1) Errors and numbers

1.1 Preliminary considerations

There are three main sources of error in the numerical analysis of physical systems:

i) rounding errors, since instead of numbers $t \in \mathbb{R}$ a finite set of numbers is calculated.

ii) "discretization" or "truncation error"

Examples:

- $\int_a^b f(x)\,dx$ is replaced by a finite sum $(dx \to \Delta x)$
- Power series $P(x) = \sum_{n=0}^{\infty} p_n x^n$ is approximated by a finite sum

iii) physical approximation errors:

usually physical approximations are made before the numerical evaluation. Examples from solid state physics: Hartree–Fock approximation, density functional theory, ...

Estimation of the order of magnitude of the errors:

iii) usually difficult

ii) order of magnitude can often be estimated. If you know, for example, that a property $F(\Delta x)$ is linear in Δx for $\Delta x \to 0$ then one may calculate $F_i = F(\Delta x_i)$ for some values Δx_i

$\Rightarrow$ linear fit $F^{fit}(\Delta x) = F(0) + m\,\Delta x$

$\Rightarrow$ $F(0)$ the property one is looking for plus statistical error

i) We consider these errors in this chapter

1.2 Rounding errors

1.2.1 Representation of numbers

A) Integer numbers

Computers construct whole ("integer") numbers in the form

$$x = \pm (a_m 2^m + a_{m-1} 2^{m-1} + \cdots + a_0 2^0) = \pm \sum_{i=0}^{m} a_i 2^i$$

with $a_i = 0$ or $a_i = 1$ (value of bit i)

With 16 bits: Bit 1 determines $\pm$

Bit 2–16 determines $|x|$

$\Rightarrow \exists\ 2^{15}$ such numbers between 0 and $2^{15}-1 = 32767$

so $x \in \{-32767, \dots, -1, 0, 1, \dots, 32767\}$

Modern programming languages translate this coding into the decimal system

$$x = \pm \sum_{i=0}^{m} a_i \cdot 10^i$$

For the sake of simplicity, we will only consider the decimal system in the following. However, all considerations apply completely analogously to the binary system.

B) "floating point"="fp" representation

Question: How do you represent real numbers (approximately)?

Idea: by using rational numbers of the form

$$x = a \cdot 10^b$$

with $0 \le a < 1 \Rightarrow a = \pm 0.\overbrace{a_1 a_2 \cdots a_t}^{\text{integer number with } t \text{ digits}}$

and $b = \pm \underbrace{b_1 b_2 \cdots b_l}_{\text{integer number with } l \text{ digits}}$

One calls a the "mantissa" and t the mantissa length

Example: $30.40 = 0.3040 \cdot 10^2$

$= 0.0304 \cdot 10^3$

The representation is therefore not unique. An fp representation is called "normalized" if $a_1 \neq 0$ (we only consider these in the following)

So: fp numbers represent a finite set A (since $t, l < \infty$) of rational numbers

problem: $A \neq \mathbb{R}$ → chapter 1.2.2

1.2.2 Rounding errors

Main problem: If $x, y \in A$ it is not guaranteed that

$$x \pm y,\ x \cdot y,\ x / y \quad \in A$$

Example: $(t = 3,\ l = 1)$

$$0.300 \cdot 10^{0} + 0.300 \cdot 10^{-4} = 0.3003 \notin A$$

Therefore, one defines for $x \notin A$ the value $rd(x) \in A$ the number that is "closest" to x, i.e.

$$|x - rd(x)| \leq |x - y| \quad \forall\, y \in A$$

satisfies. There are 2 cases:

i) $x = \pm a \cdot 10^{b} \notin A$ since the mantissa has more than t digits. Then $rd(x)$ is the well-known rounding procedure, so

$$a = 0.a_1 \dots a_t a_{t+1} \dots$$

is

$$rd(x) = \pm a' \cdot 10^{b}$$

and

$$a' = \begin{cases} 0.a_1 \dots a_t & \text{if } 0 \leq a_{t+1} \leq 4 \\ 0.a_1 \dots (a_t + 1) & \text{if } a_{t+1} > 4 \end{cases}$$

ii) $x = \pm a \cdot 10^{b} \notin A$ since b has more than l digits, so e.g. for $l = 2$ is

$$\underbrace{(0.1 \cdot 10^{60})}_{\in A} \cdot \underbrace{(0.1 \cdot 10^{60})}_{\in A} = \underbrace{0.1 \cdot 10^{\infty}}_{\notin A}$$

This is called "exponent overflow" and is almost always the result of a programming error (not relevant here)

More important: Since most operations result in numbers $x \notin A$

there occur relative rounding errors of the order of magnitude

$$\left| \frac{rd(x) - x}{x} \right| \leq \varepsilon$$

with the so-called "machine precision" $\varepsilon = 5 \cdot 10^{-(t+1)}$

1.2.3 fp arithmetic

Since rounding is required after many calculation operations, mathematically speaking, a new so-called fp arithmetic results in the set A:

Let $x, y \in A$, then

$$x \oplus y \equiv rd(x+y)$$
$$x \odot y \equiv rd(x \cdot y)$$
$$x \oslash y \equiv rd(x/y)$$

Problem: Almost all rules of real numbers are no longer valid, e.g. neither an associativity nor a distributive law.

Counterexample to the associative law: $(t=2, l=1)$

$$a = 0.52 \cdot 10^0$$
$$b = -0.51 \cdot 10^0$$
$$c = 0.50 \cdot 10^{-2}$$

$\Rightarrow$ i) $(a \oplus b) \oplus c = 0.10 \cdot 10^{-1} \oplus 0.50 \cdot 10^{-2} = 0.15 \cdot 10^{-1}$

ii) $a \oplus (b \oplus c) = 0.52 \cdot 10^0 \oplus (-0.51 \cdot 10^0) = 0.10 \cdot 10^{-1}$

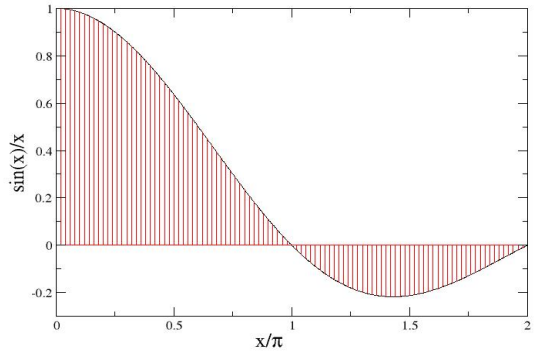
So: Be very careful when adding/subtracting numbers of different sizes and subtracting approximately equal numbers:

Example: Calculation of $\int_0^{2\pi} dx \; \frac{\sin(x)}{x}$

Simplest procedure (→ Chapter 4):
Break down the interval $[0, 2\pi]$ into small intervals of width

$$\Delta x = \frac{2\pi}{N}$$

Then $$\int_0^{2\pi} f(x)\,dx \approx \Delta x \sum_{i=1}^{N-1} f(i \cdot \Delta x)$$



Problem: If you naively just add up the strips, then with a large N you will eventually add up large numbers (the strips added up to that point) and small numbers (the next strip) -> big error.

1.3 Stability of iterative algorithms

Since there is always "numerical noise", algorithms must be stable with respect to this.

Example: Find numerical solution of the ordinary differential eq

$$\dot{y} = -y \;;\; y(0) = 1 \quad (1) \qquad [\text{exact: } y(t) = e^{-t}]$$

Anticipation of Chapter 7:

A) "Euler method" with discretization of t:

$$y_m = y(t_m) = y(m \cdot \Delta t) \text{ with } \Delta t \ll 1$$

$$\Rightarrow y_{m+1} - y_m = \underbrace{\dot{y}(t_m)}_{\overset{(1)}{=} -y_m} \cdot \Delta t + O(\Delta t^2) \qquad \text{(Taylor!)}$$

$$\Rightarrow y_{m+1} = y_m (1 - \Delta t) \qquad (2)$$

so: ODE (1) is replaced by recursion equation (2)

Case 1: no rounding errors

$\Rightarrow$ solution of (2), with initial condition $y_0 = 1$ is

$$y_n = (1 - \Delta t)^n$$

$$\Rightarrow \quad y(t) = \left(1 - \frac{t}{n}\right)^n \xrightarrow{n\to\infty} e^{-t}$$

i.e. one gets an arbitrarily good approximation by increasing n

Case 2: small rounding error at the beginning $y_0 = 1 + \underbrace{\Delta y}_{\ll 1}$

$$\overset{\text{as above}}{\Rightarrow} \quad y(t) = (1+\Delta y)\left(1 - \frac{t}{n}\right)^n \xrightarrow{n\to\infty} e^{-t} + \underbrace{\Delta y\, e^{-t}}_{\text{klein bleibt klein}}$$

$\Rightarrow$ algorithm is stable

Remark: We have only shown stability for the simplest of all disturbances $y_0 \to y_0 + \Delta y$. In reality there will be a Δy_n in every step.

B) We now seemingly "improve" the Euler method by symmetrization

$$y_{n+1} - y_{n-1} \simeq 2\,\dot{y}(t_n)\cdot\Delta t + \underbrace{O(\Delta t^3)}_{\text{more precise than in A)}}$$

$$\overset{(1)}{\Rightarrow} \quad y_{n+1} = -2\Delta t\, y_n + y_{n-1} \qquad (3)$$

Solution Ansatz for (3): $y_n = (1-T)^n$ (T: constant)

Insert: $(1-T)^2 + 2\cdot\Delta t\,(1-T) - 1 = 0$

$$\Rightarrow \quad (1-T) = -\Delta t \pm \sqrt{\Delta t^2 + 1} \overset{\Delta t \ll 1}{\simeq} -\Delta t \pm 1$$

$\Rightarrow$ General solution of (3) for $\Delta t \ll 1$:

$$y_n = \alpha\,(1-\Delta t)^n + \beta\,(-1-\Delta t)^n \qquad (4)$$

with arbitrary α, β

$\Rightarrow$ We require two initial conditions here $y_0 = 1$

$y_1 = (1 - \Delta t)$ (from A)

Insert (4) into initial condition: $\alpha = 1$, $\beta = 0$

$\Rightarrow y_n = (1 - \Delta t)^n \xrightarrow{t \to \infty}$ exact solution as in A)

But now again $y_0 = 1 + \beta$, such that $\beta \neq 0$

Problem: The solution corresponding to β

$$(-1 - \Delta t)^n = (-1)^n \underbrace{\left(1 + \frac{t}{n}\right)^n}_{\equiv x}$$

diverges and oscillates, because

$$\ln(x) = n \ln\left(\frac{t}{n}\right) \Rightarrow \text{Algorithm is unstable}$$

$n \to \infty$ $t \to \infty$ ∞

So: The accuracy of an algorithm usually says nothing about its stability

2) Solution of linear systems of equations, SVD

In CP, many algorithms require the solution of systems of linear equations (SLE) of the form

$$\tilde{A}\,\vec{x} = \vec{b}$$

with the MXN matrix $\tilde{A} = \begin{pmatrix} a_{11} & \cdots & a_{1N} \\ \vdots & & \\ a_{M1} & \cdots & a_{MN} \end{pmatrix}$

and the vectors $\vec{x} = (x_1, \ldots, x_N)^T$ (wanted)
$\vec{b} = (b_1, \ldots, b_M$ | given)

W.L.O.G.: $a_{ij}, x_i, b_i \in \mathbb{R}$. Complex LSE can be rewritten as real by splitting them into real and imaginary parts:

$$(\tilde{A}_R + i\tilde{A}_I)(\vec{x}_R + i\vec{x}_I) = \vec{b}_R + i\vec{b}_I$$

$$\Rightarrow \quad \tilde{A}_R \vec{x}_R - \tilde{A}_I \vec{x}_I = \vec{b}_R$$
$$\tilde{A}_R \vec{x}_I + \tilde{A}_I \vec{x}_R = \vec{b}_I$$

$$\Rightarrow \begin{pmatrix} \tilde{A}_R & -\tilde{A}_I \\ \tilde{A}_I & \tilde{A}_R \end{pmatrix} \begin{pmatrix} \vec{x}_R \\ \vec{x}_I \end{pmatrix} = \begin{pmatrix} \vec{b}_R \\ \vec{b}_I \end{pmatrix}$$

As is well known, there are (mathematically) three cases

I) $\exists$ exactly one solution (only possible if $M \geq N$)

II) $\exists$ infinitely many solutions ("underdetermined SLE")

III) $\exists$ no solution ("overdetermined SLE")

$\Rightarrow$ In CP usually the result of a programming error

Note: In numerical calculations the three cases can become blurred, e.g. M=N, then the following applies:

$\tilde{A}\,\vec{x} = \vec{b}$ has a unique solution if and only if $|\tilde{A}| \neq 0$

However: In numerical calculations there are no exact zeros

⇒ It is not always clear for which values of $|\tilde{A}| \neq 0$ the numerical solution of the SLE is reasonable

One is on the safe side if

$|\,|\tilde{A}|\,| \gtrsim$ typical order of magnitude of the matrix elements $|a_{i,j}| \neq 0$

2.1 Case 1: LSE with unique solution

2.1.1 M=N

A) Note: The solution of an SLE requires in general $2n^3$ operations. The various methods differ only in terms of 2 .

How better not to solve the problem?

i) Calculation of $\tilde{A}^{-1}$ and

$$\vec{x} = \tilde{A}^{-1}\vec{b}$$

Exception: you need $\tilde{A}^{-1}$ anyway (which is rarely the case)

ii) Gaussian method from HöMa 1

B) The "LU decomposition" is usually very efficient and stable:

Assume: $\tilde{A}$ is regular (i.e. $|\tilde{A}| \neq 0$ and the LSE has a unique solution). Then you can write $\tilde{A}$ as

$$\tilde{A} = \tilde{P} \cdot \tilde{L} \cdot \tilde{U}$$

where $\tilde{L}\,(\tilde{U})$ is a lower (upper) triangular matrix, i.e.

$$\tilde{L} = \begin{pmatrix} l_{11} & 0 & 0 & \cdots & 0 \\ l_{21} & l_{22} & & & \\ & & & & \\ & & & & l_{n,n} \end{pmatrix}, \quad \tilde{U} = \begin{pmatrix} u_{11} & u_{12} & & & \\ 0 & u_{22} & & & \\ 0 & & & \ddots & \\ \vdots & & & & \\ 0 & & & & u_{nn} \end{pmatrix}$$

and the orthogonal "pivoting matrix" ($\tilde{P}^T = \tilde{P}^{-1}$) in each row/column contains exactly one 1.

Proof: lengthy and boring (->Ref. 5)

Practical procedure-> maybe excercises

After the LU decomposition you can solve $\tilde{A}\vec{x}=\vec{b}$ in 3 steps:

$$\tilde{P}\left(\tilde{L}\left(\tilde{U}\cdot\vec{x}\right)\right)=\vec{b}$$

$$\Rightarrow \tilde{P}\vec{z}=\vec{b} \Rightarrow \vec{z}=\tilde{P}^{T}\cdot\vec{b} \quad (1)$$

$$\Rightarrow \tilde{L}\cdot\vec{y}=\vec{z} \quad (2) \qquad \Rightarrow \qquad \tilde{U}\cdot\vec{x}=\vec{y} \quad (3)$$

Advantage: Equations (2)/(3) are easy to solve, e.g. (2):

$$z_i=\sum_{j\le i} l_{ij}\,y_j$$

$$\Rightarrow i=1: z_1=l_{11}y_1 \Rightarrow y_1=z_1/l_{11}$$

$$i=2: z_2=l_{21}y_1+l_{22}y_2 \Rightarrow y_2=\frac{1}{l_{22}}\left(z_2-l_{21}y_1\right)$$

so:
$$y_i=\frac{1}{l_{ii}}\left(z_i-\sum_{j=1}^{i-1} l_{ij}\,y_j\right) \qquad (4)$$

Equation (4) always makes sense, because $l_{ii}\neq 0 \;\forall\, i$

Reason: Note ii) below

There are several very efficient algorithms for determining the Lu decomposition, implemented e.g. in "LAPACK" or "Eigen".

Remarks on the LU decomposition:

i) Advantage of the LU decomposition:
It only has to be carried out once (N^3 effort) and can then be applied to any $\vec{b}$. (equation (4) only contains $O(N^2)$ operations).

ii) $$|\tilde{A}|=\underbrace{|\tilde{P}|}_{=\pm 1}\cdot\underbrace{|\tilde{L}|}_{=\prod_{i=1}^{n} l_{ii}}\cdot\underbrace{|\tilde{U}|}_{=\prod_{i=1}^{n} u_{ii}} \qquad (5)$$

$\Rightarrow$ would be $l_{ii} = 0$ or $u_{ii} = 0 \Rightarrow |\tilde{A}| = 0 \Rightarrow \tilde{A}$ not regular $\lightning$

iii) Routines are common in which $l_{ii} = 1$ $\forall i$
(can always be set up)

iv) Inverse matrix

$$\tilde{A}^{-1} = \tilde{u}^{-1} \tilde{L}^{-1} \tilde{P}^{T}$$

$\tilde{u}^{-1}$, $\tilde{L}^{-1}$ are again easy to be determined, e.g.: $\tilde{L}^{-1}$

Sei: $(\tilde{L}^{-1})_{ij} \equiv \bar{l}_{ij}$

$\tilde{L} \cdot \tilde{L}^{-1} = \hat{1} \Rightarrow \sum_j l_{ij} \bar{l}_{jk} = \delta_{ik}$

$$\Rightarrow \sum_{j \le i} l_{ij} \bar{l}_{jk} = \delta_{ik}$$

- $i = 1$: $l_{11} \bar{l}_{1k} = \delta_{1k} \Rightarrow \bar{l}_{1k} = \delta_{1k} / l_{11}$
- $i = 2$: $l_{21} \bar{l}_{1k} + l_{22} \bar{l}_{2k} \Rightarrow l_{2k} = \frac{1}{l_{22}} (\delta_{2k} - l_{21} \bar{l}_{1k})$
- $i > 1$: $\bar{l}_{ik} = \frac{1}{l_{ii}} (\delta_{ik} - \sum_{j=1}^{i-1} l_{ij} \bar{l}_{jk})$ $(O(n^3))$

Remark: $\bar{l}_{ik} \neq 0$ only for $k \le i$ (proof: by induction)

$\Rightarrow \tilde{L}^{-1}$ is also a lower triangular matrix

$\tilde{u}^{-1}$ can be calculated analogously

Remarks:

For special matrices there may also be faster algorithms, examples:

i) symmetric, Hermitian, orthogonal, unitary matrices

ii) "band-diagonal matrices"

$a_{ij} = 0$ $\forall$ all's $|i-j| \ge m$

$$\tilde{A} = \begin{pmatrix} \ddots & \ddots & 0 \\ \ddots & a_{ij} \neq 0 & \ddots \\ 0 & \ddots & \ddots \end{pmatrix}$$

iii) "tridiagonal matrices" (will become important later)

like (ii) with m=1, i.e.

$$\tilde{A} = \begin{pmatrix} a_{11} & a_{12} & 0 & 0 & \\ a_{21} & a_{22} & a_{23} & & \\ 0 & a_{32} & \ddots & & \\ 0 & & & & a_{NN} \end{pmatrix}$$

iv) Sparse matrices

Matrices for which the number N' of a_{ij} with $a_{ij} \neq 0$ is

$$N' << N^2$$

Remark: ii) and iii) are special cases of iv) if

$$N >> 1 \quad \text{and} \quad m << N$$

Otherwise one must analyze how the $a_{ij} \neq 0$ are distributed (randomly, on the edge, in clusters, etc.)

$\Rightarrow$ The numerical solution of such SLE is a field in itself

2.1.2 M>N

So: $\tilde{A} \cdot \vec{x} = \vec{b}$ (1) :

$$\begin{pmatrix} & & \\ & & \\ & & \\ & & \\ & & \end{pmatrix} \cdot \begin{pmatrix} \\ \\ \vec{x} \end{pmatrix} = \begin{pmatrix} & \\ & \\ & \\ & \\ & \end{pmatrix}$$

$\tilde{A}$ $\qquad$ $\vec{b}$

In general there is no unique solution here.

Mathematical insertion:

Let $\tilde{A}$ be an MxN matrix. If the NxN matrix

$$\tilde{P} = \tilde{A}^T \cdot \tilde{A}$$

is invertible, then the matrix

$$\tilde{A}_p^{-1} \equiv \tilde{P}^{-1} \cdot \tilde{A}^T$$

is called the "Moore–Penrose–inverse" or "pseudo–inverse" of $\tilde{A}$, because

$$\tilde{A}_p^{-1} \cdot \tilde{A} = (\tilde{A}^T \cdot \tilde{A})^{-1} \cdot \tilde{A}^T \cdot \tilde{A} = \tilde{1}$$

Remarks:

i) If M=N, then obviously is $\tilde{A}_p^{-1} = \tilde{A}^{-1}$

ii) $\tilde{A} \cdot \tilde{A}_p^{-1}$ is not defined for $M \neq N$

Back to the problem:

$$\tilde{A} \cdot \vec{x} = \vec{b} \quad (1) \quad | \cdot \tilde{A}^T$$ Apply the left on both sides

$$\Rightarrow \tilde{P} \cdot \vec{x} = \tilde{A}^T \cdot \vec{b} \equiv \vec{b}' \quad (2)$$

If (1) has a unique solution, then this is obviously also a solution of (2).

Remaining question: Is this solution also unique with respect to (2)?

Answer: yes

Because: $\tilde{P}$ is regular, as can be seen as follows.

We define: $$F(\vec{x}) \equiv \sum_i (\tilde{A}\vec{x} - \vec{b})_i^2 \quad (3)$$

This function has a global minimum at the solution $\vec{x}_0$ of (1), where

$$F(\vec{x}_0) = 0$$

Evaluation of (3):

$$(\tilde{A}\vec{x} - \vec{b})_i^2 = \left(\sum_j A_{ij} x_j - b_i \right)^2$$

$$= \sum_{j,j'} x_j A_{ij} A_{ij'} x_{j'} - 2 b_i \sum_j A_{ij} x_j + b_i^2$$

Summation over i:

$$F(\vec{x}) = \vec{x}^T \underbrace{\tilde{A}^T \tilde{A}}_{=\tilde{P}} \vec{x} - 2\, \vec{x}^T \cdot \underbrace{\tilde{A}^T \vec{b}}_{=\vec{b}'} + \vec{b}^2$$

$$= \vec{x}^T \tilde{P} \vec{x} - 2\, \vec{x}^T \vec{b}' + \vec{b}^2 \qquad (4)$$

If $\tilde{P}$ was not regular, there would be a homogeneous solution $\vec{x}_h$ of (2) for which

$$\tilde{P} \cdot \vec{x}_h = \vec{0}$$

$$\Rightarrow F(\vec{x}_0 + \lambda \vec{x}_h) \overset{(\tilde{P}^T = \tilde{P})}{=} 2\lambda\, \vec{x}_h^T \underbrace{(\tilde{P} \cdot \vec{x}_0 - \vec{b}')}_{=\vec{0}} + \lambda^2\, \vec{x}_h^T \underbrace{\tilde{P} \vec{x}_h}_{=\vec{0}}$$

$$= 0 \quad \forall \lambda$$

This contradicts the assumption that $\vec{x}_0$ is the global minimum of F

Thus:

i) If (1) has a unique solution $\vec{x}_0$ then is $\vec{x}_0$ the unique solution of (2).

ii) If we know (from whatever source, e.g. for physical reasons) that (1) has a unique solution, we can alternatively solve the smaller and quadratic problem (2) using the method from 2.1.1

iii) If (1) has no solution, then the solution (2) is the least bad in the sense of the function (3) ("least square minimum"), because with $\vec{x}_0$ being a solution of (2)

$$\vec{0} = \nabla_{\vec{x}} F(\vec{x})\Big|_{\vec{x} = \vec{x}_0} = 2(\tilde{P} \vec{x}_0 - \vec{b}') = \vec{0}$$

Alternative: Chapter 2.2

2.2 Cases I)–III): The singular value decomposition (SVD)

With the SVD we have a method with which we

i) can analyze which of the cases I)–III) applies

ii) can determine solutions/solution-spaces/least-square-minima

2.2.1 On mathematics

Theorem: Every (!) MxN matrix $\tilde{A}$ can be written as

$$\tilde{A} = \tilde{u} \cdot \tilde{w} \cdot \tilde{v}^T \qquad (1)$$

Here,

i) $\tilde{u}$ is an orthogonal MxM matrix, i.e.

$$\tilde{u} = (\vec{u}_1, \dots, \vec{u}_M) = \begin{pmatrix} \vec{u}_1' \\ \vdots \\ \vec{u}_M' \end{pmatrix}, \quad \vec{u}_i \cdot \vec{u}_j = \delta_{ij}, \quad \vec{u}_i' \cdot \vec{u}_j' = \dots$$

ii) $\tilde{v} = (\vec{v}_1, \dots \vec{v}_N)$ and hence $\tilde{v}^T$ is an orthogonal NxN-Matrix

iii) $\tilde{w}$ is an MxN matrix, for which

$$w_{ij} = w_i \, \delta_{ij}$$

where the "singular values" $w_i \geq 0$ are the square-roots of the eigenvalues of $\tilde{A}^T \cdot \tilde{A}$ $(M \geq N)$ or of $\tilde{A} \cdot \tilde{A}^T$ $(M \leq N)$

M>N :

$$\tilde{w} = \begin{pmatrix} w_1 & & \\ & \ddots & \\ & & w_N \\ 0 & \cdots & 0 \\ 0 & \cdots & 0 \end{pmatrix}$$

M<N:

$$\tilde{w} = \begin{pmatrix} w_1 & & & 0 & \cdots & 0 \\ & \ddots & & \vdots & & \vdots \\ & & w_M & 0 & \cdots & 0 \end{pmatrix}$$

Remarks:

i) The right-hand side of (1) is unique, except for

- any (orthogonality-preserving) linear combinations of $\{\vec{u}_i, \vec{u}_j\}$ or $\{\vec{v}_i, \vec{v}_j\}$ $\quad w_i = w_j$
- simultaneous permutation of $\vec{u}_i, w_i, \vec{v}_i$

  e.g. $\vec{u}_1 \leftrightarrow \vec{u}_3$ in $\tilde{u}$

  $w_1 \leftrightarrow w_3$ in $\tilde{w}$

  $\vec{v}_1 \leftrightarrow \vec{v}_3$ in $\tilde{v}$

ii) Algorithms for determining the SVD: Ref. 2 or 4.
We rely on LAPACK, Eigen, etc.

Examples:

i) $M>N$

$$\tilde{A} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{pmatrix} \Rightarrow \tilde{u} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & -1 \\ 0 & \sqrt{2} & 0 \\ 1 & 0 & 1 \end{pmatrix}, \quad \tilde{w} = \begin{pmatrix} \sqrt{2} & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix}, \quad \tilde{v} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

ii) $M<N$

$$\tilde{A} = \begin{pmatrix} 0 & 0 & 1 \\ 2 & 2 & 0 \end{pmatrix} \Rightarrow \tilde{u} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad \tilde{w} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \sqrt{2} & 0 \end{pmatrix}, \quad \tilde{v} = \begin{pmatrix} 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ 0 & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ 1 & 0 & 0 \end{pmatrix}$$

iii) $M=N$

$$\tilde{A} = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \quad \tilde{u} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}, \quad \tilde{w} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad \tilde{v} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

So: Even for M=N, the SVD has nothing to do with the diagonalization of $\tilde{A}$

<u>Proof</u> of the theorem (for simplicity, for M=N)

i) $\tilde{H} := \tilde{A}^T \tilde{A}$ (or $\tilde{A}\tilde{A}^T$) is symmetric $\Rightarrow$ eigenvalues $\lambda \in \mathbb{R}$

Let $\vec{v}_i$ be an eigenvector of $\tilde{H}$ with the eigenvalue λ_i. Then

$$\underbrace{\vec{v}_i^T \cdot \tilde{A}^T \cdot \tilde{A} \cdot \vec{v}_i}_{= (\tilde{A}\vec{v}_i)^T \geq 0} = \lambda_i \underbrace{\vec{v}_i^T \cdot \vec{v}_i}_{\geq 0} \Rightarrow \lambda_i \geq 0 \Rightarrow w_i \equiv \sqrt{\lambda_i} \in \mathbb{R}, \geq 0$$

$\hat{H}$ has N orthogonal eigenvectors $\hat{H}\vec{v}_i = \lambda_i \vec{v}_i$

and we define the matrices

$$\tilde{V} = (\vec{v}_1, \dots, \vec{v}_N)$$

and

$$\tilde{W} = \begin{pmatrix} w_1 & & \\ & \ddots & \\ & & w_N \end{pmatrix}$$

W.L.O.G.: $w_i \neq 0$ for $i \leq r$, $w_i = 0$ for $i > r$ } $r = 0$ or $r = N$ is also possible

ii) What is $\tilde{U}$?

For all $i \leq r$ we define vectors $\vec{u}_i \equiv \frac{\hat{A} \cdot \vec{v}_i}{w_i}$ (2)

These are $\neq \vec{0}$ and orthonormal, because

$$\vec{u}_i^T \cdot \vec{u}_j = \frac{\vec{v}_i^T \hat{A}^T \hat{A} \vec{v}_j}{w_i \cdot w_j} = \frac{w_j}{w_i w_j} \underbrace{\vec{v}_i^T \cdot \vec{v}_j}_{= \delta_{ij}} = \delta_{ij} \quad \checkmark$$

The vectors $\vec{u}_{r+1}, \dots, \vec{u}_N$ can then be chosen arbitrarily such that $\vec{u}_i \cdot \vec{u}_j = \delta_{ij} \; \forall \; i,j$

$\Rightarrow \tilde{U} \equiv (\vec{u}_1, \dots, \vec{u}_N)$

iii) still to be shown:

$\hat{A} = \tilde{U} \tilde{W} \tilde{V}^T$ or easier $\tilde{W} = \tilde{U}^T \cdot \hat{A} \cdot \tilde{V}$

$\hat{A} \cdot \tilde{V} = (\hat{A}\vec{v}_1, \dots, \hat{A}\vec{v}_N)$

where

- $\hat{A} \cdot \vec{v}_i \overset{(2)}{=} w_i \vec{u}_i$ if $i \leq r$
- $\hat{A} \cdot \vec{v}_i = \vec{0}$ " $i > r$

because $(\hat{A}\vec{v}_i) \cdot (\hat{A}\vec{v}_i) = \vec{v}_i^T \hat{A}^T \hat{A} \vec{v}_i = w_i^2 \underbrace{\vec{v}_i^T \vec{v}_i}_{=0} = 0$ for $i > r$

$\Rightarrow \hat{A} \cdot \vec{v}_i = 0 \; \forall \; i > r$

So

$$\left(\tilde{u}^T \cdot \tilde{A}\, \tilde{v}\right)_{i,j} = \left[\begin{pmatrix} \vec{u}_1 \\ \vdots \\ \vec{u}_N \end{pmatrix} \cdot \left(w_1 \vec{u}_1, \dots, w_N \vec{u}_N\right)\right]_{i,j} = w_i \vec{u}_i \cdot \vec{u}_j$$

$$= w_i \delta_{i,j} = \left(\tilde{w}\right)_{i,j} \quad \checkmark$$

2.2.2 Reminder: Linear mappings

An MxN matrix $\tilde{A}$ defines a linear mapping

$$f: \mathbb{R}^N \to \mathbb{R}^M$$

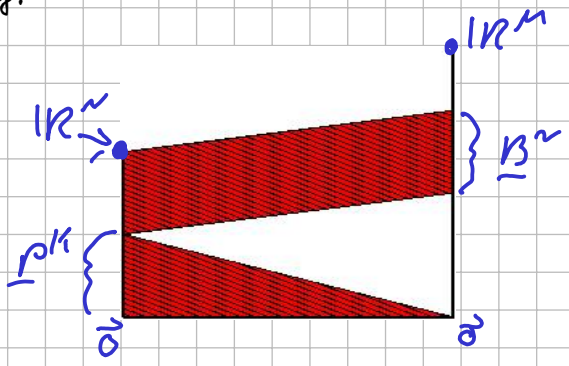
A) Important statements and terms

i) The image of f (or $\tilde{A}$) is an r-dimensional subspace $\underline{B}^r$ or $\mathbb{R}^M$. r is called the "rank" of the linear mapping (or of $\tilde{A}$).

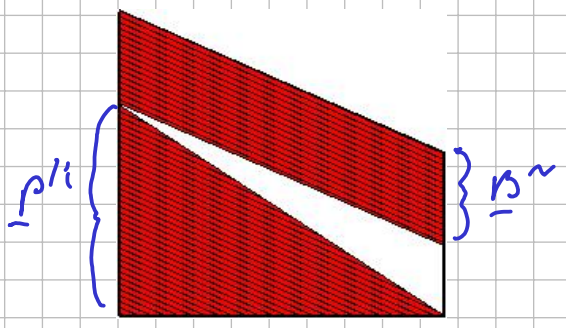
ii) The k-dimensional subspace $\underline{P}^k$ of vectors $\vec{x}$ in $\mathbb{R}^N$ for which $\tilde{A} \cdot \vec{x} = \vec{0}$ is called the "kernel" of the mapping. Its dimension k is called the "nullity".

iii) Pictorially:

$M > N$



$M < N$



iv) It applies $r + k = N$ $\quad (\Rightarrow r = N - k)$

Since $r \leq m \Rightarrow n - k \leq m \Rightarrow k \geq n - m$

For M<N, k is always>0

B) Application to LSE::

An LSE $\tilde{A} \cdot \vec{x} = \vec{b}$ has

i) no solution if $\vec{b} \notin \underline{B}^r$

ii) at least one solution if $\vec{b} \in \underline{B}^r$. This is unique if and only if k=0.

iii) If $\vec{x}_0$ is a solution, then also $\vec{x}_0 + \vec{x}_H$ with an arbitrary $\vec{x}_H \in \underline{P}^k$ ($\vec{x}_H$: homogeneous solution)

2.2.3 Determination of $\underline{B}^r$ and $\underline{P}^H$ with the SVD

A) The columns $\vec{u}_i$ of $\tilde{u}$, with $w_i \neq 0$ form an orthonormal basis (ONB) of $\underline{B}^r$, because:

We use the shorthand notation

$$\underline{B}^r = \tilde{A} \cdot \mathbb{R}^n = \text{span}(\tilde{A}\vec{e}_1, \ldots, \tilde{A}\vec{e}_n)$$

($\vec{e}_i$: cartesian basis)

$$\equiv \text{space spanned by } \tilde{A}\vec{e}_1, \ldots, \tilde{A}\vec{e}_n$$

Then

$$\underline{B}^r = \tilde{u}\tilde{w}\underbrace{\tilde{v}^T \cdot \mathbb{R}^n}_{= \mathbb{R}^n \text{ (da } \tilde{v}^T \text{ orthogonal)}}$$

So the question remains: What is the image of $\tilde{u} \cdot \tilde{w}$?

i) $m \geq n$

$$\tilde{u} \cdot \tilde{w} = (\vec{u}_1, \ldots, \vec{u}_m) \begin{pmatrix} w_1 & & & \\ & \ddots & & \\ & & w_r & \\ & & 0 & \\ & & \vdots & \\ & & 0 & \end{pmatrix} \quad w_{r+1}, \ldots w_n$$

$$(w_1\vec{u}_1, \ldots, w_r\vec{u}_r, \vec{0}, \ldots, \vec{0}) \Rightarrow \text{claim}$$

ii) $m < n$

$$\tilde{u} \cdot \tilde{w} = (\vec{u}_1 \dots \vec{u}_M) \begin{pmatrix} w_1 & & & 0 \\ & \ddots & & \vdots \\ & & w_r & \\ 0 & \dots & 0 & \vdots \end{pmatrix}$$

$[w_1 \vec{u}_1, \dots w_r \vec{u}_r, \vec{0}, \dots, \vec{0}] \Rightarrow$ lin. ind.

B) The columns $\vec{v}_i$ of $\tilde{V}$ for which $w_i = 0$ form an ONB of $\underline{P}^K$ because:

i) $M \leq N$

$$\tilde{w} \cdot \tilde{V}^T = \tilde{w} (\vec{v}_1, \dots, \vec{v}_N)^T$$
$$= (w_1 \vec{v}_1, \dots, w_M \vec{v}_M)^T$$

[$\vec{v}_{M+1}, \dots \vec{v}_N$ have disappeared from here]

so

$$(\tilde{w} \cdot \tilde{V}^T) \cdot \vec{v}_i = \begin{pmatrix} w_1 \vec{v}_1 \cdot \vec{v}_i \\ \vdots \\ w_i \vec{v}_i \cdot \vec{v}_i \\ \vdots \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ w_i \\ 0 \\ \vdots \end{pmatrix}$$

($\vec{v}_1 \cdot \vec{v}_i = 0$)

$$= w_i \vec{v}_i = \begin{cases} \vec{0} & \text{falls } w_i = 0 \Rightarrow \vec{v}_i \in \underline{P}^K \\ \neq 0 & \text{" } w_i \neq 0 \Rightarrow \vec{v}_i \notin \underline{P}^K \end{cases}$$

ii) $M > N$: analogous

2.2.4 Analysis of linear systems of equations

A) M=N:

i) Case 1: $w_i \neq 0 \quad \forall \; i$

$\Rightarrow \tilde{A} \vec{x} = \vec{s}$ has the unique solution

$$\vec{x}_0 = \tilde{A}^{-1} \cdot \vec{s} = \tilde{V} \cdot (\tilde{w}^{-1} \cdot (\tilde{u}^T \cdot \vec{s})) \quad (1)$$

with

$$\tilde{w}^{-1} = \begin{pmatrix} 1/w_1 & & 0 \\ & \ddots & \\ 0 & & 1/w_N \end{pmatrix}$$

Note: In (1) we are dealing with $\sigma(n^3)$ operations. It therefore makes no sense to solve the SLE with another method after the SVD analysis.

ii) Whether $\exists\, w_i = 0$ and $\vec{b} \in \underline{B}^r \Rightarrow \exists\, \infty$ many solutions
Whether $\vec{b} \in \underline{B}^r$ can be easily checked: With $\vec{u}_i$ $(i = 1, \dots, r)$ being a basis of $\underline{B}^r$ one checks, if

$$\vec{b} - \sum_{i=1}^{r} (\vec{b} \cdot \vec{u}_i)\, \vec{u}_i = \vec{0}$$

(1) is also a solution here if $\tilde{w}^{-1}$ ist set to $\frac{1}{0} = 0$ namely the solution with minimum $|\vec{x}|$.

Proof: That it is a solution can be shown as in B)

to be shown: $|\vec{x}_0 + \underbrace{\delta\vec{x}}_{\in \underline{P}^R \text{ s.t. } \delta\vec{x} = \sum_i \alpha_i \vec{v}_i, \text{ where } \alpha_i = 0, \text{ when } w_i \neq 0}| \geq |\vec{x}_0|$

$$\overset{(1)}{=} |\tilde{V}\tilde{w}^{-1}\tilde{U}^T\vec{b} + \delta\vec{x}| = |\tilde{V}(\tilde{w}^{-1}\tilde{U}^T\vec{b} + \tilde{V}^T\cdot\delta\vec{x})|$$

($\tilde{V}$ orthogonal) $\overset{\geq}{=} |\underbrace{\tilde{w}^{-1}\tilde{U}^T\cdot\vec{b}}_{=\vec{L}} + \underbrace{\tilde{V}^T\cdot\delta\vec{x}}_{=\vec{h}}|$

$$\vec{h} = \begin{pmatrix} \vec{v}_1^T \\ \vdots \\ \vec{v}_N^T \end{pmatrix} \cdot \sum_i \alpha_i \vec{v}_i = \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_N \end{pmatrix}$$

so: $h_i \neq 0$ only, when $w_i = 0$
$L_i \neq 0$ " $w_i \neq 0$

$\Rightarrow |\underbrace{\vec{L} + \vec{h}}_{= |\vec{x}_0 + \delta\vec{x}|}| \geq |\underbrace{\vec{L}}_{= |\vec{x}_0|}|$ ✓

iii) Case 3: $\exists\, w_i \neq 0$ und $\vec{b} \notin \underline{B}^r \Rightarrow \exists$ no solution
In this case, (1) is the minimum of $|\tilde{A}\cdot\vec{x} - \vec{b}|$
$\Rightarrow$ again a least-square-minimum (lsm) as in 2.1.2.

Proof: maybe exercises

B) $M \neq N$

Analysis of the solution space and solution (or lsm) with (1) as in A)

Physically the most important case again: A seemingly over-determined LSE with a unique solution as in 2.1.2, i.e. here

i) $w_i \neq 0 \ \forall i \Rightarrow \underline{P}^{M} = \vec{0} \Rightarrow$ no hom. solution of the LSE

ii) $\vec{b} \in \underline{B}^{r}$

Then equation (1) applies again where

$$\tilde{w}^{-1} = \begin{pmatrix} 1/w_1 & & & 0 \\ & \ddots & & \vdots \\ & & 1/w_N & 0 \end{pmatrix} \Big\} N$$

M

Proof: Simply insert into SLE

$$\tilde{A}\vec{x}_0 = (\tilde{u}\cdot\tilde{w}\cdot\tilde{v}^T)(\tilde{v}\,\tilde{w}^{-1}\,\tilde{u}^T\vec{b})$$
$$= \tilde{u}\,\tilde{w}\,\tilde{w}^{-1}\,\tilde{u}^T\cdot\vec{b}$$

It is

$$\tilde{w}\cdot\tilde{w}^{-1} = \begin{pmatrix} \tilde{1}_{N\times N} & 0 \\ 0 & 0 \end{pmatrix}$$

in A) a matrix $\tilde{1}_{r\times r}$

N not M!

If $\vec{b} \in \underline{B}^r$ so is $\vec{b} = \sum_{i=1}^{r} \lambda_i \vec{u}_i$ and $\tilde{u}^T\vec{b}$ has non-vanishing components only in the first N rows. This means that $\tilde{w}\cdot\tilde{w}^{-1}$ has the effect of an MxM $\tilde{1}$ matrix

$$\Rightarrow \tilde{A}\vec{x}_0 = \tilde{u}\,\tilde{u}^T\cdot\vec{b} = \vec{b} \quad \checkmark$$

2.2.5 Further applications of SVD

A) Construction of orthonormal bases

Given N vectors $\vec{a}_i \ (i=1,..,N)$ in $\mathbb{R}^M$

These form an N'-dimensional subspace $\underline{A}^{N'}$ of $\mathbb{R}^M$

Wanted: An ONB of $\underline{A}^{N'}$

Textbook method: Gram–Schmidt method

→ Numerically very susceptible to rounding errors

much better: SVD of the matrix

$$\tilde{A} = (\vec{a}_1, \dots, \vec{a}_N) = \tilde{U} \cdot \tilde{W} \cdot \tilde{V}^T$$

Then the vectors $\vec{u}_i$ with $w_i \neq 0$ is the seeked ONB

Reason: The image of $\tilde{A}$ is exactly the space $\underline{A}^{N'}$ spanned by the $\vec{a}_i$ $\overset{3.2.3}{\Rightarrow}$ Bild.

B) Matrix approximation → data compression

Idea for M=N ($M \neq N$ is analogous):

$$a_{ij} = \sum_{k=1}^{N} w_k u_{ik} v_{jk} \qquad (w_{i+1} \leq w_i, \text{ w.l.o.g.})$$

i.e. $\tilde{A} = \sum_{k=1}^{N} w_k \vec{u}_k \vec{v}_k^T$

Obviously the largest contributions come from the large w_k

Approximation for $\tilde{A}$: Termination of the sum at $r<N$

$\Rightarrow$ Only the $\vec{u}_k, \vec{v}_k$ with $k<r$ must be stored

N=M=512 pixels with $a_{i,j}$ = grayscale

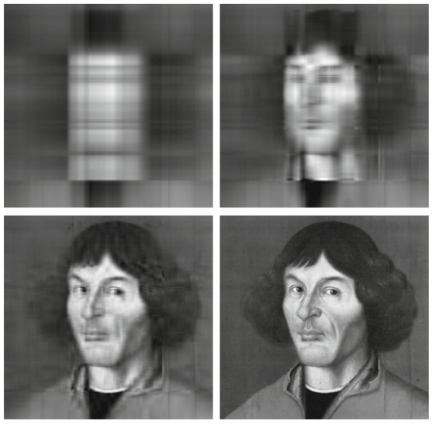


Fig. 3.7 Image compression by SVD. The original image is a 512 × 512 matrix of pixels with 256 gray levels. Shown are the approximations of rank 2 (99.6% compression), rank 5 (99.0%), rank 15 (97.1%), and rank 100 (80.4%). See Problem 3.10.4

C) Face recognition (maybe excercise)

3) Eigenvalues (EW) and eigenvectors (EV)

3.1 Mathematical repetition

Given: NxN matrix $\tilde{A}$

Wanted: All vectors $\vec{v} \in \mathbb{R}^N$ with

$$\tilde{A} \cdot \vec{v} = \lambda \cdot \vec{v} \quad , \lambda \in \mathbb{C}$$

$$\Rightarrow (\tilde{A} - \lambda \tilde{1}) \cdot \vec{v} = \vec{0}$$

$$\Rightarrow p(\lambda) = |\tilde{A} - \lambda \tilde{1}| \overset{!}{=} 0$$

Mathematical textbook method:

i) Determine roots λ_i of $p(\lambda)$

ii) Solve for each λ_i the SLE

$$\tilde{A} \vec{v}_i = \lambda_i \vec{v}_i$$

However, this is numerically inefficient.

Alternative: Find a similarity transformation $\tilde{Z}$,

$$\tilde{A} \to \tilde{A}' = \tilde{Z}^{-1} \cdot \tilde{A} \cdot \tilde{Z} \quad ,$$

such that

$$\tilde{A}' = \begin{pmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_N \end{pmatrix}$$

is diagonal. Then the λ_i are the EV of $\tilde{A}$ with eigenvectors

$$\vec{v}_i = \tilde{Z} \cdot \vec{l}_i \quad (= i\text{-th column of } \tilde{Z})$$

$\vec{l}_i$: cartesian basis vectors

because: $\tilde{A} \cdot \vec{v}_i = \tilde{Z} \cdot \tilde{Z}^{-1} \cdot \tilde{A} \cdot \tilde{Z} \cdot \vec{l}_i = \lambda_i \cdot \vec{v}_i$

$= \tilde{A}'$

$= \lambda_i \vec{l}_i$

We restrict ourselves to

i) real symmetric matrices $\tilde{A}^T = \tilde{A}$

$\Rightarrow \lambda_i \in \mathbb{R}$, $\tilde{Z}$ is orthogonal, $\vec{v}_i \cdot \vec{v}_j = 0$, if $\lambda_i \neq \lambda_j$

ii) Hermitian matrices $\tilde{A}^+ = \tilde{A}$

$\Rightarrow \lambda_i \in \mathbb{R}$, $\tilde{z}$ is unitary, $\vec{v}_i^{\,*} \vec{v}_j = 0$

Here, too, we can restrict ourselves to case i), because

$$\tilde{A}\cdot\vec{v} = \lambda\cdot\vec{v} \Rightarrow (\tilde{A}_R + i\tilde{A}_I)(\vec{v}_R + i\vec{v}_I) = \lambda(\vec{v}_R + i\vec{v}_I)$$

$$\Rightarrow \underbrace{\begin{pmatrix} \tilde{A}_R & -\tilde{A}_I \\ \tilde{A}_I & \tilde{A}_R \end{pmatrix}}_{\text{Symmetric, if}} \begin{pmatrix} \vec{v}_R \\ \vec{v}_I \end{pmatrix} = \lambda \begin{pmatrix} \vec{v}_R \\ \vec{v}_I \end{pmatrix}$$

$$\tilde{A}^+ = \tilde{A}_R^T - i\tilde{A}_I^T = \tilde{A}_R + i\tilde{A}_I = \tilde{A}$$

$$\Rightarrow -\tilde{A}_I = \tilde{A}_I^T \quad \checkmark$$

3.2 Jacobi rotation

Idea: Find the largest (absolute value) off-diagonal element $a_{i,j}$ and determine $\tilde{z}(i,j)$, such that

$$\tilde{A}' = \tilde{z}(i,j)^T \tilde{A}\, z(i,j)$$

$a'_{i,j} = 0$ is satisfied

Suitable approach:

$$z(i,j) = \begin{pmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & z_{i,i} & \cdots & z_{i,j} & \\ & & \vdots & & \vdots & \\ & & z_{j,i} & \cdots & z_{j,j} & \\ & & & & & \ddots \\ & & & & & & 1 \end{pmatrix}; \quad \begin{pmatrix} z_{i,i} & z_{i,j} \\ z_{j,i} & z_{j,j} \end{pmatrix} = \begin{pmatrix} \overbrace{\cos\theta}^{=c} & \sin\theta \\ \underbrace{-\sin\theta}_{=s} & \cos\theta \end{pmatrix}$$

$\Rightarrow \theta$ is to be determined:

$$\begin{pmatrix} c & -s \\ s & c \end{pmatrix} \begin{pmatrix} a_{ii} & a_{ij} \\ a_{ji} & a_{jj} \end{pmatrix} \begin{pmatrix} c & s \\ -s & c \end{pmatrix} \overset{!}{=} \begin{pmatrix} a'_{ii} & 0 \\ 0 & a'_{jj} \end{pmatrix}$$

$$\Rightarrow (c^2 - s^2)\, a_{ij} + s\cdot c\,(a_{ii} - a_{jj}) = 0$$

$$\Rightarrow \frac{1}{2}\frac{c^2 - s^2}{sc} = \frac{a_{jj} - a_{ii}}{2a_{ij}} \equiv \mathcal{R}$$

$(t = \tan\theta)$ $\rightarrow t = \frac{1}{2}\left(\frac{1}{t} - t\right)$ quadratic eq. for t or $\frac{1}{t}$

Solution: $t = \frac{1}{\Omega \pm \sqrt{1+\Omega^2}} \Rightarrow c = \frac{1}{\sqrt{1+t^2}}$, $s = c \cdot t$

To be clarified: Which sign?

Smallest intrusion, if $|\theta| \leq \frac{\pi}{4}$ $(\Rightarrow |t| \leq \frac{1}{2})$

since then $(a_{ii} - a'_{ii}) + (a_{jj} - a'_{jj})$ is smaller than with $\epsilon > \frac{\pi}{4}$

$\Rightarrow$ $\Omega > 0$, "+" sign
$\Omega < 0$, "−" sign
} compact $t = \frac{\text{sign}(\Omega)}{|\Omega| + \sqrt{1+\Omega^2}}$

This results in the following algorithm:

1) Choose i,j such that $|a_{ij}|$ takes a maximum
2) Calculate c and s using the above formulas
3) Transformation $\tilde{A}' = \tilde{T}(i,j)^T \cdot \tilde{A} \cdot \tilde{T}(i,j)$
   $\tilde{T} \rightarrow \tilde{T} \cdot \tilde{T}(i,j)$ (start with $\tilde{T} = \tilde{1}$)
4) Check whether
$$off(\tilde{A}') = \sum_{l \neq m} |a'_{lm}|^2$$
is below a certain limit, otherwise $\tilde{A}' \rightarrow \tilde{A}$ and back to 1).

Remarks:

Since $off(\tilde{A}') = off(\tilde{A}) - 2|a_{ij}|^2$ (1)

this value becomes smaller and smaller. Mathematically, this does of course not necessarily lead to convergence $\rightarrow 0$.

According to all experience, however, this is always the case (more on strict convergence proofs in Ref. 3)

Proof of (1):

a) It is $\tilde{A}' = \tilde{A}$ except in the rows/columns i and j.

b) $\sum_{l,m} a'^2_{lm} = Sp(\tilde{A}'\cdot\tilde{A}') = Tr(\tilde{A}\cdot\tilde{A}) = \sum_{l,m} a^2_{lm}$

Invariance of the trace under basis transformation

c) same argument as in b):

$$a^2_{ii} + a^2_{jj} + 2a^2_{ij} = (a'_{ii})^2 + (a'_{jj})^2$$

$$\Rightarrow off(\tilde{A}') = \sum_{l,m} (a'_{lm})^2 - \sum_l (a'_{ll})^2 \overset{b)}{=} off(\tilde{A}) + \sum_l a^2_{ll} - a'^2_{ll} \overset{a)\,c)}{=} -2a^2_{ij}$$ ✓

ii) For large N, step 1 is time consuming. Then better fixed run through i,j in a certain order

iii) In general, convergence is slow $\Rightarrow$ better 3.3

3.3 Diagonalization using the Housholder algorithm

The Housholder algorithm determines in N−2 steps an orthogonal $\tilde{Z}$, with which $\tilde{A}' = \tilde{Z}^T\tilde{A}\,\tilde{Z}$ has a tridiagonal form.

Step 1:

$$\tilde{Z}(1)^T\tilde{A}\,\tilde{Z}(1) = \begin{pmatrix} a_{11} & k_1 & 0 & \cdots & 0 \\ k_1 & & & & \\ 0 & & & \tilde{A}(1) & \\ \vdots & & & & \\ 0 & & & & \end{pmatrix}$$

$$\text{with } \tilde{A}'(1) = \begin{pmatrix} a'_{22} & \cdots & a'_{2N} \\ \vdots & & \\ a'_{N2} & \cdots & a'_{NN} \end{pmatrix}$$

Step 2: $\tilde{Z}(2)^T\tilde{Z}(1)^T\tilde{A}\,\tilde{Z}(1)\,\tilde{Z}(2)$

$$= \begin{pmatrix} a_{11} & k_1 & 0 & \cdots & 0 \\ k_1 & a'_{22} & k_2 & 0 & \cdots & 0 \\ 0 & & & & & \\ \vdots & k_2 & & \tilde{A}'(2) & & \\ 0 & \vdots & & & & \end{pmatrix} \quad \vdots \quad \tilde{A}'(2) = \begin{pmatrix} a'_{33} & \cdots \\ \vdots & \end{pmatrix}$$

usw. ⋮

Step N−2: tridiagonal form

Approach for $\tilde{z}(i)$:

$$\tilde{z}(i) = \begin{pmatrix} \tilde{1}_{i\times i} & \vdots & 0 \\ \cdots & \cdots & \cdots \\ 0 & \vdots & \tilde{S}(i) \end{pmatrix} \qquad (1)$$

with the $(N-i)\times(N-i) = n\times n$ matrix

$$\tilde{S}(i) \equiv \tilde{1}_{n\times n} - 2\,\vec{u}(i)\cdot\vec{u}(i)^T \qquad (2)$$

$\vec{u}(i)$: n-dimensional vector with $|\vec{u}(i)| = 1$ (3)

Check orthogonality of $\tilde{S}(i)$: (es ist $\tilde{S}(i)^T = \tilde{S}(i)$)

$$\Rightarrow \tilde{S}(i)^T\tilde{S}(i) = (\tilde{1} - 2\vec{u}\,\vec{u}^T)(1 - 2\vec{u}\,\vec{u}^T)$$
$$= \tilde{1} - 4\vec{u}\,\vec{u}^T + 4\vec{u}\,\underbrace{\vec{u}^T\vec{u}}_{(3)\,1}\,\vec{u}^T = \tilde{1}$$

Step i=1:

$$\tilde{z}(1) = \begin{pmatrix} 1 & \vdots & \vec{v} \\ \cdots & \cdots & \cdots \\ \vec{0} & \vdots & \tilde{S}(1) \end{pmatrix}$$

$$\Rightarrow \tilde{z}(1)\,\tilde{A}\,\tilde{z}(1) = \begin{pmatrix} a_{11} & \vdots & (\tilde{S}(1)\cdot\vec{v})^T \\ \cdots & \cdots & \cdots \\ \tilde{S}(1)\cdot\vec{v} & \vdots & \tilde{S}(1)\underbrace{\begin{pmatrix} a_{22} & \cdots \\ \vdots & a_{nn} \end{pmatrix}}_{\equiv \hat{A}'(1)}\tilde{S}(1) \end{pmatrix}$$

with $\vec{v} = \begin{pmatrix} a_{21} \\ a_{31} \\ \vdots \\ a_{n1} \end{pmatrix}$

Remaining question: How to choose $\vec{u}(1)$ so that

$$(\tilde{S}(1)\cdot\vec{v})^T = (k_1, 0, \ldots, 0)$$

We write $\vec{u}(1) = \dfrac{\vec{p}(1)}{|\vec{p}(1)|}$ (1) with arbitrary $|\vec{p}(1)|$

approach: $\vec{p}(1) = \vec{v} \pm |\vec{v}|\,\vec{e}_1$ (2) $\Rightarrow |\vec{p}(1)|^2 = 2(|\vec{v}|^2 \pm |\vec{v}|\,v_1)$ (3)

$\Rightarrow \tilde{S}(1)\cdot\vec{v} = \vec{v} - 2\vec{u}(1)(\vec{u}(1)\cdot\vec{v}) = 2\,\vec{p}(1)\cdot\vec{v}$ (4)

$$\overset{(4)-(9)}{=} \vec{v} - \frac{\vec{v}^2 \pm |\vec{v}|\, v_1}{\vec{v}^2 \pm |\vec{v}|\, v_1} \left(\vec{v} \pm |\vec{v}|\, \vec{e}_1\right) = \pm |\vec{v}|\, \vec{e}_1$$

therefore: Goal achieved

$$\left(\tilde{S}^{(1)} \cdot \vec{v}\right)^T = (v_1', 0, \ldots, 0)$$

$$\text{mit } v_1' = \pm |\vec{v}|$$

Choose the sign as in $-v_1 = -\alpha_{21}$ to avoid rounding errors in

$$v_1' = v_1 \pm |\vec{v}|$$

(for the conceivable case that $|v_1| \approx |\vec{v}|$)

Steps 2 to N−2 are completely analogous

$\Rightarrow$ tridiagonal matrix

$$\tilde{T} = \tilde{S}^{(N-2)} \cdots \tilde{S}^{(1)}\, \tilde{A}\, \tilde{S}^{(1)} \cdots \tilde{S}^{(N-2)}$$

Caution: When calculating $\tilde{S}^{(1)}\, \tilde{A}\, \tilde{S}^{(1)}$ don't multiply the matrices (but for example)

$$\left(\vec{u} \cdot \vec{u}^T\right) \tilde{A} \left(\vec{u} \cdot \vec{u}^T\right) = \vec{u} \left[\vec{u}^T \underbrace{\left(\tilde{A}\, \vec{u}\right)}_{\text{vector}}\right] \vec{u}^T$$

number

After Tridiagonalising of $\tilde{A}$, one can diagonalise it with N−1 further Jacobi rotations.

$\Rightarrow$ Goal is achieved

3.4 Matrix diagonalization (MD) in quantum mechanics

3.4.1 Basic idea

In quantum mechanics, MD occurs when

i) the Hilbert space is finite by definition, e.g. the "Heisenberg model"

$$\hat{H} = \sum_{i,j} J_{ij}\, \hat{\vec{S}}_i \cdot \hat{\vec{S}}_j \quad ; \quad \hat{\vec{S}}_i = \begin{pmatrix} \hat{S}_{i,x} \\ \hat{S}_{i,y} \\ \hat{S}_{i,z} \end{pmatrix}$$

on a finite chain with L sites (i, j).

Then you can choose a basis $|\varphi_l\rangle$ of the Hilbert space and and diagonalise $H_{l,l'} = \langle \varphi_l | \hat{H} | \varphi_{l'} \rangle$.

ii) $\hat{H}$ is approximately diagonalised in a properly chosen sub-space with a finite Basis $|\varphi_i\rangle$, e.g. a H_2-Molekül

$$\hat{H} = \sum_{i=1}^{L} \hat{H}_{0,i} + \hat{V}_L$$

with $\hat{H}_{0,i} = \frac{\hat{\vec{p}}_i^{\,2}}{2m} - \frac{e^2}{|\vec{R}_i - \hat{\vec{r}}_i|}$ $\left(\vec{R}_i \in \mathbb{R}^3: \text{Nuclei position}, i: \text{num}\right)$

$$\hat{V}_L = \frac{e^2}{|\hat{\vec{r}}_1 - \hat{\vec{r}}_2|} - \frac{e^2}{|\vec{R}_1 - \hat{\vec{r}}_2|} - \frac{e^2}{|\vec{R}_2 - \hat{\vec{r}}_1|}$$

eigenstates $\varphi_{i,m}(\vec{r}_i)$ with eigenvalues E_m $(m = 1, \dots, \infty$ of $\hat{H}_{0,i}$ are known (->Physics IV or Theoretical Physics II)

⇒ Form from the N low-energy states

$\varphi_{1,m}(\vec{r}_1)$, $\varphi_{2,m}(\vec{r}_2)$ $(m = 1, \dots, N)$

two-electron states either in 1st quantization (Slater determinants) or in 2nd quantization (Fock states)

⇒ Diagonalize $\hat{H}$ in this subspace

⇒ Approximation for the low-energy spectrum of $\hat{H}$

3.4.2 Symmetries

Of decisive importance is the skillful choice of the base $|\varphi_i\rangle$

Example: Heisenberg model (with spin $S = \frac{1}{2}$), L lattice sites

basis of the Hilbert space: $|\vec{\sigma}\rangle = |\sigma_1, \sigma_2, \dots, \sigma_L\rangle$ with

$\sigma_i = \uparrow$ or $\downarrow$

$\hat{H}$ commutes with $\hat{S}^z = \sum_i \hat{S}_i^z$

$$\hat{\vec{S}}^2 = \left(\sum_i \hat{\vec{S}}_i\right)^2 \quad (\text{total spin})$$

The states $|\vec{\sigma}\rangle$ are eigenstates of $\hat{S}^z$

$$\hat{S}^z |\vec{\sigma}\rangle = \left(\sum_i \sigma_i\right) |\vec{\sigma}\rangle \equiv S_z(\vec{\sigma}) |\vec{\sigma}\rangle$$

$$\Rightarrow \langle \vec{\sigma} | [\hat{H}, \hat{S}^z] | \vec{\sigma}' \rangle = 0$$

$$= \left(S_z(\vec{\sigma}) - S_z(\vec{\sigma}')\right) \langle \vec{\sigma} | \hat{H} | \vec{\sigma}' \rangle$$

$$\rightarrow \langle \vec{\sigma} | \hat{H} | \vec{\sigma}' \rangle = 0 \quad \text{if} \quad S_z(\vec{\sigma}) \neq S_z(\vec{\sigma}')$$

thus: $\langle \vec{\sigma} | \hat{H} | \vec{\sigma}' \rangle$ is block-diagonal

$\Rightarrow$ Diagonalization is only required in the subspace with fixed S_z

- dimension of the entire Hilbert space: $D_H = 2^L$
- dimension of the subspace with fixed S_z

  Be N_σ the number of places with spin σ

$$\Rightarrow \left.\begin{aligned} L &= N_\uparrow + N_\downarrow \\ S_z &= \tfrac{1}{2}(N_\uparrow - N_\downarrow) \end{aligned}\right\} \Rightarrow N_\uparrow = \tfrac{1}{2}(L + 2S_z)$$

$$\Rightarrow D_{un}(S_z) = \binom{L}{N_\uparrow} = \binom{L}{\frac{1}{2}(L+2S_z)}$$

= number of possibilities to distribute the $N_\uparrow$ $\uparrow$ spins

e.g.: L=10 $\Rightarrow$ $D_H = 1024$

$D_{un}(10) = 252$

Question: In general, how can an optimal base (i.e. with maximum block diagonality of H_{ij}) be found?

$\Rightarrow$ Group theory (summer semester 2026)

3.5 The power method and the Lanczos algorithm

Hamiltonian matrices are often sparse (given a suitable choice of basis).

Example: Spin 1/2 Heisenberg model with nearest neighbor

coupling, we use ascent and descent operators

$$\hat{H} = J \sum_{\langle i,j \rangle} \left[\hat{S}_i^z \hat{S}_j^z + \frac{1}{2} \left(\hat{S}_i^+ \hat{S}_j^- + \hat{S}_i^- \hat{S}_j^+ \right) \right]$$

$\langle i,j \rangle$ = nearest neighbors

Base as above: States $|\vec{\sigma}\rangle$

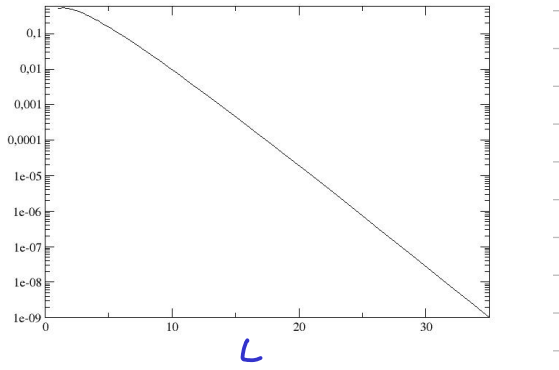
Each state $\langle \vec{\sigma}|$ has matrix elements $\neq 0$ only with $|\vec{\sigma}\rangle$, and with $|\vec{\sigma}'\rangle$ whose spins are flipped at exactly 2 neighboring sites

so: # of matrix elements $\neq 0$ per row in $\hat{H}$: $\approx L$

$\Rightarrow$ # of matrix elements $\neq 0$ in $\tilde{H}$: $\approx L \cdot 2^L$

# of matrix elements: $(2^L)^2$

$\Rightarrow$ Relation $\frac{L \cdot 2^L}{(2^L)^2} = \frac{L}{2^L}$



To analyze the spectra of very large sparse matrices we consider two methods:

3.5.1 Power method

Algorithm for determining the largest eigenvalue (plus the associated eigenvector) of $\tilde{A}$

i) Choose a starting vector $\vec{v}_0$

ii) Iterate $\vec{w}_m = \tilde{A}\, \vec{v}_{m-1}$ and $\vec{v}_m = \frac{\vec{w}_m}{|\vec{w}_m|} = \frac{\tilde{A}^m \vec{v}_0}{|\tilde{A}^m \vec{v}_0|}$

converges to the searched eigenvector $\vec{x}_N$ if

$\vec{v}_0 \cdot \vec{x}_N \neq 0$ and λ_N is not degenerate

Proof:

Let $\vec{x}_i(\lambda_i)$ be the eigenvectors (eigenvalues) of $\tilde{A}$

$(\lambda_1 \leq \lambda_2 \cdots \leq \lambda_N)$

Expand:

$$\vec{v}_0 = \sum_{i=1}^{N} \alpha_i \vec{x}_i$$

$$\Rightarrow \tilde{A}^m \vec{v}_0 = \sum_{i=1}^{N} \alpha_i \lambda_i^m \vec{x}_i = \lambda_N^m \left(\alpha_N \vec{x}_N + \sum_{i=1}^{N-1} \alpha_i \left(\frac{\lambda_i}{\lambda_N} \right)^m \vec{x}_i \right)$$

$$\Rightarrow \frac{\tilde{A}^m \vec{v}_0}{|\tilde{A}^m \vec{v}_0|} = \vec{x}_N + \underbrace{O\left(\left(\frac{\lambda_{N-1}}{\lambda_N} \right)^m \right)}_{m \to \infty \to 0}$$

Remarks:

i) $\tilde{A} \to -\tilde{A} \Rightarrow$ smallest eigenvalue of $\tilde{A}$ with the same method

ii) The numerical effort is low ($O(m \cdot N^2)$) for sparse matrices still much smaller, but:

Convergence is often very slow, especially when

$$\left| \frac{\lambda_{N-1}}{\lambda_N} \right| \lesssim 1$$

iii) With $\lambda_N, \vec{x}_N$ found, one can define

$$\tilde{A}' = \tilde{A} - \lambda_N \vec{x}_N \cdot \vec{x}_N^T$$

$$\Rightarrow \tilde{A}' \cdot \vec{x}_N = 0 \cdot \vec{x}_N$$

With the same method one can determine $\lambda_{N-1}, \vec{x}_{N-1}$ and in principle all $\lambda_i, \vec{x}_i$

3.5.2 Krylov spaces

Wanted: Ground state $\vec{v}_0$ of $\tilde{A}$ (NxN-matrix, $\tilde{A}^T = \tilde{A}$, $\vec{v}_0 \in \mathbb{R}^N$)

Principle way: Search for the minimum of

$$R(\vec{v}) = \frac{\vec{v}^T \tilde{A} \vec{v}}{\vec{v}^T \cdot \vec{v}}, \quad \lambda_0 = \min [R(\vec{v})] = R(\vec{v}_0)$$

However, if performed exactly, this minimization would be numerically even more complex than the initial problem. The aim will be the minimization in subspaces

$$\underline{U}_k := [\vec{p}_1, \dots, \vec{p}_k) = \mathrm{Span}(\vec{p}_1, \dots, \vec{p}_k)$$

of the $\mathbb{R}^N$

$$\vec{v}(\vec{\alpha}) := \sum_{i=1}^{k} \alpha_i \vec{p}_i \qquad \vec{\alpha} = \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_k \end{pmatrix}$$

$$w_k = \min_{\vec{\alpha}} f_k(\vec{\alpha}) = \min_{\vec{\alpha}} \frac{\vec{v}(\vec{\alpha})^T A \vec{v}(\vec{\alpha})}{\vec{v}(\vec{\alpha})^2} \Rightarrow \alpha_{min}$$

$$\Rightarrow \vec{v}^{(k)} := \vec{v}(\alpha_{min}) = \text{optimum } \vec{v} \text{ in } \underline{U}_k$$

Obviously is

$$w_1 \geq w_2 \geq \dots \geq w_k \geq \lambda_0$$

Thus: By increasing k, one approaches the exact minimum of $f(\vec{v})$

Central question: How to choose $\vec{p}_1, \dots, \vec{p}_k$, so that w_k is already a good approximation for λ_0 even when $k \ll N$

Inset:

anticipation of chapter 5: Minimization of functions $f(\vec{x})$

Simplest solution:

1) Choose a starting point $\vec{x}_0$
2) Calculate $\vec{F}(\vec{x}_0) := \vec{\nabla} f \big|_{\vec{x} = \vec{x}_0}$
3) Minimize $f(\vec{x}_0 - s \vec{F}(\vec{x}_0))$ (i.e. in the direction $-\vec{F}(\vec{x}_0)$ with respect to s
4) Back to 2)

Back to the initial question:

Let $\vec{v}^k$ be the minimum of $f_k(\vec{\alpha})$. How do we then choose meaningfully $\vec{p}_{k+1}$ (i.e. $\underline{U}_{k+1}$)

Answer: Such that $\pm \vec{\nabla} f(\vec{v}) \big|_{\vec{v} = \vec{v}^k} \in \mathrm{Span}[\underline{U}_k, \vec{p}_{k+1})$,

because in this direction we should ideally change $\vec{v}^k$

It is

$$\vec{\nabla} \mathcal{R}(\vec{v})\Big|_{\vec{v}=\vec{v}_n} \overset{\tilde{A}^T=\tilde{A}}{=} \frac{2}{(\vec{v}_n)^2}\left(\tilde{A}\cdot\vec{v}^{(n)} - \underbrace{\mathcal{R}(\vec{v}_n)\cdot\vec{v}^{n}}_{\in \mathcal{U}_n}\right)$$

contribution from the denominator

$\Rightarrow$ Choose $\vec{p}_{n+1} \sim \tilde{A}\cdot\vec{v}^{(n)}$ (1)

So:

1) Start with (random) $\vec{p}_1$
2) $\vec{p}_2 = \tilde{A}\cdot\vec{p}_1 \Rightarrow \vec{v}^{(2)} = \lambda_1\vec{p}_1 + \lambda_2\vec{p}_2$ (according to the current idea by minimizing $\mathcal{R}_2(\vec{\lambda})$)
3) (1) $\Rightarrow \vec{p}_3 = \tilde{A}\cdot\vec{v}^{(2)} = \lambda_1\cdot\tilde{A}\vec{p}_1 + \lambda_2 A\vec{p}_2$
$= \lambda_1\vec{p}_2 + \lambda_2\tilde{A}\vec{p}_2$

not required

but $\lambda_1\cdot\vec{p}_2$ is already in $\underline{\mathcal{U}}_2$

$\Rightarrow$ Instead of $\vec{p}_3 = \tilde{A}\cdot\vec{v}^{(2)}$ we can also choose $\vec{p}_3 = \tilde{A}\cdot\vec{p}_2$

$\Rightarrow$ The exact form of the optimal $\vec{v}^{(2)}$ is therefore irrelevant for the iteration.

4) analogous for arbitrary k $\vec{p}_{k+1} = \tilde{A}\cdot\vec{p}_k$

The space

$$\underline{\mathcal{U}}_n = \mathrm{Span}(\vec{p}_1, \vec{p}_2, \dots \vec{p}_n) = \mathrm{Span}(\vec{p}_1, \tilde{A}\vec{p}_1, \dots \tilde{A}^{n-1}\vec{p}_1)$$

is called "Krylov subspace".

3.5.3 The Lanczos method

Possible strategy:

Determine a Krylov space of sufficient size, construct the matrix $K_{ij} = \vec{p}_i^{\,T}\tilde{A}\cdot\vec{p}_j$ and determine its ground state

$\Rightarrow$ approximation for λ_0

Problems:

i) The vectors $\vec{p}_i$ are generally not orthogonal

ii) It is generally not even guaranteed that they are linearly independent $\Rightarrow$ Dimension of $\underline{\mathcal{U}}_n$ possibly $< k$

So: after the construction of a new $\vec{p}_{k+1}$ it is besser to ortho-normalise ($\vec{p}_{k+1} \to \vec{q}_{k+1}$) and abort, if $\dim(q_{k+1}) = k < k+1$

Problem: SVD does not make sense here, because

i) $\tilde{A}$ would have been in the base of the $\vec{q}_i$ few vanishing matrix elements (unlike the Lanczos method)

ii) A full SVD would have to be calculated for each new vector $\vec{p}_{k+1}$ i.e. without taking advantage of the fact that already $\vec{q}_i \cdot \vec{q}_j = \delta_{i,j} \quad \forall\ i,j \le k$

The following algorithm is better:

(use bra-ket notation, i.e. $\vec{q}_i \to |q_i\rangle$, $\tilde{A} \to \hat{A}$
$\vec{q}_i^T \hat{A} \vec{q}_i \to \langle \hat{A} \rangle_{q_i}$)

starting values and vectors: $|q_0\rangle = 0$

$|q_1\rangle = |q\rangle$ (arbitrary with $\langle q|q\rangle = 1$)

$\gamma_1 = 1$

Iterations: $|q_{i+1}\rangle = \gamma_{i+1}^{-1} \left[(\hat{A} - \delta_i \hat{1}) |q_i\rangle - \gamma_i |q_{i-1}\rangle \right]$ (1)

$\equiv \gamma_{i+1}^{-1} |v_i\rangle$

with $\delta_i \equiv \langle \hat{A} \rangle_{q_i}$ (2)

$\gamma_{i+1} \equiv \sqrt{\langle v_i | v_i \rangle}$ (3)

Abort the iteration if $\gamma_{i+1} = 0$ (i.e. when the dimension of the Krylov space no longer increases) or if the accuracy appears sufficient.

Proof of orthonormality:

i) Normalization: clear because of (3)

ii) Orthogonality by induction

A) Start of induction

i=2: $\langle q_0 | q_2 \rangle = 0$ if $|q_0\rangle = 0$ ✓

$$\gamma_2 |q_2\rangle = (\hat{A} - \delta_1 \hat{1}) |q_1\rangle = |v_2\rangle \quad (4)$$

$$\Rightarrow \langle q_1 | q_2 \rangle \sim (\langle A \rangle_{q_1} - \delta_1) \overset{(2)}{=} 0 \quad \checkmark$$

$$\text{and} \quad \gamma_2^2 = \langle \hat{A}^2 \rangle_{q_1} - \langle \hat{A} \rangle_{q_1}^2 \quad (3)$$

i=3: $\gamma_3 |q_3\rangle = (\hat{A} - \delta_2 \hat{1}) |q_2\rangle - \gamma_2 |q_1\rangle$

$$\Rightarrow \langle q_2 | q_3 \rangle \sim (\langle A \rangle_{q_2} - \delta_2) \overset{(2)}{=} 0$$

$$\langle q_1 | q_3 \rangle \sim \left(\underbrace{\langle q_1 | \hat{A} | q_2 \rangle}_{\overset{(4)}{=} \frac{1}{\gamma_2} \langle q_1 | \hat{A}(\hat{A} - \delta_1 \hat{1}) | q_1 \rangle = \gamma_2^2} - \gamma_2 \right) = 0 \quad (*)$$

B) Induction:

Let it now be $\langle q_i | q_j \rangle = \delta_{i,j} \quad \forall \; i,j \leq l$

too show: $\langle q_i | q_{l+1} \rangle = 0 \quad \forall \; i \leq l$

a) $\langle q_l | q_{l+1} \rangle = \langle q_{l-1} | q_{l+1} \rangle = 0$ as in (*)

b) $\langle q_i | q_{l+1} \rangle$ for $i < l-1$

$$\overset{(1)}{=} \gamma_{l+1}^{-1} \left[\underbrace{\langle q_i | \hat{A} | q_l \rangle}_{\overset{(5)}{=} \langle q_l | \hat{A} | q_i \rangle} - \delta_l \underbrace{\langle q_i | q_l \rangle}_{=0} - \gamma_l \underbrace{\langle q_i | q_{l-1} \rangle}_{=0} \right)$$

$$\hat{A}^T = \hat{A} \quad \overset{(5)}{=} \langle q_l | \hat{A} | q_i \rangle \overset{(1)}{=} \gamma_i \underbrace{\langle q_l | q_{i-1} \rangle}_{=0} + \delta_i \underbrace{\langle q_l | q_i \rangle}_{=1} + \gamma_{i+1} \underbrace{\langle q_l | q_{i+1} \rangle}_{=0}$$

$= 0$ (5) $\checkmark$

In the subspace $\underline{U}_K$ and in the base of the $|q_i\rangle$ is (due to (5))

$\tilde{A}$ tridiagonal and has the matrix elements

$\tilde{T}_{i,i} \equiv \langle q_i | \hat{A} | q_i \rangle \overset{(2)}{=} \delta_i$

$\tilde{T}_{(i-1),i} = \underbrace{\langle q_{i-1} | \hat{A} | q_i \rangle}_{} = \gamma_i$

$$\stackrel{(2)}{=} \gamma_i \langle q_i | + \delta_i \langle q_{i-1} | + \gamma_{i-1} \langle q_{i-2} |$$

Therefore, the matrix to be diagonalized is

$$\tilde{T} = \begin{pmatrix} \delta_1 & \gamma_2 & 0 & \cdots & 0 \\ \gamma_2 & \delta_2 & \gamma_3 & & \\ 0 & \gamma_3 & \delta_3 & & \\ & \ddots & & & \end{pmatrix}$$

Summarized: (Lanczos method)

1) Determine the Lanczos basis $\vec{q}_i$ for $i \leq k$
2) Diagonalize $\tilde{T}$, as described in 3.2.
3) Convergence check, e.g.: is $w_k \approx w_{k-1}$?

Remarks:

i) Convergence is often very fast

Example: Heisenberg model, $S = 3/2$, $c = 0$

$\Rightarrow$ Hilbert space dimension $D_k = 4^{10} \approx 10^7$

$$\Rightarrow \quad \frac{w_{39} - w_{38}}{w_{38}} < 5 \cdot 10^{-8}$$

ii) The orthogonalization formulas are identical to the Gram–Schmidt method, evaluated for our special vectors.

$\vec{p}_i = \hat{A}^i \cdot \vec{q}$

Problem with (1)–(3): rounding errors can compromise orthogonality for large k.

iii) If k=N and $\mathrm{Span}(\vec{q}_1, \ldots, \vec{q}_N) = \mathbb{R}^N$

the Lanczos method is an alternative way to tridiagonalize a matrix.

But: The Housholder–Holder algorithm is more stable and

to be preferred.

iv) In the derivation of the Krylov spaces, we could also have replaced "min" by "max", with the same result.

⇒ At the same time, the Lanczos method also provides an approximation for the largest eigenvalue and its eigenvector.

v) The effort of the matrix-vector products $\tilde{A} \cdot \vec{\rho}_n$ is $O(N')$ where N' is the number of non-vanishing matrix elements. That means, that the dimension of the sparse matrix $\tilde{A}$ is irrelevant and can therefore be very large.