Numerical methods in chemistry - lecture

Why: - to run programs not as black-box ones
- to write your own ones (using ready routines)

Numerical analysis: - methods used in computer solutions of engineering and scientific tasks
- development of algorithms, test of algorithms

+

analytic solution $\leftrightarrow$ numerical solution

↑ for few cases ↑ for many cases

Errors:
- finite length of numbers, e.g. 8-16 digits
  e.g. adding 10^8 numbers, the average is good for 8 digits
- truncation of function calculations using series (exp...)
- truncation error of input data
- error of numerical method

Stability: increases or decreases previous error

Robust: good for many different input values

Lecture: 66% Tóth, 33% Császár

Practice: mostly Tóth

best source for Tóth's topics:
- „Numerical Recipes in Fortran/C/P." Press, Teukolsky, Vetterling, Flannery
  Cambridge University Press
- Check on Internet, e.g. wikipedia

Solution of linear algebraic equations

Mathematical background: matrix → place and value

$$A = \begin{pmatrix} a_{11} & \cdots & a_{1m} \\ \vdots & \ddots & \vdots \\ a_{n1} & & a_{nm} \end{pmatrix} \qquad A^{n\times m}$$

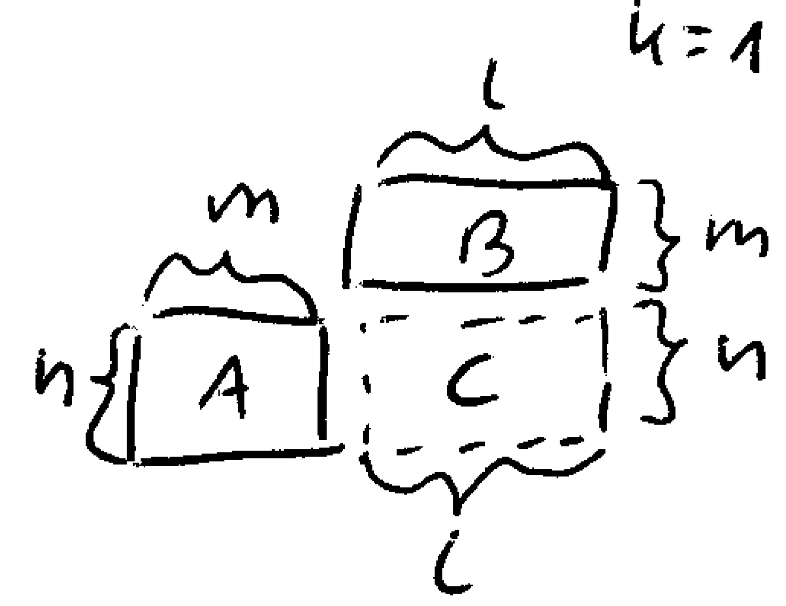
$$C = A \pm B \leftrightarrow c_{ik} = a_{ik} \pm b_{ik}$$

$$k\cdot A \leftrightarrow k\cdot a_{ij}$$

column vector $A^{n\times 1}$

row vector $A^{1\times m}$ ← dimensions in upper indeces

$$C = A\cdot B \qquad c_{ij} = \sum_{k=1}^{m} a_{ik}\cdot b_{kj}$$



determinant:

$$|A| = \det A = \sum_{\substack{\text{permut.} \\ (n!)}} (-1)^{I} \underbrace{a_{1k_1} \cdots a_{nk_n}}_{\text{one from each row and column}}$$

~~featurent~~

features of determinant:
- all a_{ij} in one row $= 0 \Rightarrow |A| = 0$
- one row $= k\cdot$ one row $\Rightarrow |A| = 0$
- One row = other row + something: $|A|$ is ~~the~~ the same as one row = something
- true for columns as well
- $|AB| = |A|\cdot|B|$

$\det A = 0$ singular

$\det A \approx 0$ quasi singular

N dimensional Euclidean space, basis vectors $\underline{a}_1 \ldots \underline{a}_n \to A$

linearly dependent vectors $\Leftrightarrow \det A = 0$

independent $\det A \neq 0$

independent: $N \approx$ dimension of $A \approx$ „rank"

singular ↔ rank $< n$ ↔ linearly dependent

nonsingular ↔ rank $= n$ ↔ linearly independent

Special matrices:

quadratic: $n = m$ unit matrix: $E = I = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

transpose: A^T $a^T_{ik} = a_{ki}$ symmetric: $A^T = A$

inverse for quadratic: $A^{-1}A = A \cdot A^{-1} = E$

A^{-1} can be calculated e.g. by ~~[illegible] method~~

$$a^{-1}_{ik} = \frac{\det A_{ki}}{\det A} \leftarrow$$ A without k-th row and i-th column + sign of ± 1

$\begin{pmatrix} + & - & + & - \\ - & + & - & + \\ + & - & + & - \end{pmatrix}$ multiplier

Moore-Penrose inverse

$n > m$: $A^+ = (A^T A)^{-1} A^T$

$n < m$: $A^+ = A^T (A \cdot A^T)^{-1}$

Set of linear equations: n unknowns, n equations

$$\left.\begin{aligned} a_{11}x_1 + a_{12}x_2 + \dots &= b_1 \\ a_{21}x_1 + \dots &= b_2 \\ a_{n1}x_1 + \dots \quad a_{nn} &= b_n \end{aligned}\right\}$$

$A^{n\times n} \in \mathbb{R}$ $\underline{x} \in \mathbb{R}$ $\underline{b} \in \mathbb{R}$

$$A\underline{x} = \underline{b}$$

inhomogen: if $\exists\, b_i \neq 0$ $\exists$ = exists   one non-zero b_i exists

Unique solution, if $\det A \neq 0$ for inhomogeneous set of linear equations.

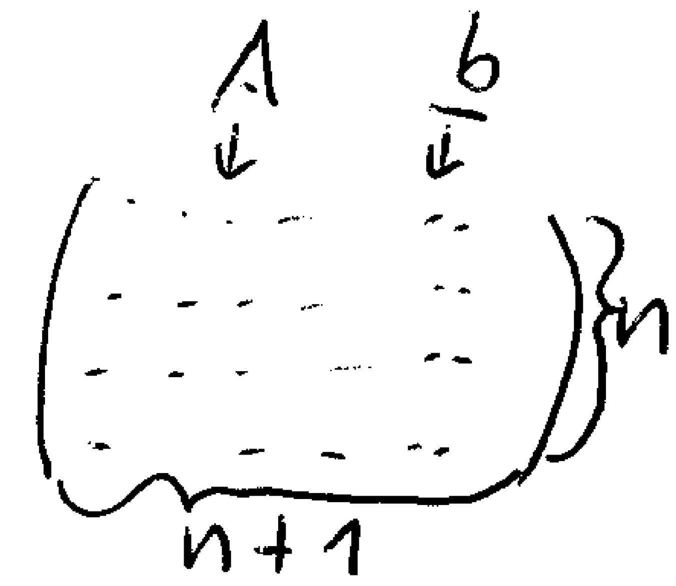
if $A^{n\times m}\underline{x}^m \approx b^n$ $n > m$ overdetermined case $\approx$ min-max search

$n < m$ underdetermined case $\approx$ extra assumption are necessary.

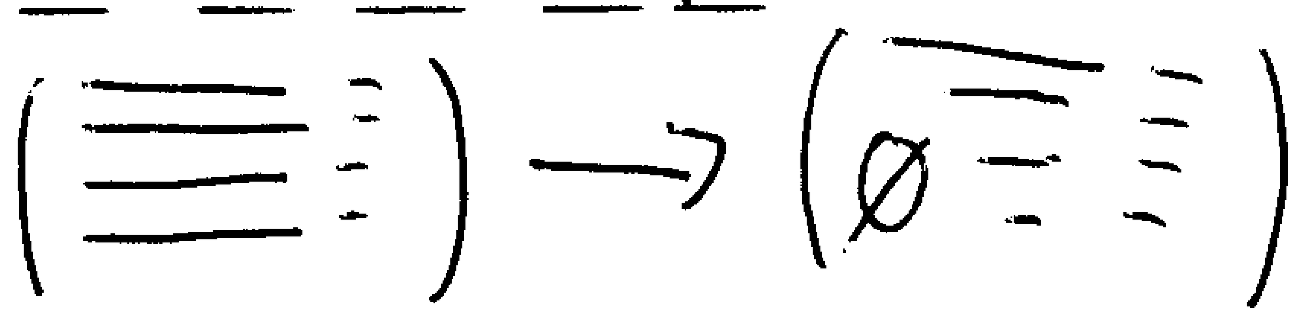
Direct methods

Gauss-elimination

extended matrix form: $\left(A \ \underline{b} \right) =$



elimination step:



in i-th iteration add to k-th row the $\left(-\frac{a_{ki}}{a_{ii}}\right)$ times the i-th row.

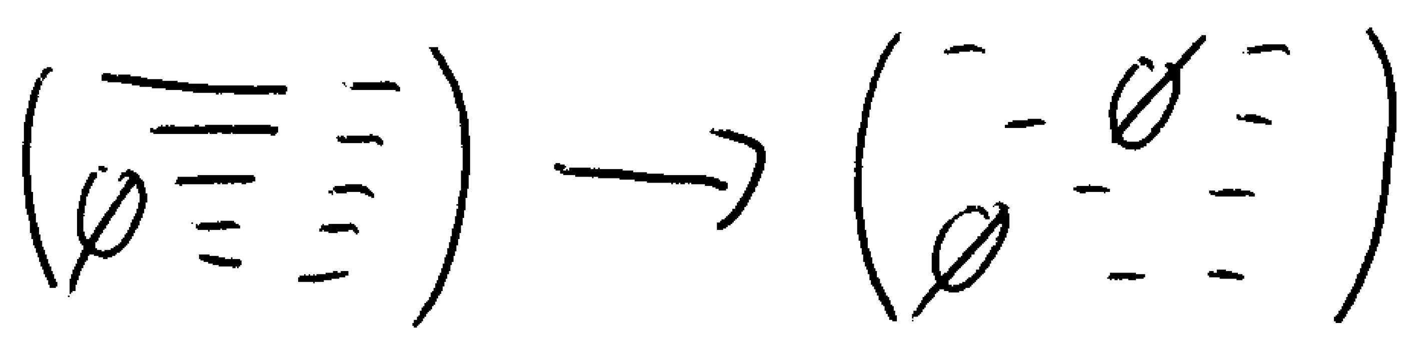
i-th version, (i-1)th version of the matrix

$$a_{kl}^{(i)} = a_{kl}^{(i-1)} - \frac{a_{ki}^{(i-1)}}{a_{ii}^{(i-1)}} \cdot a_{il}^{(i-1)} \qquad i = 1, 2, \dots, n-1$$

$$n \geq k > i$$

$$(n+1) \geq l \geq i$$

back-substitution step



In i-th step add to row j the $-\frac{a_{j,n-i+1}}{a_{n-i+1,n-i+1}}$ times row $n-i+1$. $\left(i = 1, 2, \dots (n-1) \quad j < (n-i+1)\right)$ (backward ~~downt~~ bottom to top)

then: $x_i = \frac{b_i}{a_{ii}}$

number of addings and multiplications scaled as:

$$\frac{n^3}{3} + n^2 - \frac{n}{3} \approx \text{computational cost}$$

Pivoting: a_{ii} can be $\emptyset$ or very small $|a_{ii}| < \varepsilon$

division is problematic → wrong results

1.) change of rows in ~~step~~ i-th iteration to have large a_{ii} ~~for~~ (partial pivoting)

2.) change both rows and columns

row change: no effect

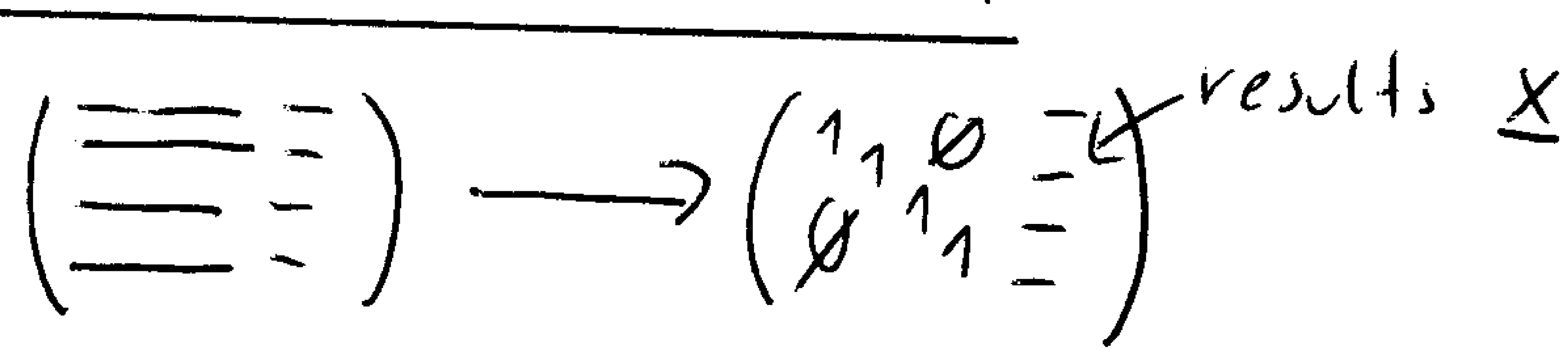
column change: track the change of unknowns.

coded in permutation matrices, e.g row permutation:

$$P = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad A' = PA$$

↑ row permutated A

Jordan-elimination



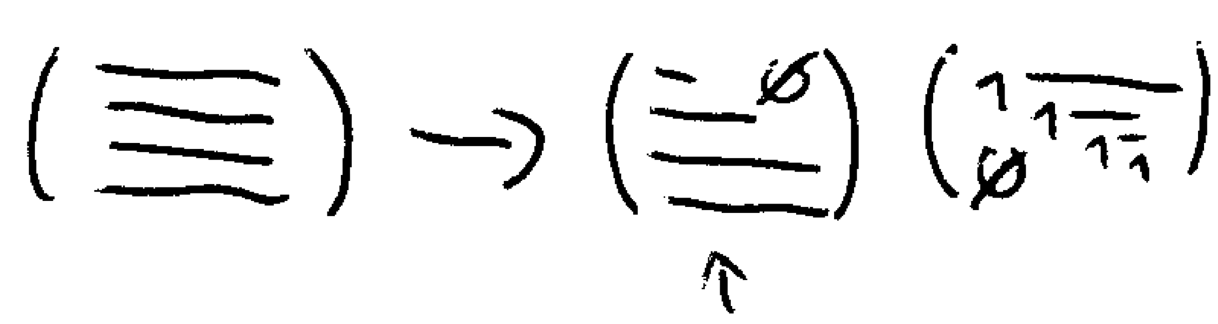
In i-th iteration: - division of i-th row with $a_{ii}^{(i-1)}$

- substract from k-th row ~~the~~ $a_{ji}^{(i-1)}$ times the new i-th row.

$$a_{kl}^{(i)} = a_{kl}^{(i-1)} - \frac{a_{ki}^{(i-1)}}{a_{ii}^{(i-1)}} \, a_{il}^{(i-1)} \qquad \begin{array}{l} i = 1, 2, \ldots n \\ 1 \le k \le n, \; k \ne i \\ i \le l \le (n+1) \end{array}$$

comp cost $\approx \frac{n^3}{2} + n^2 - \frac{7n}{2} + 2$

LU-decomposition



L = lower triangle   U = upper triangle

in L or U only 1-s at the diagonal

formally: $A\underline{x} = \underline{b}$

$PA\underline{x} = P\underline{b}$

$PA = LU$ decomposition is independent from $\underline{b}$

$LU\underline{x} = P\underline{b} = \underline{b}'$

$U\underline{x} = \underline{d}$

1.) $L\underline{d} = \underline{b}'$
2.) $U\underline{x} = \underline{d}$

} 2 backsubstitutions for ~~two~~ an lower and an upper triangle matrices.

Comp. cost: $\underbrace{\text{1 decomposition}}_{\text{only one time for } A \;\; \sim \frac{n^3}{3}} + \underbrace{\text{2 back subs.}}_{\text{for each } \underline{b} \;\; \sim n^2}$

for more than one $\underline{b}$ it is good, because the decomposition depends only on A

One algorith for LU decomposition is by Crout:

$$\begin{pmatrix} l_{11} & 0 & 0 & 0 \\ l_{21} & l_{22} & 0 & 0 \\ l_{31} & l_{32} & l_{33} & 0 \\ l_{41} & l_{42} & l_{43} & l_{44} \end{pmatrix} \begin{pmatrix} 1 & u_{12} & u_{13} & u_{14} \\ 0 & 1 & u_{23} & u_{24} \\ 0 & 0 & 1 & u_{34} \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ & & & \\ & & & \\ a_{41} & a_{42} & a_{43} & a_{44} \end{pmatrix}$$

from multiplication of matrices:

- $l_{11} = a_{11}$, $l_{21} = a_{21}$, $l_{31} = a_{31}$ $l_{41} = a_{41}$
- $l_{11} u_{12} = a_{12} \ldots$
-

calculate one by one

$\downarrow$

$n^3/3$ calculation cost.

Cholesky-decomposition (for spec. matrices)

- A symmetric
- positive definite: $\underline{x} A \underline{x} > 0$ for all $\underline{x}$

$A = L \cdot L^T$ ← transpose of lower triangle

↑ lower triangle

For other special matrices other special (fast) methods:

tridiagonal, sparse, block diagonal, Toeplitz matrix (shifted rows)

Cramer-method

$$x_i = \frac{\det A_i}{\det A} \leftarrow A_i \text{ i-th column in } A \text{ is replaced by } \underline{b}$$

How to calculate determinant?

for triangle matrices: $\det U = \prod_{i=1}^{n} u_{ii} = u_{11} \cdot u_{22} \cdot \ldots u_{nn}$

product

1.) LU decomposition

2.) product calc.

How to calculate inverse matrix?

$A \cdot A^{-1} = E$ for first column of A^{-1} $\quad A(\,) = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \leftarrow \underline{b_1}$

second .... $\quad A(\,) = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} \leftarrow \underline{b_2}$ $\quad \underline{b_n} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$

1) LU decomposition

2) n times solution for $\underline{b_1}$ to $\underline{b_n}$ $\quad$ comp. cost: $4/3 \cdot n^3$

Iterative methods

Jacobi-iteration

~~th~~theory: - permutate to all $a_{ii} \neq 0$

$$A\underline{x} = (L+D+U)\underline{x} = \underline{b}$$

$$D\underline{x} = -(L+D)\underline{x} + \underline{b}$$

x is on both sides

let ↑this as new (k+1) version ↑this as old (k)-th version ~ general scheme for iterative methods

$$\underline{x}^{(k+1)} = -D^{-1}(L+U)\underline{x}^{(k)} + D^{-1}\underline{b}$$

convergent always if $|a_{ii}| > \sum_{j \neq i} a_{ij}$

„diagonally" dominant in rows

practice:

$$x_i^{(k+1)} = -\frac{1}{a_{ii}} \left(a_{i1} x_1^{(k)} + a_{i2} x_2^{(k)} + \dots \cancel{a_{ii} x_i^{(k)}} \dots + a_{in} x_n^{(k)} - b_i \right)$$

missing

$i = 1, 2, \dots n$

$k = 1, \dots ?$

stop when $\|\underline{x}^{(k+1)} - \underline{x}^{(k)}\| < \varepsilon$

vector norm, see later

small pos. real number

starting $\underline{x}^{(0)} = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$

good convergence in 10-100 iterations

Seidel-iteration

theory: $A\underline{x} = (D + L + U)\underline{x} = \underline{b}$

$$(D+L)\underline{x} = -U\underline{x} + \underline{b}$$

$$\underline{x} = -(D+L)^{-1} U \underline{x} + (D+L)^{-1} \underline{b}$$

practice:

$$x_i^{(k+1)} = -\frac{1}{a_{ii}} \left(a_{i1} x_1^{(k+1)} + a_{i2} x_2^{(k+1)} + \dots + \cancel{a_{ii} x_i^{(k)}} + \dots + a_{in} x_n^{(k)} - b_i \right)$$

new x_i-s

missing

old x_i-s

Convergence can be adjusted ~~by~~ by replacing

$\frac{1}{a_{ii}}$ to $\frac{w}{a_{ii}}$ $\quad 1 < w < 2$

faster relaxation

Norm and Condition in linear algebra ~~equations~~

Norm is a quantity as $\|A\| \geq 0$ positive

$\|k \cdot A\| = k\|A\|$ $\quad k$ constant

$\|A + B\| \leq \|A\| + \|B\|$ triangle ~~rule~~ inequality

$\|A\| \cdot \|B\| \geq \|A \cdot B\|$

norm of a scalar $x \in \mathbb{R}$:

absolute value $|x|$

vector norms $\underline{x} \in \mathbb{R}^n$

$$\|\underline{x}\|_1 = \sum_{i=1}^{n} |x_i| \quad \text{sum of length}$$

$$\|\underline{x}\|_2 = \left(\sum_{i=1}^{n} x_i^2\right)^{1/2} \quad \text{Euclidean} \leftarrow \text{most popular}$$

$$\|\underline{x}\|_\infty = \max |x_i|$$

matrix norms: $A \in \mathbb{R}^{n \times m}$

$$\|A\|_1 = \max_{(1 \le j \le m)} \sum_{i=1}^{n} |a_{ij}| \quad \text{max columnsum}$$

$$\|A\|_\infty = \max_{(1 \le i \le n)} \sum_{j=1}^{m} |a_{ij}| \quad \text{maximal row sum}$$

$$\|A\|_2 = \sqrt{\lambda_{max}} \quad \text{maximal eigenvalue of } A \cdot A^T \text{ - spectral norm}$$

$$\|A\|_f = \left(\sum_i \sum_j a_{ij}^2\right)^{1/2} \quad \text{Frobenius-norm}$$

$\|A\|_2 \le \|A_1\|, \|A\|_\infty$

<u>Ill-conditioned set of equations:</u>

$$\begin{pmatrix} 1.01 & 0.99 \\ 0.99 & 1.01 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 2 \\ 2 \end{pmatrix} \Rightarrow x = 1 \quad y = 1$$

same $= \begin{pmatrix} 1.98 \\ 2.02 \end{pmatrix} \Rightarrow x = 2 \quad y = 0$

ill conditioned: sensitive to input $\underline{b}$ (errors)

$\underline{e} = \underline{x} - \underline{\tilde{x}}$ $\quad \underline{\tilde{x}}$ obtained, $\underline{x}$ real unknowns

$\underline{r} = \underline{b} - \underline{\tilde{b}}$ $\quad \underline{\tilde{b}}$ obtained, $\underline{b}$ real resultvector

↑ residual

can be shown:

$$\frac{\|e\|}{\|\underline{x}\|} \le \underbrace{\|A\| \cdot \|A^{-1}\|}_{\text{condition number}} \frac{\|r\|}{\|b\|}$$

$1 \le \text{Cond.num} \le \infty$

↑ good $\qquad$ ↑ large is ill-cond.

$\Rightarrow$ error in the unknowns can be estimated using condition number + error of $\underline{b}$ (result vector)