}}_{n+1}$.

Interpolation with trigonometric polynomials

Fourier analysis (Discrete Fourier Transform - DFT): We calculate the c_j complex Fourier coefficients from the data

$$\hat{f}_j = (n+1)c_j = \sum_{k=0}^{n} f_k w^{kj}, \quad j = 0, \ldots, n.$$

Fourier synthesis (Inverse Discrete Fourier Transform - IDFT): We calculate the nodal function values by the help of the Fourier coefficients c_j.

$$\frac{1}{n+1} \sum_{j=0}^{n} \hat{f}_j w^{-jk} = f_k, \quad k = 0, \ldots, n.$$

Interpolation with trigonometric polynomials

Rmk. If the function values f_k are real then $c_{2m-j} = \overline{c}_j$ $(j = 1, \ldots, m-1)$, that is these coefficients are complex conjugate of each other, $a_0 = c_0$ and $a_m = c_m$ are real values. Thus $a_j = 2\mathrm{Re}(c_j)$ and $b_j = -2\mathrm{Im}(c_j)$. From this, we obtain

$$a_0 = \frac{1}{n+1}\sum_{k=0}^{n} f_k, \quad a_m = \frac{1}{n+1}\sum_{k=0}^{n} f_k \cos(mx_k),$$

$$a_j = \frac{2}{n+1}\sum_{k=0}^{n} f_k \cos(jx_k) \quad (j = 1, \ldots, m-1),$$

$$b_j = \frac{2}{n+1}\sum_{k=0}^{n} f_k \sin(jx_k) \quad (j = 1, \ldots, m-1).$$

Interpolation with trigonometric polynomials

When the number of nodes is odd, then a similar theorem can be proven. The proof is also similar.

Thm. 65. Let us suppose that the function values f_k $(k = 0, \ldots, n)$ are given at the nodes $x_k = 2\pi k/(n+1)$. Let us suppose that n is even. Then, there exists a unique trigonometric polynomial t_m with degree $m = n/2$ such that $t_m(x_k) = f_k$ $(k = 0, \ldots, n)$.

Corollary: In this case the real discrete Fourier coefficients can be calculated as follows

$$a_0 = \frac{1}{n+1} \sum_{k=0}^{n} f_k,$$

$$a_j = \frac{2}{n+1} \sum_{k=0}^{n} f_k \cos(jx_k) \quad (j = 1, \ldots, m),$$

$$b_j = \frac{2}{n+1} \sum_{k=0}^{n} f_k \sin(jx_k) \quad (j = 1, \ldots, m).$$

Interpolation with trigonometric polynomials

Rmk. Let f be a 2π periodic function. Let us search the function in the so-called Fourier series form

$$f(x) = \alpha_0 + \sum_{j=1}^{\infty} (\alpha_j \cos(jx) + \beta_j \sin(jx)).$$

Then it can be shown that

$$\alpha_0 = \frac{1}{2\pi} \int_0^{2\pi} f(x)\,\mathrm{d}x,$$

$$\alpha_j = \frac{1}{\pi} \int_0^{2\pi} f(x)\cos(jx)\,\mathrm{d}x$$

$$\beta_j = \frac{1}{\pi} \int_0^{2\pi} f(x)\sin(jx)\,\mathrm{d}x.$$

Let us notice that the discrete Fourier coefficients are the approximations of the integrals above.

Interpolation with trigonometric polynomials

Example. $\overline{\mathbf{f}} = [0, 1, 4, 9]^T$, $n = 3$, $m = (n+1)/2 = 2$, $w = e^{-2i\pi/4} = -i$.

$$\hat{\overline{\mathbf{f}}}_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & w & w^2 & w^3 \\ 1 & w^2 & w^4 & w^6 \\ 1 & w^3 & w^6 & w^9 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 4 \\ 9 \end{bmatrix} =$$

$$= \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -\mathtt{i} & -1 & \mathtt{i} \\ 1 & -1 & 1 & -1 \\ 1 & \mathtt{i} & -1 & -\mathtt{i} \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 4 \\ 9 \end{bmatrix} = \begin{bmatrix} 14 \\ -4+8\mathtt{i} \\ -6 \\ -4-8\mathtt{i} \end{bmatrix}.$$

Thus $a_0 = 14/4 = 7/2$, $a_1 = -8/4 = -2$, $b_1 = -16/4 = -4$, $a_2 = -6/4 = -3/2$.

Fast Fourier transform

Fast Fourier transform (FFT)

The procedure was given already by Gauss in the early 1800s, but his work has been forgotten. After the advent of the computers the method was newly rediscoverd. James W. Cooley (IBM), John W. Tukey (Princeton), 1965.

$$\hat{f}_j = \sum_{k=0}^{n} f_k w^{kj}, \quad j = 0, \ldots, n.$$

The calculation of the discrete Fourier coefficients requires approximately $(n+1)^2$ complex multiplications, provided that the powers of w have been computed already (each coefficient requires $n+1$ multiplications).

How could we determine these coefficient with much less effort using the special form of the elements of the matrix.

Fast Fourier transform (FFT)

Example. In the previous problem we need to calculate the multiplication

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -\mathtt{i} & -1 & \mathtt{i} \\ 1 & -1 & 1 & -1 \\ 1 & \mathtt{i} & -1 & -\mathtt{i} \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 4 \\ 9 \end{bmatrix}.$$

Let us swap the columns of the matrix in order to put the odd numbered columns to the "left part" of the matrix!

$$\left[\begin{array}{cc|cc} 1 & 1 & 1 & 1 \\ 1 & -1 & -\mathtt{i} & \mathtt{i} \\ \hline 1 & 1 & -1 & -1 \\ 1 & -1 & \mathtt{i} & -\mathtt{i} \end{array}\right] \left[\begin{array}{c} 0 \\ 4 \\ \hline 1 \\ 9 \end{array}\right]$$

Here the two blocks on the left hand side is $\mathbf{F}_2$, the lower right block is the opposite of the upper right one, and the upper right block is

$$\begin{bmatrix} 1 & 0 \\ 0 & w \end{bmatrix} \mathbf{F}_2 = \begin{bmatrix} 1 & 0 \\ 0 & -\mathtt{i} \end{bmatrix} \mathbf{F}_2.$$

Fast Fourier transform (FFT)

In fact, we have to calculate only the product of the matrix $\mathbf{F}_2$ with the vectors $[0,4]^T$ and $[1,9]^T$, moreover the elements of the last product must be multiplied with the powers of w ($w^0, w^1, w^2, \ldots, w^{m-1}$), respectively.

General case: Let us suppose that $n+1$ is an even number. Then we need to perform the multiplication

$$\begin{bmatrix} \hat{f}_0 \\ \hat{f}_1 \\ \hat{f}_2 \\ \vdots \\ \hat{f}_{m-1} \\ \hat{f}_m \\ \hat{f}_{m+1} \\ \vdots \\ \hat{f}_n \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & w & w^2 & \ldots & w^n \\ 1 & w^2 & w^4 & \ldots & w^{2n} \\ \vdots & \vdots & \vdots & \ldots & \vdots \\ 1 & w^{m-1} & w^{2(m-1)} & \ldots & w^{(m-1)n} \\ 1 & w^m & w^{2m} & \ldots & w^{mn} \\ 1 & w^{(m+1)} & w^{2(m+1)} & \ldots & w^{(m+1)n} \\ \vdots & \vdots & \vdots & \ldots & \vdots \\ 1 & w^n & w^{2n} & \ldots & w^{n^2} \end{bmatrix} \begin{bmatrix} f_0 \\ f_1 \\ f_2 \\ \vdots \\ f_n \end{bmatrix}.$$

Fast Fourier transform (FFT)

Let us change the odd numbered columns forward!
Then the elements of the vector $\overline{\mathbf{f}}$ will be also rearranged.

We obtain the product:

$$\left[\begin{array}{cccc|cccc} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & w^2 & \dots & w^{n-1} & w & w^3 & \dots & w^n \\ 1 & w^4 & \dots & w^{2(n-1)} & w^2 & w^6 & \dots & w^{2n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & w^{2(m-1)} & \dots & w^{(m-1)(n-1)} & w^{m-1} & w^{3(m-1)} & \dots & w^{(m-1)n} \\ \hline 1 & w^{2m} & \dots & w^{m(n-1)} & w^m & w^{3m} & \dots & w^{mn} \\ 1 & w^{2(m+1)} & \dots & w^{(m+1)(n-1)} & w^{(m+1)} & w^{3(m+1)} & \dots & w^{(m+1)n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & w^{2n} & \dots & w^{n(n-1)} & w^n & w^{3n} & \dots & w^{n^2} \end{array}\right] \left[\begin{array}{c} f_0 \\ f_2 \\ f_4 \\ \vdots \\ f_{n-1} \\ \hline f_1 \\ f_3 \\ \vdots \\ f_n \end{array}\right].$$

The upper left block is $\mathbf{F}_m$ because w^2 is an mth root of unity. The lower left block is also $\mathbf{F}_m$. This can be checked easily using the fact that w is an $(n+1)$th root of unity. The upper right block can be written in the form $\mathbf{D}_m\mathbf{F}_m$ with the notation $\mathbf{D}_m = \mathrm{diag}(1, w, w^2, \dots, w^{m-1})$. The lower right block is the opposite of this.

Fast Fourier transform (FFT)

When we partition the vector $\hat{\mathbf{f}}$ and the rearranged $\overline{\mathbf{f}}$ (denoted by $\tilde{\mathbf{f}}$) accordingly, the product can be written in the form

$$\begin{bmatrix} \hat{\mathbf{f}}_1 \\ \hat{\mathbf{f}}_2 \end{bmatrix} = \mathbf{F}_{n+1}\overline{\mathbf{f}} = \begin{bmatrix} \mathbf{I}_m & \mathbf{D}_m \\ \mathbf{I}_m & -\mathbf{D}_m \end{bmatrix} \begin{bmatrix} \mathbf{F}_m & \mathbf{0} \\ \mathbf{0} & \mathbf{F}_m \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{f}}_1 \\ \tilde{\mathbf{f}}_2 \end{bmatrix}.$$

What can we win compared to the $(n+1)^2$ complex multiplication?
$\mathbf{F}_m\tilde{\mathbf{f}}_1$ and $\mathbf{F}_m\tilde{\mathbf{f}}_2$ require $((n+1)/2)^2$ complex multiplications each. The product of the diagonal matrix $\mathbf{D}_m$ and the vector $\mathbf{F}_m\tilde{\mathbf{f}}_2$ requires $(n+1)/2$ complex multiplicaitons. We do not need more multiplications. We must perform

$$2\left(\frac{n+1}{2}\right)^2 + \frac{n+1}{2}$$

complex multiplications.

Fast Fourier transform (FFT)

The algorithm become really fast if we use the above procedure in the case of the half sized matrices, too. This can be done repeatably if $n+1$ is a power of 2.
Let Q_l denote the number of complex multiplications of FFT when we use 2^l nodes. Then trivially

$$Q_l = 2Q_{l-1} + 2^{l-1}$$

and taking into the account that $Q_1 = 1$, we obtain with induction that

$$Q_l = l2^{l-1} = \frac{1}{2}(n+1)\log_2(n+1).$$

This is a significant drop in the number of operations:

$n+1$	DFT	FFT
$2^5 = 32$	1024	80
$2^{10} = 1024$	1048576	5120
$2^{20} = 1048576$	1099511627776	10485760

Numerical differentiation

The formulation of the problem

The formulation of the problem

Let us suppose that the values of the differentiable function f are known at the points $x_0, x_{\pm 1} = x \pm h, x_{\pm 2} = x \pm 2h, \ldots$ $(h > 0)$. Let us denote these values by $f_0, f_{\pm 1}, f_{\pm 2}$, etc., respectively. We approximate the derivatives of the function at the point x. These derivatives will be denoted by f_0', f_0'', etc.

Def. 66. Let us denote an arbitrary derivative of the sufficiently smooth function f at the point x_0 by Df. An approximation of this value is denoted by $\Delta f(h)$ (the approximation depends on the distance of the nodes). We say that the approximation $\Delta f(h)$ at the point x_0 is of order p (at least) if there is a real numbet $K > 0$ such that

$$|Df - \Delta f(h)| \leq Kh^p.$$

(That is $|Df - \Delta f(h)| = O(h^p)$.)

Forward difference

Forward difference

Based on the definition of the differential quotient

$$f' \approx \frac{f_1 - f_0}{h} =: \Delta f_+.$$

Moreover, if $f \in C^2$ then we have

$$\Delta f_+ = \frac{f_1 - f_0}{h} = \frac{(f_0 + f_0' h + f''(\xi)h^2/2) - f_0}{h} = f_0' + f''(\xi)h/2.$$

This shows that the order of the forward difference approximation is 1, that is halving the step-size h the error will be halved.

Backward difference

Backward difference

Based on the definition of the differential quotient

$$f' \approx \frac{f_0 - f_{-1}}{h} =: \Delta f_-.$$

Moreover, if $f \in C^2$ then we have

$$\Delta f_- = \frac{f_0 - f_{-1}}{h} = \frac{f_0 - (f_0 - f_0' h + f''(\xi) h^2/2)}{h} = f_0' - f''(\xi) h/2.$$

This shows that this approximation is of first order.

Centered difference

Centered difference

Let us investigate the arithmetic mean of the two previous approximations.

$$\Delta f_c := \frac{\Delta f_+ + \Delta f_-}{2} = \frac{f_1 - f_{-1}}{2h}.$$

Let us apply Taylor expansion at the point x_0. Let $f \in C^3$.

$$\Delta f_c = \frac{f_1 - f_{-1}}{2h}$$
$$= \frac{f_0 + f_0' h + f_0'' h^2/2 + f'''(\xi_1) h^3/6}{2h}$$
$$- \frac{f_0 - f_0' h + f_0'' h^2/2 - f'''(\xi_2) h^3/6}{2h} = f_0' + f'''(\xi)\frac{h^2}{6}.$$

Thus, this approximation has order 2.

Approximation of the second derivative

Approximation of the second derivative

The second derivative is the derivative of the first derivative.

$$\Delta^2 f_c = \frac{\Delta f_+ - \Delta f_-}{h} = \frac{f_1 - 2f_0 + f_{-1}}{h^2}.$$

Let us apply Taylor expansion again at the point x_0. Let $f \in C^4$.

$$\Delta^2 f_c =$$

$$= \frac{f_0 + f_0' h + f_0'' h^2/2 + f_0''' h^3/6 + f''''(\xi_1) h^4/24}{h^2} - \frac{2f_0}{h^2}$$

$$+ \frac{f_0 - f_0' h + f_0'' h^2/2 - f_0''' h^3/6 + f''''(\xi_2) h^4/24}{h^2} = f_0'' + f''''(\xi)\frac{h^2}{12}.$$

Thus, the approximation has order 2.

Other approximations

Other approximations

Rmk. A fourth order centered approximation of the first derivative

$$\frac{-f_2 + 8f_1 - 8f_{-1} + f_{-2}}{12h}.$$

Rmk. A second order forward and backward approximation of the first derivative

$$\frac{-3f_0 + 4f_1 - f_2}{2h}, \quad \frac{f_{-2} - 4f_{-1} + 3f_0}{2h}.$$

Rmk. The above formulas can be generalized easily to cases when the step-size is not equidistant.

Other approximations

Rmk.

- The derivative at x_0 of the polynomial fitted to the points $(x_0, f_0), (x_1, f_1)$ (at most first degree) is the same as the forward difference. The derivative at x_0 of the polynomial fitted to the points $(x_{-1}, f_{-1}), (x_0, f_0)$ (at most first degree) is the same as the backward difference.
- The derivative at x_0 of the polynomial fitted to the points $(x_{-1}, f_{-1}), (x_0, f_0), (x_1, f_1)$ (at most second degree) is the same as the centered difference, moreover, its second derivate gives the centered difference approximation of the second derivative.
- The derivative at x_0 of the third degree spline function fitted to the points $(x - h, f_{-1}), (x, f_0), (x + h, f_1)$ is the same as the the centered difference approximation of the first derivative.

Richardson extrapolation

Richardson extrapolation


Lewis Fry Richardson (1881-1953, British, physicist, metheorologist, psichologist)

Let the two values of the forward difference approximations of a function f at the point x_0 be: $\Delta f_+(h)$ and $\Delta f_+(h/2)$.

$$\Delta f_+(h) = f_0' + f''(\xi_h)\frac{h}{2},$$

$$\Delta f_+(h/2) = f_0' + f''(\xi_{h/2})\frac{h}{4}.$$

If h is small then $\xi_{h/2} \approx \xi_h$. Thus the approximation $2\Delta f_+(h/2) - \Delta f_+(h)$ may give a higher order approximation to the derivative. Indeed, the order of this approximation is 2.

Numerical integration

Motivation

Necessity of numerical integration

Newton–Leibniz formula:

$$\int_a^b f(x)\,\mathrm{d}x = F(b) - F(a).$$

We cannot use this formula if

- we cannot give the antiderivative of the function in closed form (e.g. $\sin x/x$, $\sin x^2$, e^{-x^2}).
- the computation of the antiderivative is complicated and time consuming.
- we know the values of the function at certain points only (e.g. measurements).

Requirements

Let us suppose that the function f is integrable on the interval $[a, b]$, and that we know the values of the function at the nodes

$$a \leq x_0 < x_1 < \ldots < x_n \leq b.$$

Let these function values denoted by $f_0, \ldots, f_n$, respectively. Then we should give an estimation to the integral by the help of the nodes and the function values.

Expectations:

- The approximation must be calculated easily,
- When we refine the nodes then the approximations must tend to the exact integral value of the functions,
- For sufficiently smooth functions the convergence must be fast.

Quadrature formulas

Quadrature formula

Let us denote the exact definite integral of the integrable function f by $I(f)$ and let one of its approximations at the given nodes be

$$I_n(f) = \sum_{k=0}^{n} a_k f_k.$$

Both the coefficients a_k (the so-called weights) and the function values f_k may depend on the number and the location of the nodes. The above formula is called quadrature formula.

Quadrature formula

Def. 67. We say that a quadrature formula is closed if it uses the function values both at a and b. If it does not use these values then the quadrature formula is open.

Let h be the larges step size between two adjacent nodes.

Def. 68. We say that the convergence order of the quadrature formula $I_n(f)$ is $r \geq 1$ (at least), if $|I(f) - I_n(f)| = O(h^r)$.

Def. 69. We say that the exactness order of the quadrature formula $I_n(f)$ is $r \geq 1$, if $I(p) = I_n(p)$ for all polynomials from P_{r-1} but there exists a polynomial p with degree r ($p \in P_r$) such that $I(p) \neq I_n(p)$.

Newton–Cotes formulas

Newton–Cotes formulas

Def. 70. We call a quadrature formula interpolation quadrature formula, if it approximates the integral with the integral of the interpolation polynomial fitted to the given function values.

Def. 71. If in an interpolation quadrature formula the nodes are equidistant (h), then the formula is called to be a Newton–Cotes-formula.


Roger Cotes (1682-1716, English)

Newton–Cotes formulas

The function f can be written in the form

$$f(x) = L_n(x) + r_n(x),$$

where L_n is the interpolation polynomial fitted to the function f on the given nodes, and r_n is the error term. Then the exact integral can be approximated as follows

$$\begin{aligned} I(f) &= \int_a^b f(x)\,\mathrm{d}x = \int_a^b L_n(x)\,\mathrm{d}x + \int_a^b r_n(x)\,\mathrm{d}x \\ &= \int_a^b \left(\sum_{k=0}^{n} f_k l_k(x)\right)\,\mathrm{d}x + \int_a^b r_n(x)\,\mathrm{d}x \\ &= \underbrace{\sum_{k=0}^{n} f_k \left(\overbrace{\int_a^b l_k(x)\,\mathrm{d}x}^{a_k}\right)}_{I_n(f)} + \int_a^b r_n(x)\,\mathrm{d}x. \end{aligned}$$

Newton–Cotes formulas

Here the weights depend on the interval of the integration. We can make them interval independent by changing the variable in the integral: let $x = a + (b-a)t$ ($t \in [0,1]$), thus $\mathrm{d}x/\mathrm{d}t = (b-a)$. In this way we have

$$a_k = \int_a^b l_k(x)\,\mathrm{d}x = \int_0^1 l_k(a+(b-a)t)(b-a)\,\mathrm{d}t$$
$$= (b-a)\int_0^1 l_k(a+(b-a)t)\,\mathrm{d}t,$$

where the last factor depends solely on the number of the interpolation nodes and their relative location. These values can be calculated and tabulated in advance: these are the so-called Newton–Cotes coefficients.

Closed Newton–Cotes formulas

With the setting

$$a = x_0 < x_1 < \ldots < x_n = b, \quad x_{k+1} - x_k = h = (b-a)/n$$

we obtain the weights

$$a_k = (b-a)N_{\mathsf{c}}^{n,k},$$

where the coefficients $N_{\mathsf{c}}^{n,k}$ are called closed Newton–Cotes coefficients.

$N_{\mathsf{c}}^{n,k}$	$k=0$	$k=1$	$k=2$	$k=3$	
$n=1$	$1/2$	$1/2$			$\leftarrow$ trapezoidal rule
$n=2$	$1/6$	$4/6$	$1/6$		$\leftarrow$ Simpson's rule
$n=3$	$1/8$	$3/8$	$3/8$	$1/8$	

Example. Applying the Simpson's rule to

$$\int_1^3 x^2 - 2x + 2 \,\mathrm{d}x = 2(1 \cdot 1/6 + 2 \cdot 4/6 + 5 \cdot 1/6) = 14/3$$

we obtain the exact integral value.

Open Newton–Cotes formulas

With the setting

$$a = x_{-1} < x_0 < \ldots < x_n < x_{n+1} = b, \quad x_{k+1} - x_k = h = (b-a)/(n+2)$$

we obtain the weights

$$a_k = (b-a)N_{\mathrm{o}}^{n,k},$$

where the coefficients $N_{\mathrm{o}}^{n,k}$ are called open Newton–Cotes coefficients.

$N_{\mathrm{o}}^{n,k}$	$k=0$	$k=1$	$k=2$	
$n=0$	1			$\leftarrow$ midpoint rule
$n=1$	$1/2$	$1/2$		
$n=2$	$2/3$	$-1/3$	$2/3$	

Thm. 72. A quadrature rule based on $n+1$ nodes is exact for P_n iff it is an interpolation quadrature formula.

Proof. $\Leftarrow$ Trivial.
$\Rightarrow$ It must be exact for all characteristic Lagrange polynomials $l_k(x)$. That is

$$\int_a^b l_k(x)\,\mathrm{d}x = \sum_{j=0}^{n} a_j l_k(x_j) = a_k. \blacksquare$$

Newton–Cotes formulas

Let $N^{n,k}$ denote the closed or the open Newton–Cotes coefficients.

Thm. 73.

$$\sum_{k=0}^{n} N^{n,k} = 1, \quad N^{n,k} = N^{n,n-k}.$$

Proof. In view of the previous theorem we have

$$\int_a^b 1 \, \mathrm{d}x = b - a = \sum_{k=0}^{n} (N^{n,k}(b-a)1) = (b-a) \sum_{k=0}^{n} N^{n,k}.$$

This proves the first statement. The second one follows from the symmetry $l_k(a+x) = l_{n-k}(b-x)$. ■

Rmk. If n is large then it is not practical to use the Newton–Cotes formulas. The Newton–Cotes coefficients $N^{n,k}$ may be negative that may cause cancellation. We generally use composite formulas.

Newton–Cotes formulas

Rmk. The Newton–Cotes formulas based on $n+1$ nodes are exact for P_n. If n is even, then they are exact also for P_{n+1}.

Namely, let p_{n+1} be a polynomial from P_{n+1}. Let us rewrite it to a polynomial of the term $(x-(a+b)/2)$.

$$p_{n+1}(x) = \alpha_{n+1}\left(x-\frac{a+b}{2}\right)^{n+1} + \underbrace{\alpha_n\left(x-\frac{a+b}{2}\right)^{n} + \ldots + \alpha_0}_{\text{The formula is exact for this.}},$$

moreover,

$$\int_a^b \underbrace{\alpha_{n+1}\left(x-\frac{a+b}{2}\right)^{n+1}}_{=:f(x)} \,\mathrm{d}x = (b-a)\sum_{k=0}^{n} \underbrace{N^{n,k}}_{N^{n,n-k}} \underbrace{f(x_k)}_{-f(x_{n-k})} = 0.$$

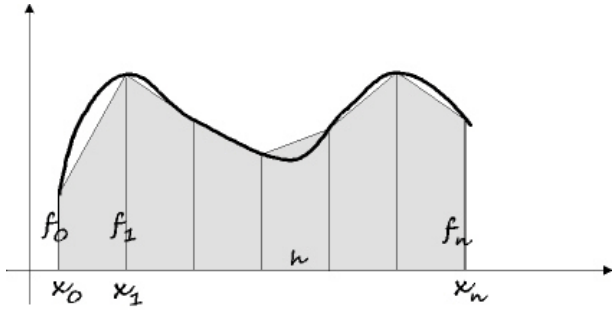
Thus the formula is exact for this polynomial.

Composite formulas

Composite trapezoidal rule

Let the nodes be equidistant with distance h. The so-called composite trapezoidal rule approximates the integral as follows:

$$I_{\text{trap}}(f) = \frac{h}{2} f_0 + h \sum_{k=1}^{n-1} f_k + \frac{h}{2} f_n = h \left(\frac{f_0}{2} + \sum_{k=1}^{n-1} f_k + \frac{f_n}{2} \right).$$



Composite trapezoidal rule

- Closed quadrature formula. The application of the formula is easy.
- $s_n \leq I_{\text{trap}}(f) \leq S_n$, that is, if the function is Riemann integrable, then the value of the formula tends to the exact integral value as the partition is refined.
- Order of exactness: 2. It is exact only on first degree polynomials. Order of the convergence is 2.

Example.

$$\int_0^1 \sin x/x \, \mathrm{d}x \approx 0.9460830704, \ n = 1/h.$$

n	$I_n(f)$	$\lvert I(f) - I_n(f)\rvert$
1	0.920735	0.25×10^{-1}
10	0.945832	0.25×10^{-3}
100	0.946080	0.25×10^{-5}
1000	0.9460830704	0.27×10^{-7}

Composite trapezoidal rule

Thm. 74. For $f \in C^2[a,b]$ functions, the error of the composite trapezoidal rule is

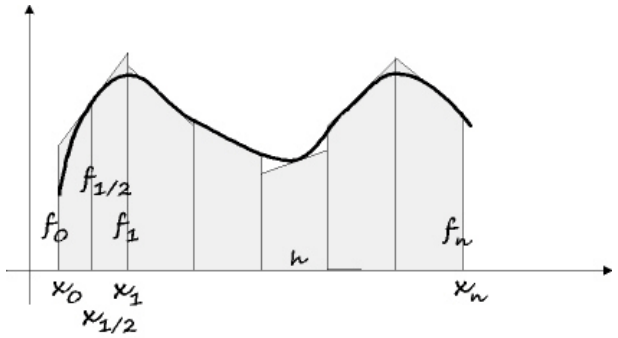
$$I(f) - I_{\text{trap}}(f) = -\frac{(b-a)h^2}{12} f^{(2)}(\eta),$$

where $\eta \in (a,b)$.

Rmk.

$$|I(f) - I_{\text{trap}}(f)| \leq \frac{(b-a)h^2}{12} M_2.$$

Composite midpoint rule



$$I_{\text{mid}}(f) = h(f_{1/2} + \ldots + f_{n-1/2}).$$

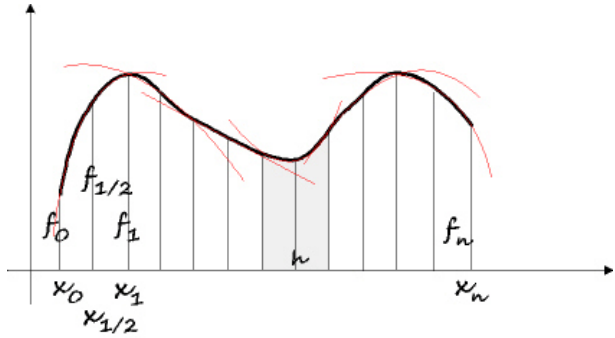
Open quadrature formula. Order: 2 (convergence and exactness).

Thm. 75. The error of the composite midpoint rule for $f \in C^2[a,b]$ functions is

$$I(f) - I_{\text{mid}}(f) = \frac{(b-a)h^2}{24} f^{(2)}(\eta),$$

where $\eta \in (a,b)$.

Composite Simpson's rule



$$I_{\mathsf{Simp}}(f) = \frac{h}{6}(f_0 + 4f_{1/2} + 2f_1 + 4f_{3/2} + 2f_2 + \ldots + 4f_{n-1/2} + f_n).$$

Closed quadrature formula. Order: 4 (convergence and exactness).

Composite Simpson's rule

Thm. 76. The error of the composite Simpson's rule for functions $f \in C^4[a,b]$ is

$$I(f) - I_{\text{Simp}}(f) = -\frac{(b-a)h^4}{2880} f^{(4)}(\eta),$$

where $\eta \in (a,b)$.

Rmk. In the case of a given partition:

$$I_{\text{Simp}}(f) = \frac{I_{\text{trap}}(f) + 2I_{\text{mid}}(f)}{3}.$$

Rmk. All the above quadrature formulas tend to the exact integral for Riemann integrable functions as $h \to 0$.

Gaussian quadrature

Gaussian quadrature

We have used equidistant nodes so far. We have seen, however, that these set of nodes are not efficient in interpolation problems.
We are looking for a better solution.

$$I_s(f) := \int_a^b s(x)f(x)\,\mathrm{d}x \approx \sum_{k=0}^{n} a_k f_k =: I_{n,s}(f),$$

where $a \leq x_0 < x_1 < \ldots < x_n \leq b$ are arbitrary nodes and s is a positive weight function.

If the quadrature formula is an interpolation quadrature formula, then we have

$$a_k = \int_a^b s(x)l_k(x)\,\mathrm{d}x$$

and the quadrature formula is exact for P_n.

How to choose the nodes to make the order of the exactness as large as possible?

Thm. 77. The interpolation quadrature formula

$$I_{n,s}(f) = \sum_{k=0}^{n} a_k f_k$$

is exact for P_{n+m} if and only if

$$\int_a^b w_{n+1}(x)s(x)p(x)\,\mathrm{d}x = 0$$

for all $p \in P_{m-1}$.

Rmk. The formula cannot be exact for P_{2n+2}. To see this, let us take $p = w_{n+1}$. From the equality

$$\int_a^b s(x)w_{n+1}^2(x)\,\mathrm{d}x = 0$$

we have $w_{n+1} \equiv 0$, which shows a contradiction.

Gaussian quadrature

Def. 78. Let $g_1, g_2 \in C[a,b]$. We call these functions orthogonal on the interval $[a,b]$ with respect to the positive weight function s, if

$$\int_a^b s(x) g_1(x) g_2(x)\,\mathrm{d}x = 0.$$

Thm. 79. Let us suppose that the polynomials $p_0, p_1, \ldots$ (the subscript denotes the degree of the polynomial) are pairwise orthogonal on $[a,b]$ with respect to the weight function s. Then all the zeros of these polynomials are real, single and lie in the interval $[a,b]$.

Construction of the Gaussian quadrature formulas: We orthogonalize the polynomials $1, x, \ldots$ with respect to the weight function: $p_0, p_1, \ldots$. We define the zeros of these polynomials $(x_0, \ldots, x_n)$ to be the nodes of the quadrature formula. We calculate the quadrature weights as $a_k = \int_a^b s(x) l_k(x)\,\mathrm{d}x$.
Then the form of the quadrature formula is

$$I_{n,s}(f) = \sum_{k=0}^{n} a_k f(x_k).$$

Gaussian quadrature

Legendre polynomials ($s(x) = 1$, $[-1, 1]$): $p_0 = 1$, $p_1 = x$, $p_2 = x^2 - 1/3$, etc.

Chebishev polynomials ($s(x) = 1/\sqrt{1 - x^2}$, $[-1, 1]$): $p_0 = 1$, $p_1 = x$, $p_2 = x^2 - 1/2$, $p_3 = x^3 - 3x/4$ etc.

Example. Let us construct the three-point Gauss–Chebyshev quadrature formula! The zeros of p_3 are 0 and $\pm\sqrt{3}/2$. These are the nodes. The weights

$$a_0 = \int_{-1}^{1} \frac{x(x - \sqrt{3}/2)}{-\sqrt{3}/2(-\sqrt{3}/2 - \sqrt{3}/2)} \frac{1}{\sqrt{1 - x^2}} \, \mathrm{d}x = \pi/3,$$

similarly $a_1 = a_2 = \pi/3$. Thus the formula is:

$$\int_{-1}^{1} \frac{f(x)}{\sqrt{1 - x^2}} \, \mathrm{d}x \approx \frac{\pi}{3}(f(-\sqrt{3}/2) + f(0) + f(\sqrt{3}/2)).$$

Gaussian quadrature

Some nodes and weights of Gaussian quadrature.

	Gauss–Legendre		Gauss–Chebyshev	
	$s(x) = 1$		$s(x) = 1/\sqrt{1-x^2}$	
Nr. of points	Nodes	Weights	Nodes	Weights
1	0	2	0	π
2	$\frac{-1}{\sqrt{3}}, \frac{1}{\sqrt{3}}$	1,1	$\frac{-1}{\sqrt{2}}, \frac{1}{\sqrt{2}}$	$\frac{\pi}{2}, \frac{\pi}{2}$
3	$-\sqrt{\frac{3}{5}}, 0, \sqrt{\frac{3}{5}}$	$\frac{5}{9}, \frac{8}{9}, \frac{5}{9}$	$\frac{-\sqrt{3}}{2}, 0, \frac{\sqrt{3}}{2}$	$\frac{\pi}{3}, \frac{\pi}{3}, \frac{\pi}{3}$

Numerical solution of initial value problems

Introduction

Examples

- Motion of a pendulum ($\phi(0) = \alpha$)

$$\phi'(t) = \pm\sqrt{\frac{2g}{l}}\sqrt{\cos\phi(t) - \cos\alpha}$$

- (Alfred James) Lotka (1925, USA) - (Vito) Volterra (1926, Italian) predator-pray model ($u(0)$, $v(0)$ are given)

$$\begin{aligned} u'(t) &= u(t)(2 - v(t)), \\ v'(t) &= v(t)(u(t) - 1). \end{aligned}$$

- Deflection of a rod ($y(0) = y(L) = 0$)

$$EIy''(x) + P\cos(y(x)) = 0.$$

The first two examples are so-called initial value problems, while the third one is a so-called boundary value problem.

Initial value problems

$$\overline{\mathbf{y}}' = \mathbf{f}(x, \overline{\mathbf{y}}), \quad \overline{\mathbf{y}}(x_0) \text{ given}$$

where $\overline{\mathbf{y}} : [a, b] \to \mathbb{R}^n$ is the unknown function, $\mathbf{f} : [a, b] \times \mathbb{R}^n \to \mathbb{R}^n$, moreover $x_0 \in [a, b]$.

Other forms:

$$\overline{\mathbf{y}}'(x) = \mathbf{f}(x, \overline{\mathbf{y}}(x)),$$

or componentwise

$$\begin{aligned} y_1'(x) &= f_1(x, y_1, \ldots, y_n), \\ &\vdots \\ y_n'(x) &= f_n(x, y_1, \ldots, y_n), \end{aligned}$$

Order of the equation: the highest order of the derivative of the unknown function that appear in the equation.

Initial value problems

Example. Higher order equations with one unknown can be rewritten to a system of ordinary differential equations. In case of

$$y'' + 3y'y + xy = 0, \quad y(x_0), y'(x_0) \text{ given}$$

we can rewrite the equation as

$$\begin{aligned} y_1'(x) &= y_2, \quad y_1(x_0) \text{ given} \\ y_2'(x) &= -xy_1 - 3y_2y_1, \quad y_2(x_0) \text{ given.} \end{aligned}$$

Solution: A function $\overline{\mathbf{y}}$ that is differentiable sufficiently many times, fulfils the initial condition and if we substitute it back into the equation then we arrive at an identity.

Rudolf Otto Sigismund Lipschitz (1832 - 1903, German)

Def. 80. We say that the function $\mathbf{f} : [a, b] \times \mathbb{R}^n \to \mathbb{R}^n$ is Lipschitz continuous in its second argument, if $\exists L \geq 0$ such that for all $x \in [a, b]$ and $\overline{\mathbf{z}}_1, \overline{\mathbf{z}}_2 \in \mathbb{R}^n$ we have

$$\|\mathbf{f}(x, \overline{\mathbf{z}}_1) - \mathbf{f}(x, \overline{\mathbf{z}}_2)\| \leq L\|\overline{\mathbf{z}}_1 - \overline{\mathbf{z}}_2\|.$$

Thm. 81. If the right hand side function $\mathbf{f}$ of the initial value problem

$$\overline{\mathbf{y}}' = \mathbf{f}(x, \overline{\mathbf{y}}), \quad \overline{\mathbf{y}}(a) \text{ given}$$

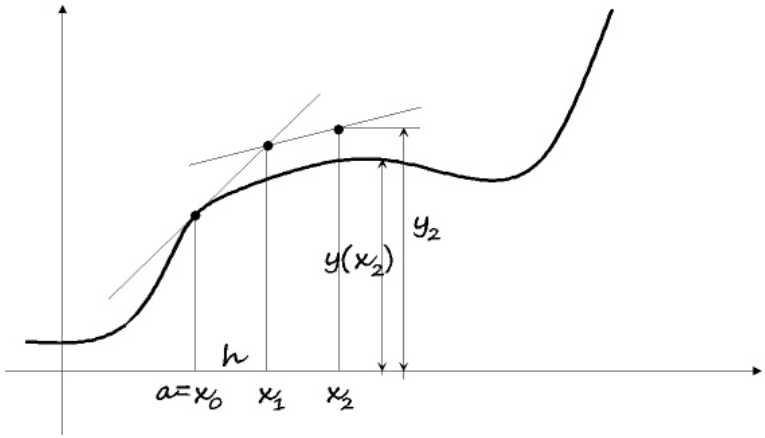
is continuous in its first argument on $[a, b]$ and Lipschitz continuous in its second argument then the problem has a unique solution, which is continuously differentiable.

Explicit Euler method

Explicit Euler method (EE)

The method was published by Euler in a three-volume book between 1768 and 1770.

$$y'(x) = f(x, y(x)), \quad y(x_0) \text{ given.}$$



Explicit Euler method (EE)

We define a mesh on the interval $[x_0, x_{\max}]$ and we approximate the value of the solution function only at these points.

The mesh is $x_k = x_0 + hk$ $(k = 0, 1, \ldots, N_h)$, where h is an arbitrary positive step size. N_h is the maximum positive integer that satisfies $hN_h \leq x_{\max}$. Let us denote the approximations in the mesh points by $\overline{\mathbf{y}}_k$.

The formula of the Explicit Euler method is:

$$\overline{\mathbf{y}}_{k+1} = \overline{\mathbf{y}}_k + h\mathbf{f}(x_k, \overline{\mathbf{y}}_k), \quad \overline{\mathbf{y}}_0 \text{ is known from } \overline{\mathbf{y}}(x_0).$$

Def. 82. The iteration formula that prescribes how to calculate the approximation values at the mesh points is called numerical scheme (or method).

The general numerical schemes we will deal with have the form

$$\overline{\mathbf{y}}_{k+1} = \overline{\mathbf{y}}_k + h\Phi(h, x_k, \overline{\mathbf{y}}_{k+1}, \overline{\mathbf{y}}_k, \ldots, \overline{\mathbf{y}}_{k+1-s}),$$

where Φ is the so-called increment function (EE-case: $\Phi = \mathbf{f}(x_k, \overline{\mathbf{y}}_k)$), and s is a positive integer. Notice that the other mesh points can be expressed with x_k and h.

Def. 83. A numerical scheme (or method) is explicit, if Φ is independent of $\overline{\mathbf{y}}_{k+1}$, that is we do not need to solve equations to get $\overline{\mathbf{y}}_{k+1}$. Otherwise the scheme is implicit.

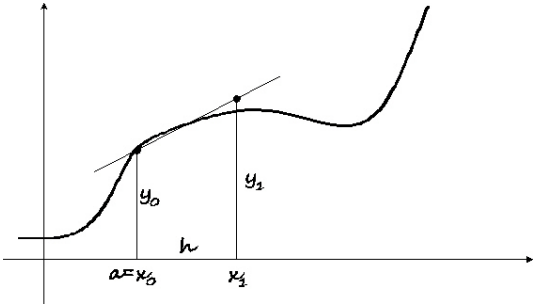
Def. 84. The number s is called the number of steps of the scheme. The scheme is called one-step scheme (method) if $s = 1$ (only the data at the kth point are used to the approximation at the $(k+1)$th point). The scheme is a multistep scheme if $s > 1$.

Example. The EE (scheme) method is a one-step explicit (scheme) method.

In the sequel we will investigate one-step methods only. The multistep methods are considered in a separate section.

Implicit Euler and Crank–Nicolson methods

Implicit Euler method (IE)



The scheme is

$$\overline{\mathbf{y}}_k = \overline{\mathbf{y}}_{k+1} - h\mathbf{f}(x_{k+1}, \overline{\mathbf{y}}_{k+1}),$$

where we have to solve a non-linear system of equations in each iteration step. This can be solved e.g. with fixed point iteration starting from the estimate in the previous point $\overline{\mathbf{y}}_k$.

Rmk. The implicit Euler method is a one-step implicit method.

Crank–Nicolson method (CN, trapezoidal)

John Crank (1916 - 2006), Phyllis Nicolson (1917 - 1968), English.

The scheme

$$\overline{\mathbf{y}}_{k+1} = \overline{\mathbf{y}}_k + \frac{h}{2}(\mathbf{f}(x_k, \overline{\mathbf{y}}_k) + \mathbf{f}(x_{k+1}, \overline{\mathbf{y}}_{k+1})).$$

This is also a one-step implicit scheme.

Other derivations

Numerical integration:

$$y'(x) = f(x,y) \Rightarrow \int_{x_0}^{x_0+h} y'(x)\,\mathrm{d}x = \int_{x_0}^{x_0+h} f(x,y(x))\,\mathrm{d}x$$

Thus

$$y(x_0+h) - y(x_0) = \int_{x_0}^{x_0+h} f(x,y(x))\,\mathrm{d}x$$

$$\approx \begin{cases} f(x_0,y_0)h, & (EE) \\ f(x_0+h, y(x_0+h))h, & (IE) \\ (f(x_0,y_0)+f(x_0+h,y(x_0+h)))h/2 & (CN). \end{cases}$$

Other derivations

Numerical differentiation:

We change the derivative with the forward difference approximation.

$$\frac{y(x_0+h)-y(x_0)}{h} \approx f(x_0, y(x_0)),$$

After rearrangement we arrive at the scheme of the EE method.

Taylor's method

$$y(x_0+h) = y(x_0) + \underbrace{y'(x_0)}_{f(x_0,y(x_0))} h + \frac{y''(x_0)h^2}{2} + \frac{y'''(x_0)h^3}{6} + etc.$$

When we stop after the first order term, then we get the EE scheme. If we can compute the derivatives of function $f(x, y)$ with respect to x, then we can produce the Taylor's series of the solution to arbitrary order.

The θ-method

Let $\theta \in [0,1]$ be an arbitrary parameter and let us consider the numerical integration formula

$$y(x_0+h) - y(x_0) = \int_{x_0}^{x_0+h} f(x, y(x))\,\mathrm{d}x$$
$$\approx h(\theta f(x_0+h, y(x_0+h)) + (1-\theta) f(x_0, y(x_0))).$$

Special cases:

- The $\theta = 0$ case gives the EE scheme,
- the $\theta = 1$ case gives the IE scheme,
- and the $\theta = 1/2$ case gives the CN scheme.

Consistency, stability, convergence

A numerical experiment (EE method)

Example.

$$y'(x) = \frac{y(x) + x}{y(x) - x}, \quad y(0) = 1.$$

Exact solution

$$y(x) = x + \sqrt{1 + 2x^2}.$$

$h = 1/k$	$y_1 - y(x_1)$	$y_k - y(1)$
$1/2$	-0.2247	-0.2321
$1/4$	-0.0607	-0.1065
$1/8$	-0.0155	-0.0510
$1/16$	-0.0039	-0.0249

The error is second order at the first mesh point and first order at the point $x = 1$.

Convergence

Let $\overline{\mathbf{e}}_k$ denote the difference $\overline{\mathbf{y}}_k - \overline{\mathbf{y}}(x_k)$ $(k = 0, \ldots, N_h)$.

Def. 85. A numerical scheme (method) is said to be convergent, if

$$\max_{k=1,\ldots,N_h} \|\overline{\mathbf{e}}_k\| = O(h^r)$$

$(r \geq 1)$, and we say that the order of the convergence is (at least) r.

Def. 86. Local truncation error (LTE): the remainder when we pretend that the exact solution satisfies the scheme is written in the form $h\boldsymbol{\tau}_{k+1}$. $\boldsymbol{\tau}_{k+1}$ is called the local truncation error at the point x_{k+1}.

Example. For one-step schemes we have

$$\overline{\mathbf{y}}(x_{k+1}) = \overline{\mathbf{y}}(x_k) + h\Phi(h, x_k, \overline{\mathbf{y}}(x_k), \overline{\mathbf{y}}(x_{k+1})) + h\boldsymbol{\tau}_{k+1}.$$

Consistency

Example. Computation of the local truncation error for the EE method ($\overline{\mathbf{y}} \in C^2$):

$$\begin{aligned}\boldsymbol{\tau}_{k+1} &= \frac{\overline{\mathbf{y}}(x_{k+1}) - \overline{\mathbf{y}}(x_k)}{h} - \mathbf{f}(x_k, \overline{\mathbf{y}}(x_k)) \\ &= \frac{\overline{\mathbf{y}}(x_k) + \overline{\mathbf{y}}'(x_k)h + \overline{\mathbf{y}}''(\xi_k)h^2/2 - \overline{\mathbf{y}}(x_k)}{h} - \mathbf{f}(x_k, \overline{\mathbf{y}}(x_k)) \\ &= \overline{\mathbf{y}}''(\xi_k)h/2.\end{aligned}$$

Thus, all local truncation errors are bounden by $M_2 h/2$.

Def. 87. If all the truncation errors are bounded by Ch^r ($C \geq 0$ constant and $r \geq 1$), then the numerical scheme is called consistent with the order of consistency r.

Example. The EE method ($\overline{\mathbf{y}} \in C^2$) is consistent with consistency order 1.

Stability

Def. 88. A numerical scheme is called to be (zero-)stable on the interval $[x_0, x_{\max}]$ if there are numbers $K > 0$ (independent of h) and $h_0 > 0$ such that

$$\max_{k=1,\ldots,N_h} \|\overline{\mathbf{y}}_k - \overline{\mathbf{z}}_k\| \leq K \|\overline{\mathbf{y}}_0 - \overline{\mathbf{z}}_0\|$$

if $0 < h < h_0$. ($\overline{\mathbf{z}}_k$ is a vector sequence starting from $\overline{\mathbf{z}}_0$ and defined by the numerical scheme.)

Thm. 89. (The equivalence theorem.) Let us suppose that the order of the consistency of a numerical scheme is $r \geq 1$. Then the necessary and sufficient condition of the convergence is the stability. The order of the convergence is r.

Thm. 90. Let us consider the initial value problem

$$\overline{\mathbf{y}}' = \mathbf{f}(x, \overline{\mathbf{y}}), \quad \overline{\mathbf{y}}(x_0) \text{ given}$$

with a solution $\overline{\mathbf{y}} \in C^2$. Then the explicit Euler method is convergent and the convergence order is 1, moreover we have

$$\|\overline{\mathbf{e}}_k\| \leq e^{(x_{\max}-x_0)L} h(x_{\max} - x_0) M_2/2.$$

Convergence

Proof. We prove only the stability, which gives the convergence due to the equivalence theorem.
We start from two arbitrary vector sequences that are generated by the explicit Euler scheme

$$\overline{\mathbf{y}}_{k+1} = \overline{\mathbf{y}}_k + h\mathbf{f}(x_k, \overline{\mathbf{y}}_k),$$
$$\overline{\mathbf{z}}_{k+1} = \overline{\mathbf{z}}_k + h\mathbf{f}(x_k, \overline{\mathbf{z}}_k).$$

We subtract the two equalities and use the Lipschitz continuity of the function $\mathbf{f}$.

$$\|\overline{\mathbf{y}}_{k+1} - \overline{\mathbf{z}}_{k+1}\| = \|\overline{\mathbf{y}}_k - \overline{\mathbf{z}}_k\| + h\|\mathbf{f}(x_k, \overline{\mathbf{y}}_k) - \mathbf{f}(x_k, \overline{\mathbf{z}}_k)\| \leq$$
$$\leq \|\overline{\mathbf{y}}_k - \overline{\mathbf{z}}_k\| + hL\|\overline{\mathbf{y}}_k - \overline{\mathbf{z}}_k\| \leq (1 + hL)\|\overline{\mathbf{y}}_k - \overline{\mathbf{z}}_k\|.$$

Thus we have

$$\|\overline{\mathbf{y}}_k - \overline{\mathbf{z}}_k\| \leq (1 + hL)^k\|\overline{\mathbf{y}}_0 - \overline{\mathbf{z}}_0\| \leq e^{khL}\|\overline{\mathbf{y}}_0 - \overline{\mathbf{z}}_0\| = e^{(x_{\max} - x_0)L}\|\overline{\mathbf{y}}_0 - \overline{\mathbf{z}}_0\|.$$

This estimation shows the stability of the scheme. ■

Convergence of the θ method

Thm. 91. Let us consider the initial value problem

$$\overline{\mathbf{y}}' = \mathbf{f}(x, \overline{\mathbf{y}}), \quad \overline{\mathbf{y}}(x_0) \text{ given}$$

($\overline{\mathbf{y}} \in C^3$). Then the θ method is convergent and

$$\|\overline{\mathbf{e}}_k\| \leq \frac{h}{4} \left(\left| \frac{1}{2} - \theta \right| M_2 + \frac{h}{3} M_3 \right) \left(e^{\frac{(b-a)L}{1-\theta L h}} - 1 \right),$$

where $M_3 = \max_{x \in [a,b]} \|\overline{\mathbf{y}}'''(x)\|$.

Rmk. The Crank–Nicolson method has second order, while the other methods are only first order convergent.

Runge–Kutta methods

Runge–Kutta methods

Carl David Tolmé Runge (1856 - 1927, German),
Martin Wilhelm Kutta (1867 - 1944, German)

Let us assume that $\mathbf{f}$ is sufficiently smooth. Then the solution $\overline{\mathbf{y}}$ will be also sufficiently smooth. Let us expand $\overline{\mathbf{y}}$ into Taylor series at the point x_0:

$$\overline{\mathbf{y}}(x_0+h) = \overline{\mathbf{y}}(x_0) + h \underbrace{\overline{\mathbf{y}}'(x_0)}_{=\mathbf{f}(x_0,\overline{\mathbf{y}}(x_0))} + \frac{h^2}{2}\underbrace{\overline{\mathbf{y}}''(x_0)}_{=?} + \dots.$$

$\overline{\mathbf{y}}''(x_0)$ can be calculated, but we need the derivatives of $\mathbf{f}$.

$$\overline{\mathbf{y}}''(x_0) = \mathbf{f}'_x(x_0, \overline{\mathbf{y}}(x_0)) + \mathbf{f}'_y(x_0, \overline{\mathbf{y}}(x_0))\mathbf{f}(x_0, y(x_0)).$$

Thus

$$\overline{\mathbf{y}}(x_0 + h) = \overline{\mathbf{y}}(x_0)$$
$$+h\left(\mathbf{f}(x_0, \overline{\mathbf{y}}(x_0)) + \frac{h}{2}\left(\mathbf{f}'_x(x_0, \overline{\mathbf{y}}(x_0)) + \mathbf{f}'_y(x_0, \overline{\mathbf{y}}(x_0))\mathbf{f}(x_0, \overline{\mathbf{y}}(x_0))\right)\right) + \dots$$

Let us search for a sufficiently accurate approximation of the highlighted factor in the form

$$a\mathbf{f}(x_0, \overline{\mathbf{y}}(x_0)) + b \underbrace{\mathbf{f}(x_0 + \alpha h, \overline{\mathbf{y}}(x_0) + \beta h \mathbf{f}(x_0, \overline{\mathbf{y}}(x_0)))}_{\mathbf{f}(x_0, \overline{\mathbf{y}}(x_0)) + \mathbf{f}'_x(x_0, \overline{\mathbf{y}}(x_0))\alpha h + \mathbf{f}'_y(x_0, \overline{\mathbf{y}}(x_0))\beta h \mathbf{f}(x_0, \overline{\mathbf{y}}(x_0)) + O(h^2)},$$

where a, b, α, β are suitable real constants.

Runge–Kutta methods

We obtain that

$$a + b = 1, \quad \alpha b = \beta b = \frac{1}{2},$$

and writing all parameters as functions of b we obtain

$$a = 1 - b, \quad \alpha = \beta = \frac{1}{2b}.$$

General form:

$$\overline{\mathbf{y}}_{k+1}$$
$$= \overline{\mathbf{y}}_k + h((1-b)\mathbf{f}(x_k, \overline{\mathbf{y}}_k) + b\mathbf{f}(x_k + h/(2b), \overline{\mathbf{y}}_k + \mathbf{f}(x_k, \overline{\mathbf{y}}_k)h/(2b))).$$

Rmk. The consistency order of these methods ($b \neq 0$) is 2. It can be proven that they are also stable. Thus these methods are convergent and the order of the convergence is 2.

Runge–Kutta methods

Rmk. Special cases:

Modified Euler method (RK2, $b = 1$):

$$\overline{\mathbf{y}}_{k+1} = \overline{\mathbf{y}}_k + h\mathbf{f}(x_k + h/2, \overline{\mathbf{y}}_k + \mathbf{f}(x_k, \overline{\mathbf{y}}_k)h/2).$$

Simplified Runge–Kutta or Heun method ($b = 1/2$):

$$\overline{\mathbf{y}}_{k+1} = \overline{\mathbf{y}}_k + h\left(\mathbf{f}(x_k, \overline{\mathbf{y}}_k)/2 + \mathbf{f}(x_k + h, \overline{\mathbf{y}}_k + \mathbf{f}(x_k, \overline{\mathbf{y}}_k)h)/2\right).$$

Runge–Kutta methods - general form

$$\overline{\mathbf{y}}_{k+1} = \overline{\mathbf{y}}_k + h\Phi(x_k, \overline{\mathbf{y}}_k, h),$$

where

$$\Phi(x, \overline{\mathbf{y}}, h) = \sum_{r=1}^{R} c_r k_r$$

and

$$k_1 = \mathbf{f}(x, \overline{\mathbf{y}}),$$

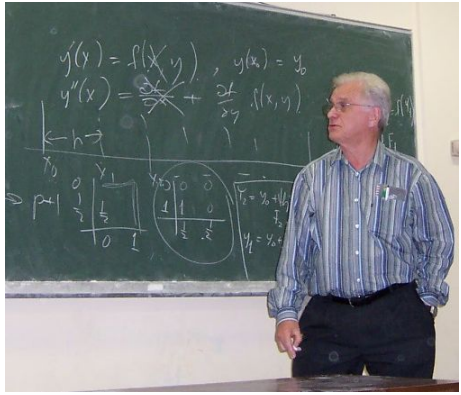
$$k_r = \mathbf{f}(x + ha_r, \overline{\mathbf{y}} + h\sum_{s=1}^{r-1} b_{rs}k_s), \quad r = 2, \ldots, R,$$

$$a_r = \sum_{s=1}^{r-1} b_{rs}, \quad r = 2, \ldots, R.$$

R is called the number of the stages of the method.

Runge–Kutta methods - Butcher's tableau

The coefficients can be conveniently written in a tabular form (so-called Butcher's tableau).



John C. Butcher (1933 -, New-Zealand)

Runge–Kutta methods - Butcher's tableau

$$
\begin{array}{c|cccc}
0 & & & & \\
a_2 & b_{21} & & & \\
a_3 & b_{31} & b_{32} & & \\
\vdots & & & & \\
a_R & b_{R1} & b_{R2} & b_{R,R-1} & \\
\hline
 & c_1 & c_2 & c_{R-1} & c_R
\end{array}
\qquad
\begin{array}{c|c}
\overline{\mathbf{a}} & \mathbf{B} \\
\hline
 & \overline{\mathbf{c}}^T
\end{array}
$$

The consistency order of the methods (the conditions are understood cumulatively):

cons. order	condition
1	$\overline{\mathbf{a}} = \mathbf{B}\overline{\mathbf{e}} \quad \overline{\mathbf{c}}^T\overline{\mathbf{e}} = 1$
2	$\overline{\mathbf{c}}^T\overline{\mathbf{a}} = 1/2$
3	$\overline{\mathbf{c}}^T(\overline{\mathbf{a}}^{\cdot 2}) = 1/3 \quad \overline{\mathbf{c}}^T\mathbf{B}\overline{\mathbf{a}} = 1/6$
4	$\overline{\mathbf{c}}^T(\overline{\mathbf{a}}^{\cdot 3}) = 1/4 \quad \overline{\mathbf{c}}^T\mathsf{diag}(\overline{\mathbf{a}})\mathbf{B}\overline{\mathbf{a}} = 1/8$ $\overline{\mathbf{c}}^T\mathbf{B}(\mathbf{a}^{\cdot 2}) = 1/12 \quad \overline{\mathbf{c}}^T\mathbf{B}^2\overline{\mathbf{a}} = 1/24$

Runge–Kutta methods - RK2, Heun, RK4 methods

Example. Modified Euler (RK2) and Heun methods (two-stage methods):

$$\begin{array}{c|cc} 0 & & \\ 1/2 & 1/2 & \\ \hline & 0 & 1 \end{array} \qquad \begin{array}{c|cc} 0 & & \\ 1 & 1 & \\ \hline & 1/2 & 1/2 \end{array}$$

Rmk. The achievable highest order with fixed number of stages:

number of stages (m)	$1, 2, 3, 4$	$5, 6, 7$	$8, 9, 10$
max. order	m	$m-1$	$m-2$

Runge–Kutta methods - RK2, Heun, RK4 methods

Example. Fourth order (four-stage) Runge–Kutta method (RK4):

$$\begin{array}{c|cccc} 0 & & & & \\ 1/2 & 1/2 & & & \\ 1/2 & 0 & 1/2 & & \\ 1 & 0 & 0 & 1 & \\ \hline & 1/6 & 1/3 & 1/3 & 1/6 \end{array}$$

$$\overline{\mathbf{y}}_{k+1} = \overline{\mathbf{y}}_k + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4),$$

$$\begin{aligned} k_1 &= \mathbf{f}(x_k, \overline{\mathbf{y}}_k) \\ k_2 &= \mathbf{f}(x_k + h/2, \overline{\mathbf{y}}_k + k_1 h/2) \\ k_3 &= \mathbf{f}(x_k + h/2, \overline{\mathbf{y}}_k + k_2 h/2) \\ k_4 &= \mathbf{f}(x_k + h, \overline{\mathbf{y}}_k + k_3 h) \end{aligned}$$

Absolute stability

The test problem

Let us applied the studied methods to the initial value problem

$$y' = \lambda y, \quad y(0) = 1,$$

where $\lambda < 0$ is an arbitrary negative real number.

The solution is $y(x) = e^{\lambda x}$, which converges to 0 as $x \to \infty$.

Def. 92. If a numerical method with a fixed step size h is applied to the test problem and the numerical solution $|y_k|$ tends to 0 as $k \to \infty$ then the method is called absolute stable.

Naturally, the absolute stability depends on both λ and h.

Def. 93. The set $\mathcal{A} = \{z = h\lambda \in \mathbb{R} \,|\, \text{the method is absolute stable with } z\ \}$ is called the domain of absolute stability. If $\mathbb{R}^- \subset \mathcal{A}$, then the method is called to be A-stable.

Absolute stability of the EE and IE methods

EE method:

$$y_k = (1 + h\lambda)^k,$$

which tends to zero only if $|1 + h\lambda| < 1$, that is if $z = h\lambda$ lies in a circle with radius 1 and with center at -1.

The method is absolute stable iff $h < -2/\lambda$.

IE method:

$$y_k = \frac{1}{(1 - h\lambda)^k},$$

which tends to zero only if $z = h\lambda$ lies outside the circle with radius 1 and with center at 1.

The method is A-stable.

Rmk. None of the (explicit) Runge–Kutta methods are A-stable.

Solution of stiff equations

Solution of stiff equations

The high stability of the equations results in instability in the numerical solution.

Equations for which implicit methods work well and explicit methods behave badly.

Equations for which the choice of h is restricted not by the accuracy but by the absolute stability.

Example. The efficient solution of the van der Pol equation ($\mu = 100000$):

$$y_1' = y_2$$
$$y_2' = \mu(1 - y_1^2)y_2 - y_1;$$

Example. The solution of the equation $y' = -15y + 1$, $y(0) = 0$.

Predictor-corrector methods

Multistep methods

Predictor-corrector methods

A simple example

CN method:

$$\overline{\mathbf{y}}_{k+1} = \overline{\mathbf{y}}_k + \frac{h}{2}(\mathbf{f}(x_k, \overline{\mathbf{y}}_k) + \mathbf{f}(x_{k+1}, \overline{\mathbf{y}}_{k+1})).$$

This method is an implicit one. If $h \leq 2/L$ (L is the Lipschitz constant), then the equation has a unique solution for $\overline{\mathbf{y}}_{k+1}$. $\overline{\mathbf{y}}_{k+1}$ can be computed by fixed point iteration:

$$\overline{\mathbf{y}}_{k+1}^{(s+1)} = \overline{\mathbf{y}}_k + \frac{h}{2}(\mathbf{f}(x_k, \overline{\mathbf{y}}_k) + \mathbf{f}(x_{k+1}, \overline{\mathbf{y}}_{k+1}^{(s)})).$$

Problems:

- When to stop the iteration?
- $\mathbf{f}(x, \overline{\mathbf{y}})$ must be computed many times.
- What is a good choice for $\overline{\mathbf{y}}_{k+1}^{(0)}$?

A simple example

Solution: Let us apply an explicit method to obtain a good guess for $\overline{\mathbf{y}}_{k+1}^{(0)}$.

For example, we can use the explicit Euler method. That is we set

$$\overline{\mathbf{y}}_{k+1}^{(0)} = \overline{\mathbf{y}}_k + h\mathbf{f}(x_k, \overline{\mathbf{y}}_k).$$

Iterating only once we obtain the method

$$\overline{\mathbf{y}}_{k+1} = \overline{\mathbf{y}}_k + \frac{h}{2}(\mathbf{f}(x_k, \overline{\mathbf{y}}_k) + \mathbf{f}(x_{k+1}, \overline{\mathbf{y}}_k + h\mathbf{f}(x_k, \overline{\mathbf{y}}_k)),$$

which is an explicit method.

Advantage of this technique: What is the order of this method? The EE method is first order, the CN method is second order, but the combined method above is second order. This is the Heun method ($b = 1/2$), which is second order indeed.

The idea of predictor-corrector methods

The application of an explicit and an implicit method after each other.

- **Predictor:** An explicit method that predicts a good starting value for the iteration in the case of an implicit method.
- **Corrector:** The applied implicit method, with which we correct the value of $\overline{\mathbf{y}}_{k+1}$.

Multistep methods

General form of s-step methods

$$a_s y_{k+1} + a_{s-1} y_k + \ldots + a_0 y_{k-(s-1)}$$
$$= h(b_s \underbrace{f_{k+1}}_{f(x_{k+1},y_{k+1})} + b_{s-1} \underbrace{f_k}_{f(x_k,y_k)} + \ldots + b_0 \underbrace{f_{k-(s-1)}}_{f(x_{k-(s-1)},y_{k-(s-1)})})$$

- $a_s \neq 0$, because it is used to calculate y_{k+1}.
- If $b_s = 0$, then the method is explicit, otherwise it is implicit.
- To start the method we need the values $y_0, \ldots, y_{s-1}$. These can be calculated with a sufficiently accurate one-step method (e.g. with some RK methods).

Adams methods

If $a_s = 1$, $a_{s-1} = -1$ and $a_k = 0$ $(k = s-2, \ldots, 0)$, then the method is called Adams method. The explicit Adams methods are called Adams-Bashforth methods (John Couch Adams (1819 - 1892, English), astronomer, mathematician; Francis Bashforth (1819 - 1912, English), mathematician), and the implicit ones Adams-Moulton methods (Forest Ray Moulton (1872 - 1952, USA), astronomer).
Construction:

$$\int_{x_k}^{x_{k+1}} y'(x)\,\mathrm{d}x = \int_{x_k}^{x_{k+1}} f(x, y(x))\,\mathrm{d}x$$

$$y_{k+1} - y_k = \int_{x_k}^{x_{k+1}} \sum_{j=k-s+1}^{k(AB),\ k+1(AM)} \underbrace{f(x_j, y_j)}_{f_j} l_j(x)\,\mathrm{d}x,$$

$$= \sum_{j=k-s+1}^{k(AB),\ k+1(AM)} f_j \int_{x_k}^{x_{k+1}} l_j(x)\,\mathrm{d}x$$

where l_j $(j = k-s+1, \ldots, k(AB), k+1(AM))$ is the jth characteristic Lagrange polynomial to the points $x_{k-s+1}, \ldots, x_k(AB), x_{k+1}(AM)$.

Adams methods

The maximal order Adams–Bashforth formulas:

Steps	Formula	Order
1	$y_{k+1} = y_k + hf_k$ (EE)	1
2	$y_{k+1} = y_k + \frac{h}{2}(3f_k - f_{k-1})$	2
3	$y_{k+1} = y_k + \frac{h}{12}(23f_k - 16f_{k-1} + 5f_{k-2})$	3
4	$y_{k+1} = y_k + \frac{h}{24}(55f_k - 59f_{k-1} + 37f_{k-2} - 9f_{k-3})$	4
5	$y_{k+1} = y_k + \frac{h}{720}(1901f_k - 2774f_{k-1} + 2616f_{k-2} - 1274f_{k-3} + 251f_{k-4})$	5

The maximal order Adams–Moulton formulas:

Steps	Formula	Order
1	$y_{k+1} = y_k + hf_{k+1}$ (IE)	1
1	$y_{k+1} = y_k + \frac{h}{2}(f_{k+1} + f_k)$ (CN)	2
2	$y_{k+1} = y_k + \frac{h}{12}(5f_{k+1} + 8f_k - f_{k-1})$	3
3	$y_{k+1} = y_k + \frac{h}{24}(9f_{k+1} + 19f_k - 5f_{k-1} + f_{k-2})$	4
4	$y_{k+1} = y_k + \frac{h}{720}(251f_{k+1} + 646f_k - 264f_{k-1} + 106f_{k-2} - 19f_{k-3})$	5

Backward differentiation formulas (BDF)

If $b_s = 1$ and $b_k = 0$ $(k = s-1, \ldots, 0)$, then the method is called backward differentiation formula - BDF.

Construction:

We start with the differential equation at the point x_{k+1}

$$y'(x_{k+1}) = f(x_{k+1}, y(x_{k+1})).$$

The right hand side is approximated by $f(x_{k+1}, y_{k+1}) = f_{k+1}$, and on the left hand side we apply a backward difference formula.

The maximal order BDF methods:

Steps	Formula	Order
1	(IE) $y_{k+1} - y_k = hf_{k+1}$	1
2	$\frac{3}{2}y_{k+1} - 2y_k + \frac{1}{2}y_{k-1} = hf_{k+1}$	2
3	$\frac{11}{6}y_{k+1} - 3y_k + \frac{3}{2}y_{k-1} - \frac{1}{3}y_{k-2} = hf_{k+1}$	3
4	$\frac{25}{12}y_{k+1} - 4y_k + 3y_{k-1} - \frac{4}{3}y_{k-2} + \frac{1}{4}y_{k-3} = hf_{k+1}$	4
5	$\frac{137}{60}y_{k+1} - 5y_k + 5y_{k-1} - \frac{10}{3}y_{k-2} + \frac{5}{4}y_{k-3} - \frac{1}{5}y_{k-4} = hf_{k+1}$	5

Consistency

We calculate h·LTE (let we develop the Taylor expansion at $z = x_{k-s+1}$):

$$\begin{aligned}
h \cdot LTE =& a_s y(x_{k+1}) + \ldots + a_0 y(x_{k-s+1}) \\
& - h(b_s f(x_{k+1}, y(x_{k+1})) + \ldots + b_0 f(x_{k-s+1}, y(x_{k-s+1}))) = \\
=& \sum_{i=0}^{s} \left(a_i y(z+ih) - hb_i \underbrace{f(z+ih, y(z+ih))}_{y'(z+ih)} \right) \\
=& \sum_{i=0}^{s} a_i (y(z) + y'(z) ih + y''(z)(ih)^2/2 + \ldots) \\
& - h \sum_{i=0}^{s} b_i (y'(z) + y''(z) ih + y'''(z)(ih)^2/2 + \ldots) \\
=& d_0 y(z) + d_1 y'(z) h + d_2 y''(z) h^2 + \ldots,
\end{aligned}$$

Consistency

where

$$d_0 = \sum_{i=0}^{s} a_i$$

$$d_1 = \sum_{i=0}^{s} (i a_i - b_i)$$

$$\vdots$$

$$d_j = \sum_{i=0}^{s} \left(\frac{i^j a_i}{j!} - \frac{i^{j-1} b_i}{(j-1)!} \right)$$

$$\vdots$$

Thus we have

$$\text{LTE} = d_0 y(z) \frac{1}{h} + d_1 y'(z) + d_2 y''(z) h + d_3 y'''(z) h^2 + \dots.$$

Consistency

From the form of the local truncation error it follows the following result directly.

Thm. 94. The multistep method is consistent iff $d_0 = d_1 = 0$. If the solution y is in C^{m+1} and

$$d_0 = \ldots = d_m = 0 \ (m \geq 1)$$

and

$$d_{m+1} \neq 0,$$

then the local truncation error is $O(h^m)$, thus the consistency order of the method is m.

Example. The AB5 and AM4 methods have consistency order 5.

Example. The method $y_{k+1} - y_{k-1} = \frac{h}{3}(f_{k+1} + 4f_k + f_{k-1})$ has consistency order 4. $a_2 = 1, a_1 = 0, a_0 = -1$, $b_2 = 1/3, b_1 = 4/3, b_0 = 1/3$. Thus $d_0 = \ldots = d_4 = 0$ és $d_5 = -1/90$.

Consistency

What is the maximal achievable consistency order?

The method has $2s+1$ free coefficients (because the coefficients are unique only up to a nonzero constant multiplier). There is some hope that we can choose the coefficients in such a way that $d_0 = \ldots = d_{2s} = 0$ ($2s+1$ equations and $2s+1$ unknowns).

Theorem

(Dahlquist (1956)) The system of equations $d_0 = \ldots = d_{2s} = 0$ has always a solution up to a nonzero constant multiplier. Thus, with an s-step method, we can achieve a consistency order as high as $2s$. (For explicit methods, the achievable highest order is $2s-1$ (b_s must be zero). For AB methods: s, and for AM methods: $s+1$.)

Germund Dahlquist, 1925-2005, Swedish

Stability

Def. 96. An s-step method is called to be (zero)stable if there are two constants $K > 0$ and $h_0 > 0$ independent of the step size such that for $0 < h < h_0$ we have

$$|y_k - \hat{y}_k| \leq K \max\{|y_0 - \hat{y}_0|, \ldots, |y_{s-1} - \hat{y}_{s-1}|\}, \ k = s, \ldots, N_h,$$

that is starting the scheme from two different sets of initial values, the difference of the solutions remains bounded on finite intervals. ($\hat{y}_k$ is the sequence produced with the hatted values.)

Thm. 97. An s-step method is stable iff all zeros of the so-called first characteristic polynomial $\zeta(z) = a_s z^s + \ldots + a_1 z + a_0$ lie in the closed complex unit circle centered at the origin and the zeros on the boundary are single.

Thm. 98. (Equivalence theorem) Let us suppose that the solution of the initial value problem is in C^{r+1}, moreover, let us suppose that the multistep method has consistency order r. Then the stability is a necessary and sufficient condition of the convergence. The order of the convergence is r.

Example.
EE, IE: $\zeta(z) = z - 1$, thus the methods are stable, they are also consistent (order is 1), and these imply that they are convergent with order 1.

Theorem
The Adams methods are convergent and their convergence order equals the order of their consistency.

Proof: $\zeta(z) = z^s - z^{s-1} = z^{s-1}(z-1)$. Thus, the method is always stable. The other part of the theorem follows from the equivalence theorem. ■

Stability

Theorem

There are valid the following so-called Dahlquist's (first and second) barriers (indicated by blue and extended by some previously discussed results).

s: number of steps of the method	Impicit	Explicit
The greatest possible consistency order	$2s$	$2s-1$
The greatest possible consistency order of a stable method	$s+1$ (s odd) $s+2$ (s even)	s
The greatest possible order of an A-stable method	2	-
The greatest possible order of a convergent Adams method	$s+1$ (AM)	s (AB)

A not stable method

$$y_{n+1} + 4y_n - 5y_{n-1} = h(4f_n + 2f_{n-1})$$

This 2-step method is explicit and third order, thus it cannot be stable. This can be verified on the test equation $y' = 0$, $y(0) = 0$.

Then for $y_0 = 0$ and $y_1 = \varepsilon_h$ we have

$$y_n = (1 - (-5)^n)\varepsilon_h/6.$$

The numerical solution at $x = 1$ is $(n = 1/h)$

$$(1 - (-5)^{1/h})\varepsilon_h/6.$$

With the choice $y_0 = 0$ and $y_1 = 0$ we obtain zero. This shows that the method cannot be stable.

Solution of boundary value problems

Solution of boundary value problems

Initial value problems: The values of all the unknown functions are known at the same fixed point.

Boundary value problems: The values of the unknown functions are known at more different points (generally at the two ends of an interval).

Example. The equation of the deflection of a rod:

$$EIy''(x) + P\cos(y(x)) = 0,\ \ y(0) = 0, y(L) = 0.$$

$$\Downarrow$$

$$y_1' = y_2,\ \ y_1(0) = y_1(L) = 0,$$
$$y_2' = -\frac{P}{EI}\cos(y_1).$$

Boundary value problems

Let us consider the two-point boundary value problems in the form

$$y'' = f(x, y, y'), \quad y(a) = A,\ y(b) = B,$$

where $a < b$ and $x \in [a, b]$.

Theorem

Assume that f is continuous, the derivative with respect to the second argument is continuous and positive, the derivative with respect to the third argument is continuous and bounded. Then the boundary value problem has a unique solution.

Example. The problem $y'' = -y$, $y(0) = 3$, $y(\pi) = 7$ has no solution.

Shooting method

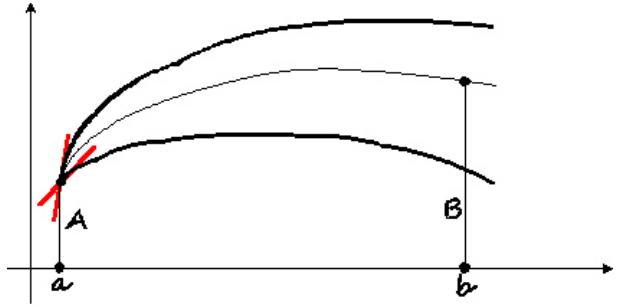
Shooting (garden hose) method

Let us rewrite the problem to an initial values problem

$$y_1' = y_2, \quad y_1(a) = y(a) = A,$$
$$y_2' = f(x, y_1, y_2), \quad y_2(a) = y'(a) =: D,$$

where we have replaced the unknown value $y_2(a) = y'(a)$ by a fixed real number D.

Let us denote the solution of the above problem by $y(x; D)$. If $y(b; D) = B$, then $y(x; D)$ solves the original boundary value problem. Otherwise, we choose another value D. This can be done in a systematic way.



Shooting method

We have to solve the nonlinear equation

$$y(b; D) - B = 0$$

for the parameter D.

We can use the previously studied methods to find the appropriate D.

- Bisection method,
- Newton's method.

Finite difference method

Finite difference method (matrix method)

$$y'' = f(x, y, y'),\ y(a) = A,\ y(b) = B$$

Let us define an equidistant mesh on $[a, b]$. Let the length of the subintervals be $h = (b - a)/(n + 1)$, thus $x_i = a + ih$ $(i = 0, \ldots, n + 1)$.

Let y_i denote $(i = 0, \ldots, n + 1)$ the approximations of the exact solution at x_i. Let us replace the derivatives of the solution to finite difference approximations:

$$\frac{y_{i-1} - 2y_i + y_{i+1}}{h^2} = f\left(x_i, y_i, \frac{y_{i+1} - y_{i-1}}{2h}\right),$$

moreover let $y_0 = A$ and $y_{n+1} = B$. If f is nonlinear, then the solution is difficult. We must use one of the solvers for nonlinear systems of equations (Newton's method, fixed point iteration).

Finite difference method

Let us investigate only linear equations, that is the boundary value problems in the form:

$$y''(x) = u(x) + v(x)y + w(x)y',\ y(a) = A,\ y(b) = B$$

Then the finite difference method results in the problem ($u(x_i) = u_i$, $v(x_i) = v_i$, $w(x_i) = w_i$):

$$\frac{y_{i-1} - 2y_i + y_{i+1}}{h^2} = u_i + v_i y_i + w_i \frac{y_{i+1} - y_{i-1}}{2h}, \tag{3}$$

moreover $y_0 = A$ and $y_{n+1} = B$. After rearrangement we obtain

$$y_0 = A,$$

$$\overbrace{\left(\frac{1}{h^2} + \frac{w_i}{2h}\right)}^{a_i} y_{i-1} \overbrace{- \left(\frac{2}{h^2} + v_i\right)}^{b_i} y_i + \overbrace{\left(\frac{1}{h^2} - \frac{w_i}{2h}\right)}^{c_i} y_{i+1} = u_i$$

$$y_{n+1} = B,$$

Finite difference method

In order to obtain the approximations y_i, we have to solve the linear system:

$$\begin{bmatrix} b_1 & c_1 & & & \\ a_2 & b_2 & c_2 & & \\ & \ddots & \ddots & \ddots & \\ & & a_{n-1} & b_{n-1} & c_{n-1} \\ & & & a_n & b_n \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_{n-1} \\ y_n \end{bmatrix} = \begin{bmatrix} u_1 - a_1 A \\ u_2 \\ \vdots \\ u_{n-1} \\ u_n - c_n B \end{bmatrix}.$$

Assume that $\inf v > 0$ and $w \not\equiv 0$ is a bounded function on $[a, b]$. Moreover, let us assume that the step size h is sufficiently small, that is $h \leq 2/\sup_{x\in[a,b]}\{|w(x)|\}$.

Then the matrix is strictly diagonally dominant, that implies that the system can be solved using the Gaussian elimination.

Convergence

Def. 102. The previous scheme for the solution of the boundary value problem is convergent if $\max_{i=1,\dots,n} |y_i - y(x_i)| = O(h^r)$ $(r \geq 0)$ provided that $h \to 0$ $(n \to \infty)$, moreover r is the order of the convergence.

Theorem

If $y \in C^4$ then the investigated scheme is convergent with convergence order 2.

Proof. Let us compute first the LTE at the point x_i:

$$\tau_i = \frac{y(x_{i-1}) - 2y(x_i) + y(x_{i+1})}{h^2} - u_i - v_i y(x_i) - w_i \frac{y(x_{i+1}) - y(x_{i-1})}{2h}$$

$$= \frac{h^2}{12} y''''(\xi) - w_i \frac{h^2}{6} y'''(\eta).$$

That is with a positive constant C we have

$$|\tau_i| \leq h^2 M_3 C.$$

Thus the method is consistent and the order of the consistency is 2.

Convergence

Let us subtract the scheme (3) from the inequility obtained for the LTE. Let us intruduce the notation $e_i = y(x_i) - y_i$ for the error at the point x_i. We obtain the linear system of equations:

$$\frac{e_{i-1} - 2e_i + e_{i+1}}{h^2} - v_i e_i - w_i \frac{e_{i+1} - e_{i-1}}{2h} = \tau_i,$$

that is componentwisely

$$\begin{bmatrix} b_1 & c_1 & & & \\ a_2 & b_2 & c_2 & & \\ & \ddots & \ddots & \ddots & \\ & & a_{n-1} & b_{n-1} & c_{n-1} \\ & & & a_n & b_n \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_{n-1} \\ e_n \end{bmatrix} = \begin{bmatrix} \tau_1 \\ \tau_2 \\ \vdots \\ \tau_{n-1} \\ \tau_n \end{bmatrix}.$$

Convergence

The matrix of the system is the -1 multiple of an M-matrix. The main diagonal is negative, the other elements are nonnegative, and the main diagonal is strictly dominant. We can apply the estimation for the inverses of M-matrices. Together with the expression for the LTE τ_i, we obtain that

$$\left\| \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_{n-1} \\ e_n \end{bmatrix} \right\|_\infty \leq \frac{1}{\inf_{x\in[a,b]} v(x)} \left\| \begin{bmatrix} \tau_1 \\ \tau_2 \\ \vdots \\ \tau_{n-1} \\ \tau_n \end{bmatrix} \right\|_\infty \leq \frac{M_3 h^2 C}{\inf_{x\in[a,b]} v(x)}.$$

This shows second order convergence. ■

The end

Summary of some main concepts

- Normed spaces (norms, normed spaces, equivalence of norms, Banach spaces, Banach fixed point theorem)
- Vector and matrix norms
- Euclidean spaces (scalar product, euclidean space, orthogonality, Gram–Schmidt orthogonalization, orthogonal polynomials)
- Special properties of matrices
- Eigenvalues and eigenvectors of matrices
- Diagonalizability of matrices

Normed spaces

Vector space (linear space)

Def. 104. A set $V \neq \emptyset$ is called (real) vector space, if an addition and a multiplication with scalar operation is defined on it with the properties:

1. $x + y = y + x$, $\forall a, b \in V$;
2. $(x + y) + z = x + (y + z)$, $\forall x, y, z \in V$;
3. $\exists o \in V$, $x + o = x$, $\forall x \in V$;
4. $\forall x \in V$, $\exists \hat{x} \in V$, $x + \hat{x} = o$;
5. $1 \cdot x = x$, $\forall x \in V$;
6. $\alpha(x + y) = \alpha x + \alpha y$, $\forall x, y \in V$, $\forall \alpha \in \mathbb{R}$;
7. $(\alpha + \beta)x = \alpha x + \beta x$, $\forall x \in V$, $\forall \alpha, \beta \in \mathbb{R}$;
8. $\alpha(\beta x) = (\alpha\beta)x$, $\forall x \in V$, $\forall \alpha, \beta \in \mathbb{R}$.

Ex.: Vectors on the plane and in space, $\mathbb{R}^n$, $\mathbb{R}^{m \times n}$, $C[a, b]$, P_n etc. with the usual operations.

Special vector systems in vector spaces

Def. 105. A vector $x \in V$ is called the linear combination of the vectors $x_1, \dots, x_k \in V$, if $\exists\ \alpha_1, \dots \alpha_k \in \mathbb{R}$ such that $x = \alpha_1 x_1 + \dots + \alpha_k x_k$.
If $W \subset V$ then we denote
$Lin(W) := \{x \in V \,|\, x$ is the linear combination of the vectors in $W\}$

Def. 106. The vectors $x_1, \dots, x_k \in V$ $(k \in \mathbb{N})$ are called lin. independent if $\alpha_1 x_1 + \dots + \alpha_k x_k = o \Rightarrow \alpha_i = 0$ $(i = 1, \dots, k)$. If we have infinitely many vectors, then we require the above property for all finite subset. ($\leftrightarrow$ lin. dependent)

Def. 107. The vector system $\mathcal{B} \subset V$ is called the basis of V if it is linearly independent and $Lin(\mathcal{B}) = V$.

If V possesses a bases with finitely many elements, then V is called finite dimensional vector space. In finite dimensional vector spaces the number of elements in each basis are equal. This is the dimension of the vector space.

Normed spaces

Def. 108. The pair $(V, \|.\|)$ is called normed space if V is a vector space and $\|.\| : V \to \mathbb{R}$ is a given function (so-called norm) with the properties:

1. $\|x\| = 0 \Leftrightarrow x = o$;
2. $\|\alpha x\| = |\alpha| \cdot \|x\|$, $\forall x \in V, \forall \alpha \in \mathbb{R}$;
3. $\|x + y\| \leq \|x\| + \|y\|$, $\forall x, y \in V$.

Ex.

- Vectors on the plane and in the space, $\|\vec{v}\| =$ is the usual length of the vectors.
- $\mathbb{R}^n$, $\mathbf{x} = [x_1, \ldots, x_n]^T$:
  $\|\mathbf{x}\|_1 = |x_1| + \cdots + |x_n|$,
  $\|\mathbf{x}\|_2 = \sqrt{x_1^2 + \cdots + x_n^2}$,
  $\|\mathbf{x}\|_\infty = \max\{|x_1|, \ldots, |x_n|\}$.
- $C[a, b]$, f
  $\|f\|_{C[a,b]} = \max_{x \in [a,b]}\{|f(x)|\}$
- $\mathbb{R}^{m \times n}$, $\mathbf{A} = [a_{ij}] \in \mathbb{R}^{m \times n}$
  $\|\mathbf{A}\| = \max_{i=1:m, j=1:n}\{|a_{ij}|\}$ (see later).

Convergence in normed spaces, $V = (V, \|.\|)$

Def. 109. The distance of the elements $x, y \in V$ is the value $\|x - y\|$.

Thm. 110.

- $\|x - y\| \geq 0$, $\forall x, y \in V$, $\|x - y\| = 0 \Leftrightarrow x = y$,
- $\|x - y\| = \|y - x\|$, $\forall x, y \in V$,
- $\|x - y\| \leq \|x - z\| + \|z - y\|$, $\forall x, y, z \in V$.

Def. 111. We say that the sequence $\{x_k\} \subset V$ tends to the element $x \in V$ if the real number sequence $\{\|x_k - x\|\}$ tends to zero. Notation: $x_k \to x$.

Def. 112. The norms $\|.\|_1$ és $\|.\|_2$ defined on the same vector space are called equivalent if $\exists c_1, c_2 > 0$ such that

$$c_1\|x\|_1 \leq \|x\|_2 \leq c_2\|x\|_1, \ \forall x \in V.$$

Convergence in normed spaces, $V = (V, \|.\|)$

Rmk. Equivalent norms define the same convergence. In finite dimensional vector spaces all norms are equivalent.

Def. 113. We say that the sequence $\{x_k\} \subset V$ is a Cauchy sequence if $\forall \varepsilon > 0$, $\exists M \in \mathbb{N}$, $\forall n, m \geq M$ $\|x_n - x_m\| < \varepsilon$.

Thm. 114. All convergent sequences in V are Cauchy sequences.

Rmk. The converse of the theorem is not true.

Def. 115. We say that the normed space $(V, \|.\|)$ is a Banach space if all Cauchy sequences in V are convergent.

Example. The examples listed for normed spaces are examples also for Banach spaces.

Banach fixed point theorem

Thm. 116. Let $(V, \|.\|)$ be a Banach space and $\emptyset \neq H \subset (V, \|.\|)$ a closed subset ($\{x_k\} \subset H$, $x_k \to x$ implies $x \in H$). Let $F : H \to H$ be a contraction ($\exists\ 0 \leq q < 1$, $\|F(x) - F(y)\| \leq q\|x - y\|$, $\forall\ x, y \in H$).

- Then F possesses one and only one fixed point in H, that is an element $x^\star \in H$ such that $F(x^\star) = x^\star$.
- With arbitrary initial element $x_0 \in H$, the sequence produced with the iteration $x_{k+1} = F(x_k)$ tends to $x^\star$.
- It is valid the estimation

$$\|x^\star - x_m\| \leq \frac{q^m}{1-q}\|x_1 - x_0\|. \tag{4}$$

Euclidean spaces

Euclidean spaces

Def. 117. The pair $(V, \langle ., . \rangle)$ is called euclidean space if V is a vector space and $\langle ., . \rangle : (V \times V) \to \mathbb{R}$ is a so-called scalar product with the properties:

1. $\langle x, y \rangle = \langle y, x \rangle$ for all $x, y \in V$,
2. $\langle \alpha x, y \rangle = \alpha \langle x, y \rangle$, for all $x, y \in V,\ \alpha \in \mathbb{R}$,
3. $\langle x + y, z \rangle = \langle x, z \rangle + \langle y, z \rangle$, for all $x, y, z \in V$,
4. $\langle x, x \rangle > 0$, for all $o \neq x \in V$.

Two important examples

- In the space of the column vectors $\mathbb{R}^n$: with the notations $\overline{\mathbf{x}} = [x_1, \ldots, x_n]^T$ and $\overline{\mathbf{y}} = [y_1, \ldots, y_n]^T$, the assignment $\langle \overline{\mathbf{x}}, \overline{\mathbf{y}} \rangle = x_1 y_1 + \ldots + x_n y_n$ defines a scalar product $(\overline{\mathbf{x}}^T \overline{\mathbf{y}})$.
- In the vector space $C[a, b]$, the assignment

$$\langle f, g \rangle = \int_a^b s(x) f(x) g(x)\, \mathrm{d}x$$

defines a scalar product for all positive weight function $s \in C[a, b]$.

Euclidean spaces

Thm. 118. In a euclidean space $(V, \langle ., . \rangle)$, the assignment $\|x\| = \sqrt{\langle x, x \rangle}$ defines a norm (norm induced be the scalar product).

Def. 119.

- $x, y \in V$ orthogonal if $\langle x, y \rangle = 0$,
- $x_1, x_2, \ldots \in V$ orthogonal vector system if the vectors are pairwise orthogonal,
- $x \in V$ is normed if $\|x\| = 1$ is fulfilled in the norm induced by the scalar product.
- $x_1, x_2, \ldots \in V$ is an orthonormal vector system if the vectors are pairwise orthogonal and each vector is normed.

Gram–Schmidt orthogonalization

Thm. 120. Let $x_1, \dots, x_k$ be a linearly independent vector system in a euclidean space. Then we can set an orthonormal vector system $q_1, \dots, q_k$ with the properties $\operatorname{lin}(q_1, q_2, \dots, q_l) = \operatorname{lin}(x_1, x_2, \dots, x_l)$ for all $l = 1, \dots, k$.

Rmk. The polynomials p, q are called orthogonal on the interval $[a, b]$ with respect to the positive weight function s if

$$\int_a^b s(x)p(x)q(x)\,\mathrm{d}x = 0.$$

Def. 121. Let us consider the polynomials $1, x, x^2$ on the interval $[-1, 1]$. Then the polynomials obtained with the Gram–Schmidt orthogonalization using the weight function $s(x) \equiv 1$ in the scalar product are called Legendre polynomials, while with the weight function $s(x) = 1/\sqrt{1-x^2}$ we obtain the so-called Chebyshev polynomials.

Orthogonal polynomials

Degree	Legendre	Chebyshev
0	1	1
1	x	x
2	$(3x^2-1)/2$	$2x^2-1$
3	$(5x^3-3x)/2$	$4x^3-3x$
4	$(35x^4-30x^2+3)/8$	$8x^4-8x^2+1$

$T_0 = 1$, $T_1 = x$
Chebyshev: $T_{k+1} = 2xT_k - T_{k-1}$.
Legendre: $(k+1)T_{k+1} = (2k+1)xT_k - kT_{k-1}$.

Orthogonal polynomials

Thm. 122. Let us suppose that the polynomials $p_0, p_1, \ldots$ (subscripts denote the degrees) are pairwise orthogonal on the interval $[a, b]$ with respect to the positive weight function s. Then all roots of the polynomial are real, single and located in the interval $[a, b]$.

Proof. Let us consider the polynomial p_l and denote the distinct real roots from $[a, b]$ with odd multiplicity by $z_1, \ldots, z_k$. If $k = l$, then the statement is true, if $k < l$, then let us consider the polynomial $p(x) = (x - z_1) \ldots (x - z_k)$ ($p \equiv 1$ if $k = 0$), which has degree k. The polynomial $p_l \cdot p$ has degree $(l + k)$ and it does not change sign in the interval $[a, b]$. Thus the condition

$$\int_a^b p_l(x)p(x)s(x)\,\mathrm{d}x = 0$$

cannot hold. This completes the proof. ■

Special properties of matrices

Special matrices

- Band matrix: $\exists p, q \in \mathbb{N}$, $a_{i,j} = 0$ if $j < i - p$ or $i < j - q$. $1 + p + q$ is the so-called bandwidth.
- Diagonal matrix: offdiagonal elements are zero ($p = 0, q = 0$), $\mathbf{I}$ identity matrix.
- Upper triangular matrix: elements "below" the diagonal are zero ($p = 0$).
- Lower triangular matrix: elements "above" the diagonal are zero ($q = 0$).
- Upper Hessenberg matrix: elements "below" the subdiagonal are zero ($p = 1$).
- Lower Hessenberg matrix: elements "above" the superdiagonal are zero ($q = 1$).

Special matrices

- Tridiagonal matrix: all elements outside the main, sub- and superdiagonals are zero. $(p = q = 1)$.
- Symmetric matrix: $\mathbf{A}^T = \mathbf{A}$
- Skew-symmetric matrix: $\mathbf{A}^T = -\mathbf{A}$
- The vectors $\overline{\mathbf{x}}$ and $\overline{\mathbf{y}} \in \mathbb{R}^n$ are called orthogonal if $\overline{\mathbf{x}}^T\overline{\mathbf{y}} = 0$. Moreover, we trivially have $\overline{\mathbf{y}}^T\overline{\mathbf{x}} = 0$. If $\overline{\mathbf{x}}$ and $\overline{\mathbf{y}}$ are orthogonal, then $\|\overline{\mathbf{x}} + \overline{\mathbf{y}}\|_2^2 = \|\overline{\mathbf{x}}\|_2^2 + \|\overline{\mathbf{y}}\|_2^2$ (Pythagorean theorem).

  Orthogonal matrix: $\mathbf{A}\mathbf{A}^T = \mathbf{A}^T\mathbf{A} = \mathbf{I}$

  $(\|\mathbf{A}\mathbf{x}\|_2^2 = \mathbf{x}^T\mathbf{A}^T\mathbf{A}\mathbf{x} = \|\mathbf{x}\|_2^2,\ \|\mathbf{A}\|_2 = 1,\ \|\mathbf{A}\mathbf{B}\|_2 = \|\mathbf{B}\|_2)$

Special matrices

- $\mathbf{P}$ is a permutation matrix if, with the notation $\overline{\mathbf{e}}_k = [0, \ldots, 0, \overbrace{1}^{k\text{-adik}}, 0, \ldots, 0]^T$ $(k = 1, \ldots, n)$, $\mathbf{P} = [\overline{\mathbf{e}}_{i_1}, \ldots, \overline{\mathbf{e}}_{i_n}]$, where $i_1, \ldots, i_n$ is a permutation of the numbers $1, 2, \ldots, n$. The product $\mathbf{AP}$ rearranges the columns of $\mathbf{A}$ in the order $i_1, \ldots, i_n$, while the product $\mathbf{P}^T\mathbf{A}$ does the same with the rows of $\mathbf{A}$. It is valid the relation $\mathbf{PP}^T = \mathbf{P}^T\mathbf{P} = \mathbf{I}$.
- Let $\mathbf{A}$ be a symmetric matrix, and we investigate the possible values of the expression $f(\mathbf{x}) := \mathbf{x}^T\mathbf{Ax}$ if $\mathbf{x} \neq \mathbf{0}$:
  – always positive (negative): $\mathbf{A}$ positive (negative) definite,
  – always nonnegative (nonpositive): $\mathbf{A}$ positive (negative) semidefinite,
  – can be both positive and negative: $\mathbf{A}$ indefinite.
- Diagonally dominant matrix: $|a_{ii}| \geq \sum_{j=1, j\neq i}^{n} |a_{ij}|$, $\forall i = 1, \ldots, n$. Strictly diagonally dominant matrix if "$>$" is valid.

Eigenvalues and eigenvectors of matrices

Eigenvalues and eigenvectors

Def. 123. Suppose that there is a vector $\overline{\mathbf{v}} \neq \mathbf{0}$ and a number λ to the matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ such that $\mathbf{A}\overline{\mathbf{v}} = \lambda\overline{\mathbf{v}}$. Then the number λ is called the eigenvalue of the matrix $\mathbf{A}$, while the vector $\overline{\mathbf{v}}$ is called an eigenvector corresponding to the eigenvalue λ.

Thm. 124. Eigenvalues are the solutions of the so-called characteristic equation $\det(\mathbf{A} - \lambda\mathbf{I}) = 0$. (Real values or complex conjugate pairs.) The number of eigenvalues counted with multiplicity is n (algebraic multiplicity). Proof. Trivial. ■

Thm. 125. The linear combinations of eigenvectors are also eigenvectors ($\neq \mathbf{0}$). Proof. Trivial. ■

Thm. 126. $\exists \mathbf{A}^{-1} \Leftrightarrow \lambda_i \neq 0, \ \forall\ i = 1, \ldots, n$. Proof. Trivial. ■

Thm. 127.

$$\det(\mathbf{A}) = \prod_{i=1}^{n} \lambda_i, \quad \mathrm{tr}(\mathbf{A}) = \sum_{i=1}^{n} \lambda_i.$$

Proof. It can be proven with investigation of the coefficients of the characteristic polynomial. ■

Rmk. The eigenvalues can be complex numbers. In this case the eigenvectors also have complex elements.

Def. 128. For complex matrices $\mathbf{A}$, $\mathbf{A}^H$ denotes the transpose conjugate of the matrix. If $\mathbf{A}^H = \mathbf{A}$ is valid, then the matrix is called hermitian matrix. A matrix in unitary if $\mathbf{A}^H\mathbf{A} = \mathbf{A}\mathbf{A}^H = \mathbf{I}$.

Thm. 129. All eigenvalues of symmetric (real) matrices are real, the eigenvectors can be chosen to real vectors.

Proof. Let $\overline{\mathbf{v}}$ be an eigenvector with the eigenvalue λ. Then $\overline{\mathbf{v}}^H \mathbf{A}\overline{\mathbf{v}} = \overline{\mathbf{v}}^H \lambda \overline{\mathbf{v}} = \lambda \overline{\mathbf{v}}^H \overline{\mathbf{v}}$. Trivially

$$(\overline{\mathbf{v}}^H \mathbf{A}\overline{\mathbf{v}})^H = \overline{\mathbf{v}}^H \mathbf{A}\overline{\mathbf{v}}, \quad (\overline{\mathbf{v}}^H \overline{\mathbf{v}})^H = \overline{\mathbf{v}}^H \overline{\mathbf{v}},$$

that is these are 1×1 matrices. The conjugate transpose of these matrices are themselves. Thus λ must be real. The eigenvectors are the solutions of the system of equations $(\mathbf{A} - \lambda \mathbf{I})\overline{\mathbf{x}} = \mathbf{0}$, which can be chosen to be real. ■

Thm. 130. All eigenvalues of symmetric, positive (semi)definite matrices are (nonnegative) positive.

Proof. Let $\overline{\mathbf{v}}$ be an eigenvector with the eigenvalue λ (real). Then the statement follows from the equalities $\overline{\mathbf{v}}^T\mathbf{A}\overline{\mathbf{v}} = \overline{\mathbf{v}}^T\lambda\overline{\mathbf{v}} = \lambda\overline{\mathbf{v}}^T\overline{\mathbf{v}} > 0$ and $\overline{\mathbf{v}}^T\overline{\mathbf{v}} > 0$ (the proof is similar for semidefinite matrices). ■

Def. 131. The greatest absolute value of the eigenvalues of the matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$ is called the spectral radius of $\mathbf{A}$. Notation: $\varrho(\mathbf{A})$. That is

$$\varrho(\mathbf{A}) = \max\{|\lambda_i| \,|\, \lambda_i \text{ is an eigenvalue of } \mathbf{A}\}.$$

Gershgorin theorem

Thm. 132. Let us consider the matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$. Let K_i be the closed circle on the complex plane defined as follows. Its center is a_{ii} and its radius is $\sum_{j=1, j \neq i}^{n} |a_{ij}|$ $(i = 1, \ldots, n)$. Then all the eigenvalues of the matrix are in the set $\cup_i K_i$.

Proof. Let λ be an eigenvalue of the matrix. If λ equals one of the diagonal elements, then the statement is true for this eigenvalue. Otherwise, let us write $\mathbf{A}$ in the form $\mathbf{A} = \mathbf{D} + \mathbf{T}$, where $\mathbf{D}$ is the diagonal matrix of $\mathbf{A}$. $\mathbf{A} - \lambda \mathbf{I}$ is singular, thus there exists a vector $\overline{\mathbf{x}} \neq \mathbf{0}$, with which $(\mathbf{A} - \lambda \mathbf{I})\overline{\mathbf{x}} = \mathbf{0}$, that is $(\mathbf{D} - \lambda \mathbf{I})\overline{\mathbf{x}} = -\mathbf{T}\overline{\mathbf{x}}$.

Gershgorin theorem

Hence

$$\|\overline{\mathbf{x}}\|_\infty \leq \|(\mathbf{D} - \lambda\mathbf{I})^{-1}\mathbf{T}\|_\infty \|\overline{\mathbf{x}}\|_\infty,$$

that is

$$1 \leq \frac{\sum_{j=1, j\neq k}^{n} |a_{kj}|}{|a_{kk} - \lambda|}$$

for some index $k = 1, \ldots, n$. Thus $\lambda \in K_k$. ■

Rmk. When the union of s Gershgorin circles is disjoint from the other circles, then the union contains exactly s eigenvalues (2. Gershgorin theorem).

Diagonalizability of matrices

Diagonalizability

Def. 133. Two quadratic matrices $(\mathbf{A}, \mathbf{B})$ are similar if $\exists$ $\mathbf{S}$ nonsingular matrix, for which $\mathbf{B} = \mathbf{S}^{-1}\mathbf{A}\mathbf{S}$.

Thm. 134. The eigenvalues of similar matrices are equal.

Proof.

$$\det(\mathbf{B} - \lambda\mathbf{I}) = \det(\mathbf{S}^{-1}\mathbf{A}\mathbf{S} - \lambda\mathbf{I})$$

$$= \det(\mathbf{S}^{-1})\det(\mathbf{A} - \lambda\mathbf{I})\det(\mathbf{S}) = \det(\mathbf{A} - \lambda\mathbf{I}). \blacksquare$$

Rmk. If $\overline{\mathbf{v}}$ is an eigenvector of $\mathbf{B}$ then $\mathbf{S}\overline{\mathbf{v}}$ is an eigenvector of $\mathbf{A}$.

Def. 135. A matrix $\mathbf{A}$ is called diagonalizable if it is similar to a diagonal matrix.

Diagonalizability

Ex.: Not diagonalizable:

$$\mathbf{A} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$$

. 1 is double eigenvalue, thus it must be similar to the identity matrix but then $\mathbf{A} = \mathbf{S}^{-1}\mathbf{I}\mathbf{S} = \mathbf{I}$, which is not true.

Thm. 136. Eigenvectors that belong to different eigenvalues are linearly independent.

Proof. Suppose $\mathbf{A}\overline{\mathbf{v}} = \lambda\overline{\mathbf{v}}$ és $\mathbf{A}\overline{\mathbf{w}}_i = \mu\overline{\mathbf{w}}_i$ $(i = 1, \ldots, l)$, $\lambda \neq \mu$ and $\overline{\mathbf{v}} = \sum_{i=1}^{l} \alpha_i \overline{\mathbf{w}}_i$ for some constant $\alpha_i \neq 0$. Then

$$\lambda\overline{\mathbf{v}} = \mathbf{A}\overline{\mathbf{v}} = \mathbf{A}\sum_{i=1}^{l} \alpha_i \overline{\mathbf{w}}_i = \mu \sum_{i=1}^{l} \alpha_i \overline{\mathbf{w}}_i = \mu\overline{\mathbf{v}},$$

which implies the equality $\lambda = \mu$. ■

Cor.: When all the eigenvectors of a matrix are different, then the matrix has a linearly independent eigenvector system.

Diagonalizability

Thm. 137. An $n \times n$ matrix is diagonalizable if and only if it has a linearly independent eigenvector system with n vectors.

Proof. $\Leftarrow \mathbf{A}\overline{\mathbf{v}}_j = \lambda_j \overline{\mathbf{v}}_j \ (j = 1, \ldots, n)$

$$\mathbf{A} \underbrace{\begin{bmatrix} \overline{\mathbf{v}}_1 & \ldots & \overline{\mathbf{v}}_n \end{bmatrix}}_{:=\mathbf{S}} = \begin{bmatrix} \overline{\mathbf{v}}_1 & \ldots & \overline{\mathbf{v}}_n \end{bmatrix} \underbrace{\begin{bmatrix} \lambda_1 & 0 & 0 & \ldots \\ 0 & \lambda_2 & 0 & \ldots \\ & & \ddots & \end{bmatrix}}_{:=\mathbf{\Lambda}}$$

Thus $\mathbf{S}^{-1}\mathbf{A}\mathbf{S} = \mathbf{\Lambda}$, that is the matrix is diagonalizable.

$\Rightarrow \exists \mathbf{S}$ regular matrix, with which $\mathbf{S}^{-1}\mathbf{A}\mathbf{S} = \mathbf{\Lambda}$ for some diagonal matrix $\mathbf{\Lambda}$. Then the eigenvalues of $\mathbf{A}$ equal the elements of $\mathbf{\Lambda}$. Since the system $\overline{\mathbf{e}}_j$ is an eigenvector system of $\mathbf{\Lambda}$, $\mathbf{S}\overline{\mathbf{e}}_j$ is an eigenvector system of $\mathbf{A}$. These are linearly independent vectors because of the regularity of $\mathbf{S}$. ■

Diagonalizability

Def. 138. A matrix $\mathbf{A}$ is called normal if $\mathbf{A}^H\mathbf{A} = \mathbf{A}\mathbf{A}^H$.

Thm. 139. Normal matrices are diagonalizable.

Proof. Let λ_1 and $\overline{\mathbf{v}}_1$ be an eigenvalue and the corresponding eigenvector of the matrix (these always exist - they can be complex). Let $\overline{\mathbf{v}}_1$ satisfy the condition $\overline{\mathbf{v}}_1^H\overline{\mathbf{v}}_1 = 1$ (the vector is normed). Let us extend this vector to a unitary system $(\overline{\mathbf{v}}_2, \ldots, \overline{\mathbf{v}}_n)$. Then

$$\mathbf{A}\underbrace{\begin{bmatrix} \overline{\mathbf{v}}_1 & \ldots & \overline{\mathbf{v}}_n \end{bmatrix}}_{:=\mathbf{S}_1 \text{ unitér}} = \begin{bmatrix} \overline{\mathbf{v}}_1 & \ldots & \overline{\mathbf{v}}_n \end{bmatrix} \begin{bmatrix} \lambda_1 & * & * & \ldots \\ 0 & * & * & \ldots \\ & & \ddots & \\ 0 & * & * & \ldots \end{bmatrix}.$$

Thus

$$\mathbf{S}_1^H\mathbf{A}\mathbf{S}_1 = \begin{bmatrix} \lambda_1 & * \\ \mathbf{0} & \mathbf{A}_2 \end{bmatrix}.$$

Diagonalizability

Let us repeat the previous procedure for the matrix $\mathbf{A}_2$! There exists a unitary matrix $\tilde{\mathbf{S}}_2$ such that

$$\tilde{\mathbf{S}}_2^H \mathbf{A}_2 \tilde{\mathbf{S}}_2 = \begin{bmatrix} \lambda_2 & * & * & \dots \\ 0 & * & * & \dots \\ & & \ddots & \\ 0 & * & * & \dots \end{bmatrix}.$$

Let

$$\mathbf{S}_2 = \begin{bmatrix} 1 & \mathbf{0} \\ \mathbf{0} & \tilde{\mathbf{S}}_2 \end{bmatrix}.$$

Then

$$\mathbf{S}_2^H \mathbf{S}_1^H \mathbf{A} \mathbf{S}_1 \mathbf{S}_2 = \begin{bmatrix} \lambda_1 & * & * & \dots \\ 0 & \lambda_2 & * & \dots \\ & & \ddots & \\ 0 & 0 & * & \dots \end{bmatrix}.$$

Diagonalizability

Similarly, we can obtain the unitary matrices $\mathbf{S}_3, \ldots, \mathbf{S}_{n-1}$. With these matrices we have

$$\mathbf{S}_{n-1}^H \ldots \mathbf{S}_2^H \mathbf{S}_1^H \mathbf{A} \mathbf{S}_1 \mathbf{S}_2 \ldots \mathbf{S}_{n-1} = \underbrace{\begin{bmatrix} \lambda_1 & * & * & \ldots & * \\ 0 & \lambda_2 & * & \ldots & * \\ & & \ddots & & \\ 0 & 0 & 0 & \ldots & \lambda_n \end{bmatrix}}_{:=\mathbf{T} \text{ (upper triangular)}}.$$

Let $\mathbf{S} = \mathbf{S}_1 \ldots \mathbf{S}_{n-1}$. This is a unitary matrix.

$$\mathbf{T}^H \mathbf{T} = \mathbf{S}^H \mathbf{A}^H \mathbf{S} \mathbf{S}^H \mathbf{A} \mathbf{S} = \mathbf{S}^H \mathbf{A}^H \mathbf{A} \mathbf{S},$$

$$\mathbf{T} \mathbf{T}^H = \mathbf{S}^H \mathbf{A} \mathbf{S} \mathbf{S}^H \mathbf{A}^H \mathbf{S} = \mathbf{S}^H \mathbf{A} \mathbf{A}^H \mathbf{S},$$

thus $\mathbf{T}$ is normal. $\mathbf{T}$ can be upper triangular only if it is diagonal. ■

Diagonalizability

Rmk. Every matrix can be written in the form $\mathbf{A} = \mathbf{STS}^H$, where $\mathbf{S}$ is unitary and $\mathbf{T}$ is an upper triangular matrix. This is the so called Schur decomposition.

Rmk. Normal matrices can be diagonalized with a unitary matrix. Matrices that are diagonalizable with a unitary matrix are normal.

Rmk. Real normal matrices are e.g. symmetric, skew-symmetric and orthogonal matrices.

Thm. 140. A real matrix is diagonalizable with an orthogonal matrix if and only if it is symmetric.

Proof. $\Rightarrow$ Let $\mathbf{S}$ be orthogonal and $\mathbf{A} = \mathbf{S\Lambda S}^T$. Then $\mathbf{A}^T = \mathbf{S\Lambda S}^T = \mathbf{A}$, which shows the symmetry.

Diagonalizability

$\Leftarrow$ Let $\overline{\mathbf{v}}_\lambda$ and $\overline{\mathbf{v}}_\mu$ be two eigenvalues corresponding to two different eigenvectors (λ and μ).

$$\overline{\mathbf{v}}_\lambda^T \mathbf{A} \overline{\mathbf{v}}_\mu = \overline{\mathbf{v}}_\lambda^T \mu \overline{\mathbf{v}}_\mu = \mu \overline{\mathbf{v}}_\lambda^T \overline{\mathbf{v}}_\mu,$$

$$\overline{\mathbf{v}}_\mu^T \mathbf{A} \overline{\mathbf{v}}_\lambda = \overline{\mathbf{v}}_\mu^T \lambda \overline{\mathbf{v}}_\lambda = \lambda \overline{\mathbf{v}}_\mu^T \overline{\mathbf{v}}_\lambda = \lambda \overline{\mathbf{v}}_\lambda^T \overline{\mathbf{v}}_\mu$$

These two values must be equal. This is possible only if $\overline{\mathbf{v}}_\lambda^T \overline{\mathbf{v}}_\mu = 0$. Thus the eigenvectors corresponding to different eigenvalues are orthogonal. Thus we can choose an orthonormal system of eigenvectors. The matrix can be diagonalized with the matrix that have the orthonormal eigenvectors in the columns. ■