Modern Numerical Methods with Medical Applications
Part III: Hierarchical Matrices

Wolfgang Hackbusch

Max-Planck-Institut für *Mathematik in den Naturwissenschaften*

Inselstr. 22-26, 04103 Leipzig, Germany
wh@mis.mpg.de

http://www.mis.mpg.de/scicomp/hackbusch_e.html

Havanna, 2023

In our application we have solve equations $K_h u_h = f_h$ for many different right-hand sides f_h.

The lead-field matrix $\mathbf{L} \in \mathbb{R}^{M \times N}$ is a fully populated matrix requiring a large storage (e.g., $NM \gtrsim 10^8$):

$$\mathbf{L} = \left(R_i K_h^{-1} d_j \right)_{ij} \; .$$

The technique of hierarchical matrices allows the handling of huge matrices arising from elliptic PDEs (like K_h^{-1}) or integral equations, even if they are fully populated.

Literature:

W. Hackbusch: Hierarchical Matrices: Algorithms and Analysis. Springer 2015

1 Introduction

- Treatment of large-scale linear systems of equations is a common need in modern computations

- The use of matrices leads in general to difficulties

Large-scale systems: size $n = 10^6, 10^7$ or larger, depending on the storage size.

Fully populated matrices have n^2 entries; storage of $O(n^2)$ is usually not available.

Standard remedy: Restrict computations to sparse matrices ($O(n)$ non-zero entries) and use only matrix-vector multiplications (cost in computer time $O(n)$).

Goal of the hierarchical matrix technique: all matrix operations, in particular for full matrices.

Typical fields of application:

- Boundary Element Method (BEM):

Formulation of homogeneous elliptic boundary value problems by integral equation formulations

⇒ System matrices are fully populated

- Finite Element Method (FEM):

Elliptic boundary value problems lead to sparse matrices A, but for instance A^{-1} is full. LU-factors are partially filled.

Sometimes Schur complements

$$A_{11} - A_{12}\, A_{22}^{-1}\, A_{21}$$

are needed, which are also full.

- Further Applications

The costs in standard matrix approaches are:

- storage, $A * x$, $A + B$: $O(n^2)$
- $A * B$, A^{-1}, LU decomposition: $O(n^3)$.

The technique of hierarchical matrices tries to perform all matrix operations with a computational cost of

$$O(n \log^* n).$$

!! The results are only approximate (only $A * x$ is exact).

Already existing discretisation error $\varepsilon = O(n^{-\alpha})$. The additional approximation error should be $\leq \varepsilon$.

Preview: How do $\mathcal{H}$-matrices look like?

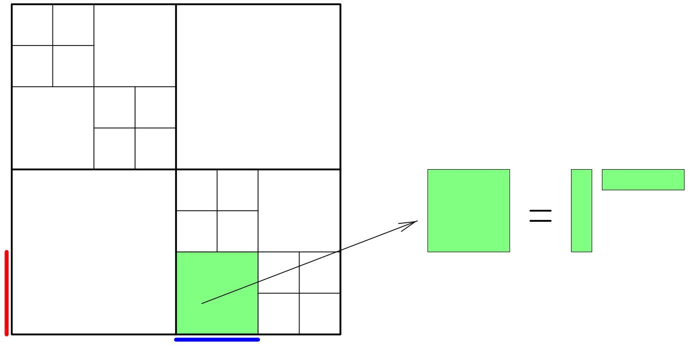
- Decompose the matrix into suitable subblocks.

- Approximate the matrix in each subblock by a rank-k-matrix*

$$block = \sum_{i=1}^{k} a_i b_i^{\top}$$

(for suitably small local rank k).

Illustration:



*k is upper bound. The true rank may be smaller.

Two Questions:

- How large is the representation error?
  More precisely: How does the local rank k correspond to the error of the matrix representation?

- How can the (approximate) matrix operations be performed such that

$$cost = O(n * \log^* n) \text{ ?}$$

Side Remark: About $\mathcal{R}k$-Matrices

Let the Rk-matrix $\sum_{i=1}^{k} a_i b_i^\top$ be of size $n \times m$.

REMARK: (a) The amount of storage is $(n+m)\,k$ (a_i and b_i to be stored).
(b) The amount of work for the matrix-vector multiplication $A * c$ ($c \in \mathbb{R}^m$) are

$$2k\,(m+n) - k - n \;\; \text{operations.}$$

Sums of Rk-Matrices, Truncation to Rank k
In general, the sum of Rk-matrices is of rank $2k$. Apply truncation to rank k by means of the singular-value decomposition:

$$A = U * D * V^\top, \quad (U, V \text{ unitary, } D \text{ diagonal with } d_1 \geq ... \geq d_{2k} \geq 0).$$

Truncation to rank k:

$$A' = U * D' * V^\top \quad \text{with } D' := \text{diag}\{d_1, ..., d_k, 0, ...\}$$

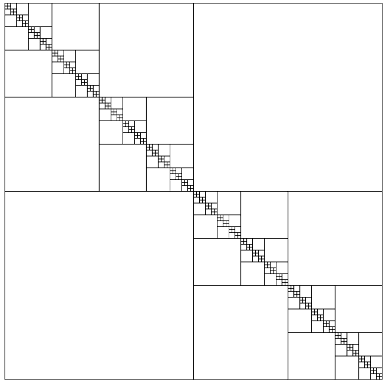
is of rank k and has the smallest Frobenius norm $\|A - A'\|_F$.

NOTATION: $A \oplus_{Rk} B$:=truncation of $A + B$ to rank k

REMARK: The *R1*-addition $\oplus_{R1}$ of two $R1$-matrices costs $9(n+m) + O(1)$ operations.

1.1 Example for Demonstration

Let $n = 2^p$, $p = 0, 1, \ldots$

The construction of the $\mathcal{H}$-matrix format  is recursive:
For $n = 1$, A is a rank-1-matrix. Otherwise the format of an $n \times n$ matrix of level p ($n = 2^p$) is

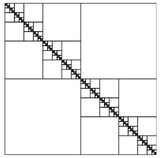
$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}$$

with

- A_{ij} are blocks of the size $\frac{n}{2} \times \frac{n}{2}$,
- A_{ii} ($i = 1, 2$) are $\mathcal{H}$-matrices (of level $p - 1$),
- A_{12}, A_{21} are rank-1-matrix (abbreviation: $R1$, $k = 1$).

2 Complexity of the $\mathcal{H}$-Matrix Arithmetic

2.1 Storage

Dimension: $n = 2^p$, p : hierarchy level: 

The construction yields

$$N_{storage}(p) = 2n + 2N_{storage}(p-1) \quad \text{for } p > 1.$$

Proof: The 2 off-diagonal blocks contain 4 vectors of size $n/2$.

Together with the induction start

$$N_{storage}(0) = 1 \text{ (case of } n = 1 = 2^0\text{)},$$

this leads to

LEMMA: The storage requirement for an $n \times n$ $\mathcal{H}$-matrix with $n = 2^p$ is

$$N_{storage}(p) = (2p+1)n = n(1 + 2\log_2 n).$$

2.2 Addition

A, B : $n \times n$ $\mathcal{H}$-matrices. Result: $C := A + B$.

For all blocks b we have to perform $C|_b := A|_b + B|_b$ (parallelisation possible!).

LEMMA: The R1-addition of two $n \times n$ $\mathcal{H}$-matrices or an $\mathcal{H}$-matrix and an R1-matrix requires

$$18n \log_2 n + O(n) \text{ operations.}$$

Proof: Exercise

2.3 Matrix-Vector Multiplication

A : $n \times n$ $\mathcal{H}$-matrix, x : n-vector, $(A, x) \longmapsto A * x$.

$$A = \begin{bmatrix} A_{11} & ab^\top \\ cd^\top & A_{22} \end{bmatrix}, \; x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \Rightarrow y := Ax = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} \text{ obtain by}$$

$$\begin{aligned} y_1 &= A_{11}x_1 + \alpha a, \\ y_2 &= A_{22}x_2 + \beta c \end{aligned}$$

with $\alpha := \langle b, x_2 \rangle$, $\beta := \langle d, x_1 \rangle$.

LEMMA: The matrix-vector multiplication of an $n \times n$ $\mathcal{H}$-matrix by a vector requires

$$4n \log_2 n - n + 2 \text{ operations.}$$

Proof: Exercise

2.4 Matrix-Matrix Multiplication

Three types of products are to be distinguished:

. 1) $R * R$ ($R1$-matrix times $R1$-matrix)

. 2) $R * H$ ($\mathcal{H}$-matrix times $R1$-matrix) or $H * R$

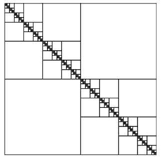
. 3) $H * H$ ($\mathcal{H}$-matrix times $\mathcal{H}$-matrix)

Type 1: $\left(ab^\top\right)\left(cd^\top\right) = (\alpha * a)\, d^\top$, with $\alpha = b^\top c$.

LEMMA: $N_{R1*R1}(p) = 3n - 1$ operations.

Type 2: $H * \left(ab^\top\right) = (H * a)\, b^\top$ requires only a matrix-vector multiplication.

LEMMA: $N_{H*R1}(p) = 4n \log_2 n - n + 2$ operations. Same for $R * H$.



Type 3: $H * H$ is computed recursively by

$$\begin{aligned} H * H &= \begin{bmatrix} H & R \\ R & H \end{bmatrix} * \begin{bmatrix} H & R \\ R & H \end{bmatrix} \\ &= \begin{bmatrix} \underline{H * H} + R * R & H * R + R * H \\ R * H + H * R & \underline{H * H} + R * R \end{bmatrix}. \end{aligned}$$

This leads to the recursion

$$\begin{aligned} N_{H*H}(p) &= 2N_{H \cdot H}(p-1) + 2N_{R \cdot R}(p-1) + 2N_{H \cdot R}(p-1) \\ &\quad +2N_{R \cdot H}(p-1) + 2N_{H+R}(p-1) + 2N_{R+R}(p-1) \end{aligned}$$

with the starting value $N_{H*H}(0) = 1$.

LEMMA: The multiplication of two $\mathcal{H}$-matrices requires

$$13n \log_2^2 n + 65n \log_2 n - 51n + 52 \text{ operations.}$$

Exercise: Prove the Lemma.

2.5 Matrix Inversion

Approximation of the inverse A^{-1} by an $\mathcal{H}$-matrix $Inv_{R1}(A)$.

Recursion with respect to p ($n = 2^p$): For $p = 0$, $Inv_{R1}(A) := A^{-1}$.

Having defined Inv_{R1} on level $p - 1$, the (exact) inverse of A is

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}^{-1} = \begin{bmatrix} A_{11}^{-1} + A_{11}^{-1} A_{12} S^{-1} A_{21} A_{11}^{-1} & -A_{11}^{-1} A_{12} S^{-1} \\ -S^{-1} A_{21} A_{11}^{-1} & S^{-1} \end{bmatrix}$$

with the Schur complement $S = A_{22} - A_{21} A_{11}^{-1} A_{12}$.

Recursion for the cost $N_{inv}(p)$:

$$\begin{aligned} N_{inv}(p) &= 2N_{inv}(p-1) + 4N_{H*R1}(p-1) \\ &\quad +2N_{H+R1}(p-1) + 2N_{R1*R1}(p-1). \end{aligned}$$

LEMMA: The approximate inversion of an $\mathcal{H}$-matrix requires

$13n \log_2^2 n + 47n \log_2 n - 109n + 110$ operations.

Exercise: Prove the Lemma.

2.6 LU-Decomposition

A is to be represented by

$$A \approx LU,$$

where L is a lower triangular matrix and U a upper triangular matrix of the $\mathcal{H}$-format.

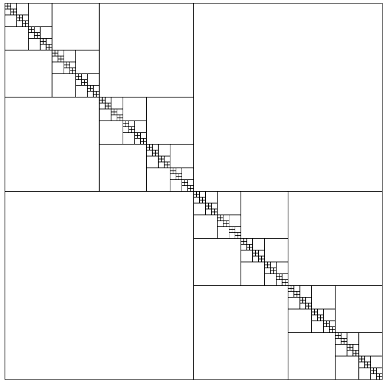
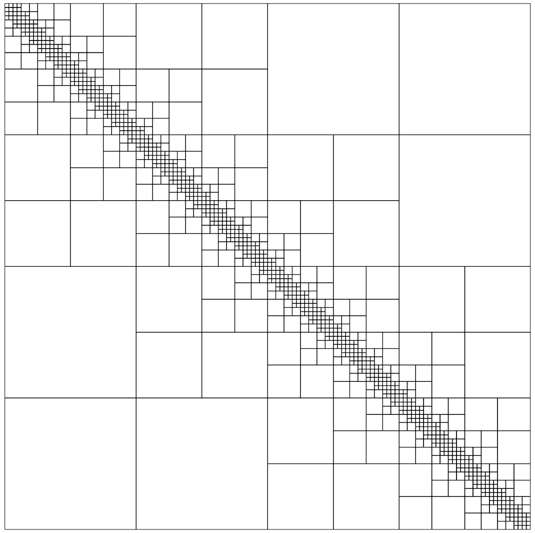
LEMMA: The approximate LU-decomposition costs

$$N_{LU}(p) = \frac{11}{2} n \log_2^2 n + 25n \log_2 n - 28n + 28$$

operations.

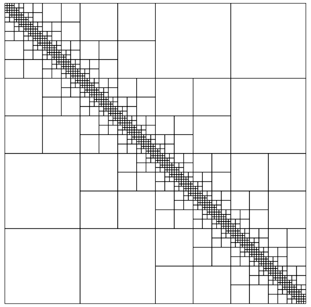
2.7 Concluding Remarks to the Introductory Case

At least, the rank 1 is to be replaced by a larger rank k.

Moreover, in general, the simple format  is to be replaced by a more refined format like 

3 General Construction of Hierarchical Matrices

Partition of the Matrix



How to partition the matrix in subblocks?

The structure of the matrix is described by index sets:

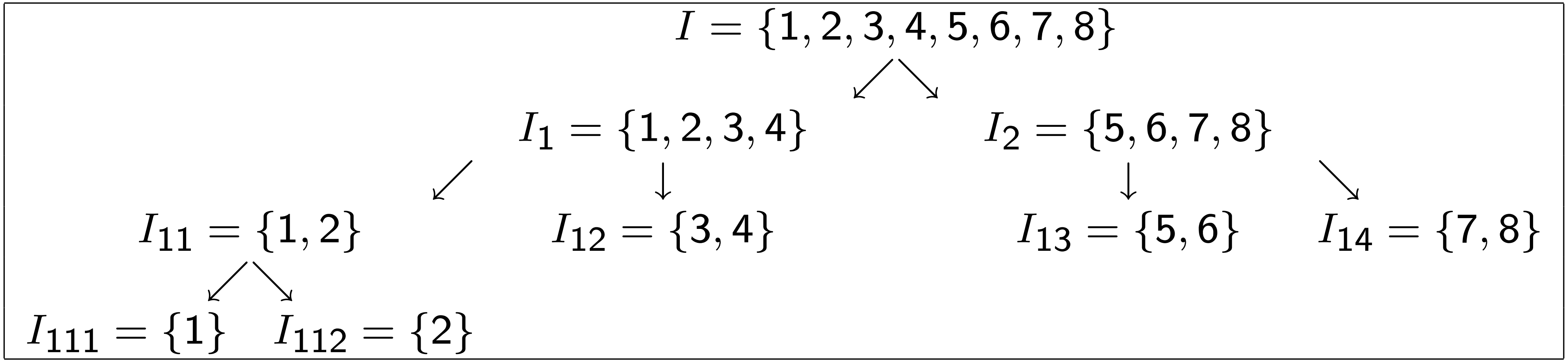
$$\begin{array}{rcl} I & : & \text{index set of matrix rows,} \\ J & : & \text{index set of matrix columns,} \\ M & \in & \mathbb{R}^{I \times J}. \end{array}$$

A block of the matrix is described by subsets: $\tau \subset I$, $\sigma \subset J$ defined the block $b = \tau \times \sigma$.

Remark: There is no need for an ordering of the index sets.

The subsets $\tau \subset I$ are organised by a **cluster tree** $T(I)$ [same for J]:
I is the root. The sons of a node τ are disjoint subsets $\tau_1, \tau_2, \ldots$ with $\cup \tau_i = \tau$.

Example of a binary cluster tree $T(I)$:



$I = \{1,2,3,4,5,6,7,8\}$

$I_1 = \{1,2,3,4\}$ $I_2 = \{5,6,7,8\}$

$I_{11} = \{1,2\}$ $I_{12} = \{3,4\}$ $I_{13} = \{5,6\}$ $I_{14} = \{7,8\}$

$I_{111} = \{1\}$ $I_{112} = \{2\}$

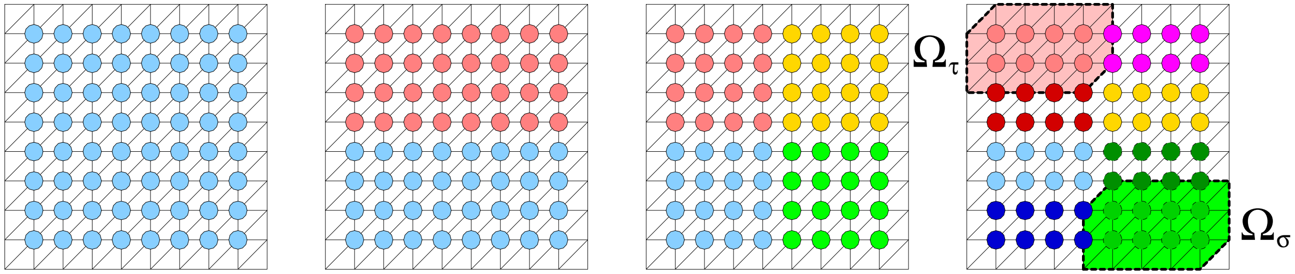
The cluster tree $T(I)$ contains blocks $\tau \in T(I)$ of different sizes (here sizes 1,2,4,8).

Strategy for the decomposition:

For discretisation of PDEs or integral equations each index $i \in I$ corresponds to a grid point or a FE nodal point $x_i \in \Omega$. It makes sense that each subset $\tau \in T(I)$ contains *neighboured* grid points.

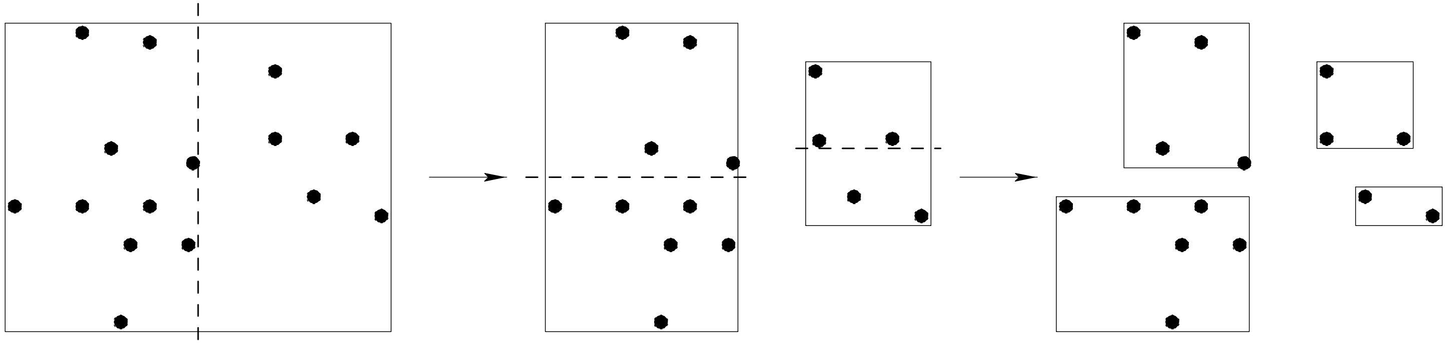
Possible *stopping criterion*: no decomposition for sufficient small blocks.

Example of grid points in a square (no matrix!):



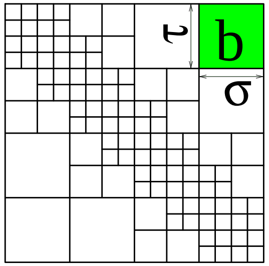
REMARK: For usual discretisations, an index $i \in I$ is associated to an nodal point $x_i \in \mathbb{R}^d$ or the support $\mathrm{supp}(\phi_i) \subset \mathbb{R}^d$ of a basis function ϕ_i.

The practical performance uses bounding boxes:



Block-Cluster Tree

$T(I)$: tree of the vector blocks.



$T(I \times J)$: tree containing the matrix blocks.
Each block $b \in T(I \times J)$ will be of the form

$$b = \tau \times \sigma \quad \text{with } \tau \in T(I) \text{ and } \sigma \in T(J).$$

Construction: 1) $I \times J$ is the root of $T(I \times J)$ (note that $I \in T(I)$, $J \in T(J)$)

2) Let $b = \tau \times \sigma \in T(I \times J)$.

2a) Either this block should not decomposed further,
in particular if τ is a leaf of $T(I)$ or σ is a leaf of $T(J)$. Otherwise:

2b) Let $\{\tau_i\}$ be the sons of $\tau \in T(I)$ and $\left\{\sigma_j\right\}$ the sons of $\sigma \in T(J)$. Then the sons of b are given by

$$b_{ij} = \tau_i \times \sigma_j.$$

Example: If $T(I)$ and $T(J)$ are binary trees, each block of $T(I \times J)$ is decomposed into 4 subblocks.

Admissibility

Critical question: Should $b \in T(I \times J)$ be decomposed or not?

Pro: small blocks yield a better approximation

Contra: A finer block decomposition leads to more blocks $\Rightarrow$ larger storage cost and more computational cost.

For applications from elliptic PDEs and integral equations the optimal choice is defined by an admissibility condition defined next.

If b is admissible it is not decomposed, otherwise it is decomposed.

Support of $\tau \in T(I)$:

In the case of FE discretisation, each $i \in I$ corresponds to a basis function ϕ_i. Set

$$\Omega_\tau = \bigcup_{i\in\tau} \mathrm{supp}\{\phi_i\} \subset \mathbb{R}^d.$$

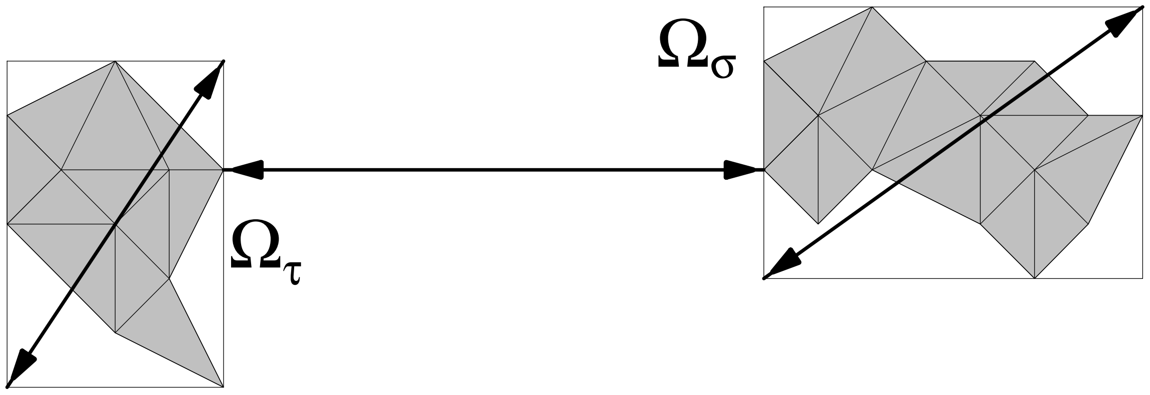
In the case of a difference scheme with grid points x_i, set

$$\Omega_\tau = \{x_i : i \in \tau\} \subset \mathbb{R}^d.$$

Define:

$$\mathrm{diam}(\tau) := \mathrm{diam}(\Omega_\tau), \quad \mathrm{dist}(\tau,\sigma) := \mathrm{dist}(\Omega_\tau, \Omega_\sigma).$$

Simplification: Replace the set Ω_τ by its bounding box:

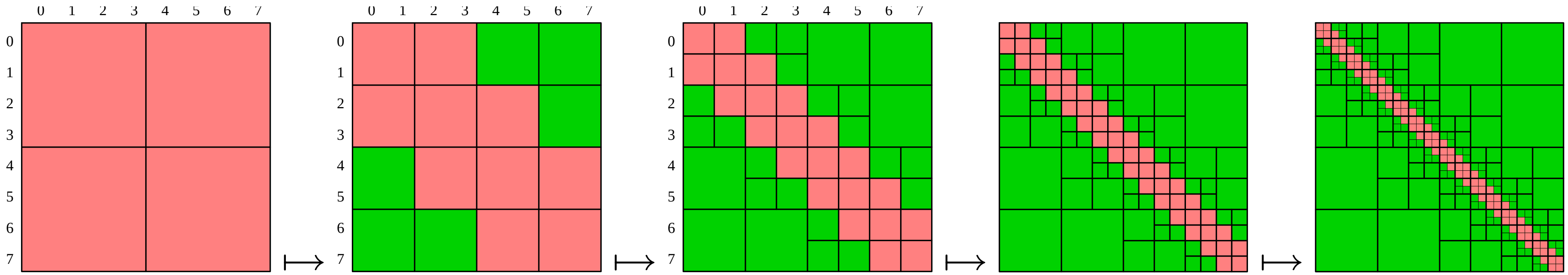


Admissibility condition: A block $\tau \times \sigma \in T(I \times J)$ is called admissible if

$$\min\{\mathrm{diam}(\Omega_\tau), \mathrm{diam}(\Omega_\sigma)\} \le \eta\, \mathrm{dist}(\Omega_\tau, \Omega_\sigma)$$

for some fixed $\eta > 0$.

Example: $x_i = ih$ are grid points in [0,1]:



green blocks: admissible, red: non-admissible

4 Application to Boundary Element Methods (BEM)

Example: $$(\mathcal{A}u)(x) := \int_0^1 \log|x-y|\,u(y)dy \qquad \text{for } x \in [0,1].$$

Discretisation: collocation with piecewise constant elements in

$$[x_{i-1}, x_i],\ x_i = ih, \qquad i = 1, \ldots, n,\ h = 1/n,$$

Midpoints $x_{i-1/2} = (i-1/2)h$ are the collocation points:

$$A = (a_{ij})_{i,j=1,\ldots,n} \qquad \text{with } a_{ij} = \int_{x_{j-1}}^{x_j} \log\left|x_{i-1/2} - y\right| dy.$$

Replace the kernel function $\kappa(x,y) = \log|x-y|$ in a certain range of x, y by an approximation $\tilde{\kappa}(x,y)$ of separable form

$$\tilde{\kappa}(x,y) = \textstyle\sum_{\iota \in J} X_\iota(x) Y_\iota(y).$$

$$\tilde{\kappa}(x,y) = \textstyle\sum_{\iota \in J} X_\iota(x) Y_\iota(y).$$

Possible choice: Taylor's formula applied with respect to y:

$$\begin{aligned} J &= \{0, 1, \ldots, k-1\}, \\ X_\iota(x) &= \text{derivatives of } \kappa(x,\cdot) \text{ evaluated at } y = y^*, \\ Y_\iota(y) &= (y - y^*)^\iota / \iota!. \end{aligned}$$

The kernel $\kappa(x,y) = \log|x-y|$ leads to the error estimate

$$|\kappa(x,y) - \tilde{\kappa}(x,y)| \le \frac{|y-y^*|^k/k}{(|x-y^*| - |y-y^*|)^k} \quad \text{for} \quad |x-y^*| \ge |y-y^*|.$$

If κ is replaced by $\tilde{\kappa}$, the integral $a_{ij} = \int_{x_{j-1}}^{x_j} \kappa(x_{i-1/2}, y)dy$ becomes

$$\tilde{a}_{ij} = \sum_{\iota \in J} X_\iota(x_{i-1/2}) \int_{x_{j-1}}^{x_j} Y_\iota(y)dy. \qquad (*)$$

Let b be a block and restrict i, j in $(*)$ to b. Then $(*)$ describes a block matrix $\tilde{A}|_b$. Each term of the sum in $(*)$ is an *R1*-matrix $ab^\top$ with

$$a_i = X_\iota(x_{i-1/2}), \quad b_j = \int_{x_{j-1}}^{x_j} Y_\iota(y)dy.$$

Since $\#J = k$, the block $\tilde{A}|_b$ is of *Rk*-type.

Furthermore, one can check that

$$|\kappa(x,y) - \tilde{\kappa}(x,y)| \le \frac{1}{k}\left(\frac{1}{2}\right)^k, \qquad \|A - \tilde{A}\|_\infty \le 2^{-k}/k.$$

Discretisation error $h^{\varkappa}$, where the step size h is related to $n = \#I$ by $h \sim \frac{1}{n}$. Hence k should be chosen such that

$$2^{-k} \sim \left(\frac{1}{n}\right)^{\varkappa}.$$

Hence,

$$k = O(\log n)$$

is the required rank.

NOTE: a) The construction of the cluster and block-cluster tree is automatic (black box) and fast. Even refinements with form-regular elements are allowed. b) Similarly, the construction of the approximation $\tilde{A}$ is black box like (usually by interpolation instead of Taylor expansion).

5 Application to FEM

REMARK a) A FEM system matrix is an $\mathcal{H}$-matrix (without any approximation error).
Proof: Non-trivial blocks $= 0$.

REMARK b) For a uniformly elliptic differential operator with L^∞-coefficients in a Lipschitz domain, the inverse of the FEM-matrix can be exponentially well approximated by an hierarchical matrix.

Literature:
Bebendorf-Hackbusch, Numer. Math. 95 (2003) 1-28
Faustmann-Melenk-Praetorius, Numer. Math. 131 (2015) 615-642.

Analytical Background

Boundary value problem:

$$\mathrm{div}(\sigma(x)\,\mathrm{grad}\,u) = f(x), \quad x \in \Omega \subset \mathbb{R}^d, \text{ with}$$
$$\sigma \in L^\infty(\Omega), \text{ eigenvalues} \in [c', c''], \ c' > 0.$$

$X, Y \subset \Omega$ admissible subsets, i.e., $\min\{\mathrm{diam}(X), \mathrm{diam}(Y)\} \leq \eta \ \mathrm{dist}(X, Y)$.

Then the Green function $G(x, y)$ admits an expansion

$$G(x, y) = \sum_{\nu=1}^{\infty} g'_\nu(x) g''_\nu(y) \qquad \text{for } x \in X, y \in Y,$$

which is **exponentially** convergent.

6 $\mathcal{H}$-LU iteration

Linear system of equations:

$$Ax = b.$$

Determine the LU decomposition of A by using hierarchical factors $L_{\mathcal{H}}$ and $U_{\mathcal{H}}$.

Since $L_{\mathcal{H}}U_{\mathcal{H}}$ is very close to A, it is a very good ‘preconditioner’; i.e., the iteration

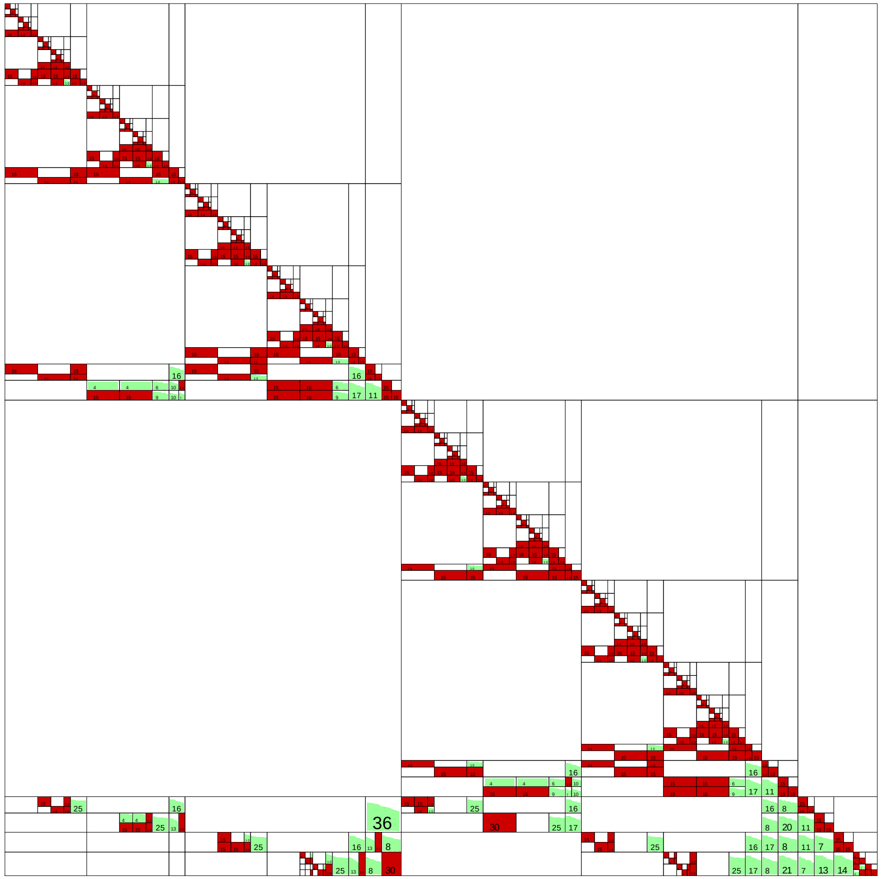
$$x^{m+1} = x^{m} - \left(L_{\mathcal{H}}U_{\mathcal{H}}\right)^{-1}\left(Ax^{m} - b\right)$$

is a very fast iteration.

One can prove:
If the inverse matrix can be approximated by hierarchical matrices, then also the LU factors do so.

$\mathcal{H}$-LU iteration for sparse matrices

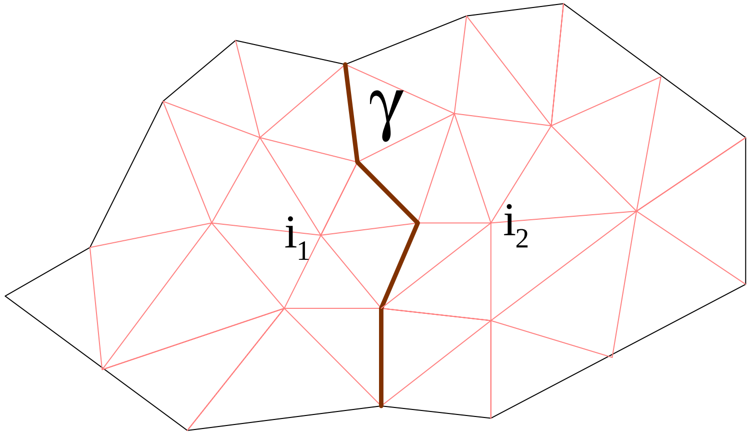
The partition of the matrix can be modified so that it corresponds to the nested dissection technique of A. George (1973).
Then sparsity of A is partially inherited by $L_{\mathcal{H}}$ and $U_{\mathcal{H}}$!

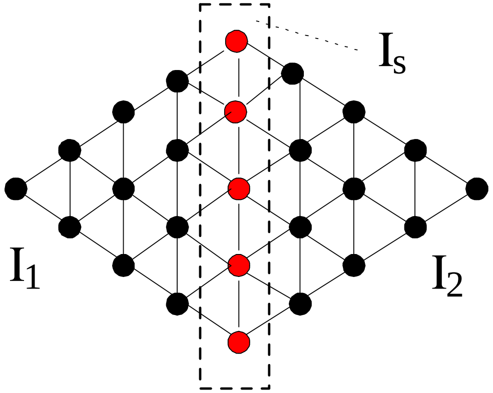


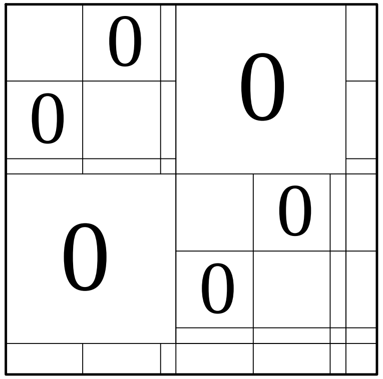
Example of a factor L:

Special Cluster Tree (nested dissection)

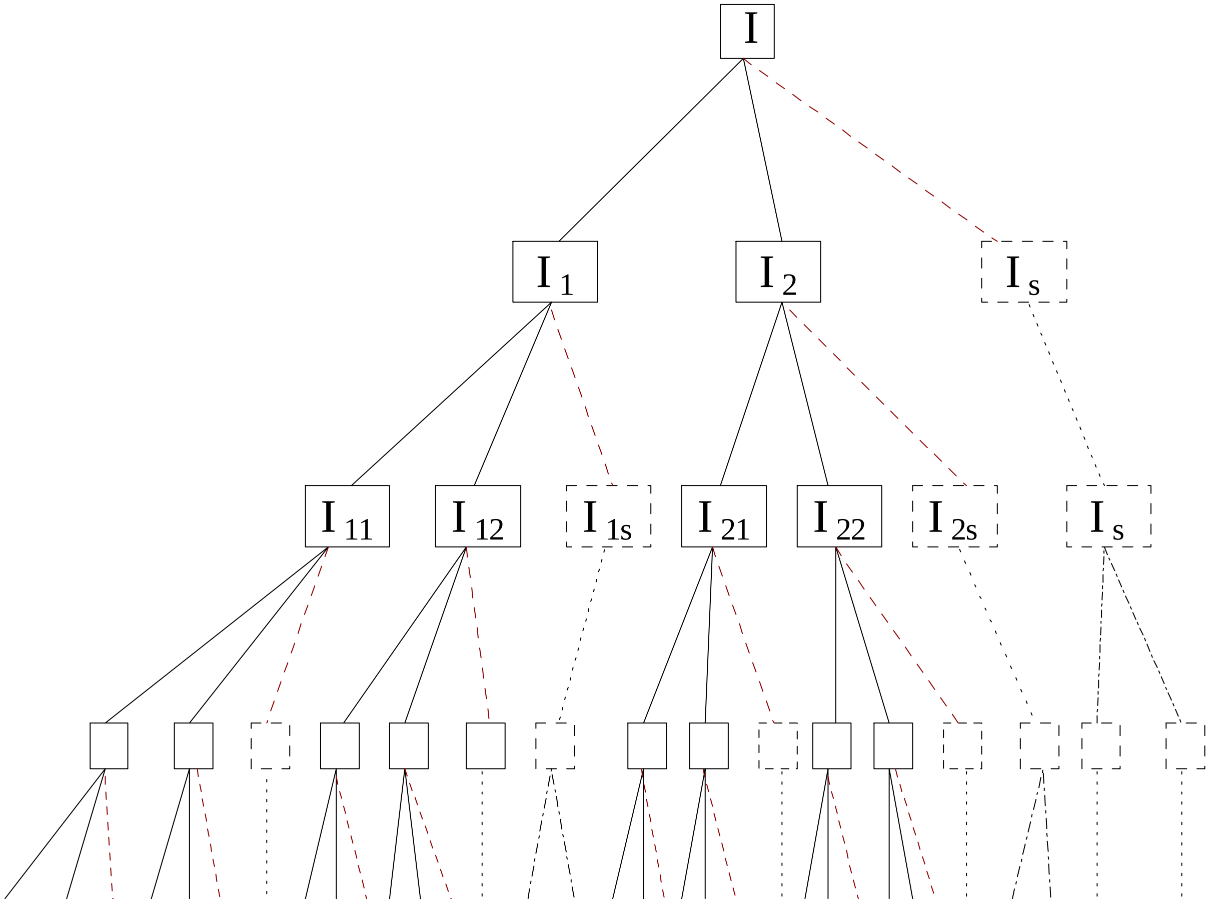
To get L, U with many zero blocks, the binary block decomposition is replaced by a ternary one. The FE grid is separated by an internal boundary γ:





The index set splits into the corresponding subsets I_1, I_2 and I_s (nodal points on γ). In the next step I_1, I_2 can be split again in the subsets I_{11}, I_{12}, I_{1s} and I_{21}, I_{22}, I_{2s}. The zero structure is shown in . However, this approach is not yet optimal.

Improved cluster tree (illustration for the 2D-case):



While I_1, I_2 are split, the cluster I_s is repeated identically at the next level and split in the next but one level. This treatment guarantees that the diameters of the clusters are similar in each level.

Geometry-free Approach:

So far, the admissibility condition used the Euclidean diameter and distance of clusters.

Problem: The geometric description of x_i or ϕ_i may be not available.

In the case of a FEM there is a simple remedy:

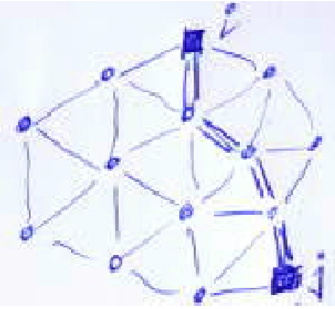
The matrix graph $G(A)$ for a matrix $A \in \mathbb{R}^{I \times I}$ consists of the nodes I and the egdes

$$G(A) = \left\{ (i,j) \in I \times I : \ a_{ij} \neq 0 \right\}.$$

FE Case: only neighboured nodes connected ⇒

the Euclidean distance can be replaced by the graph distance.

.



dist$(i,j) = 4$

This *algebraic approach* is perfect for blackbox applications.

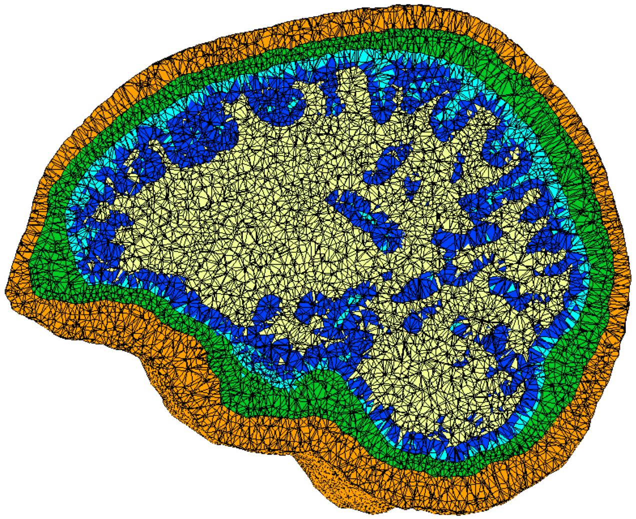
EXAMPLE (inverse Problem, Wolters-Grasedyck-Hackbusch, 2004):

Given: electric/magnetic field (EEG,MEG) at $\approx$ 400 sensor positions on the head surface.

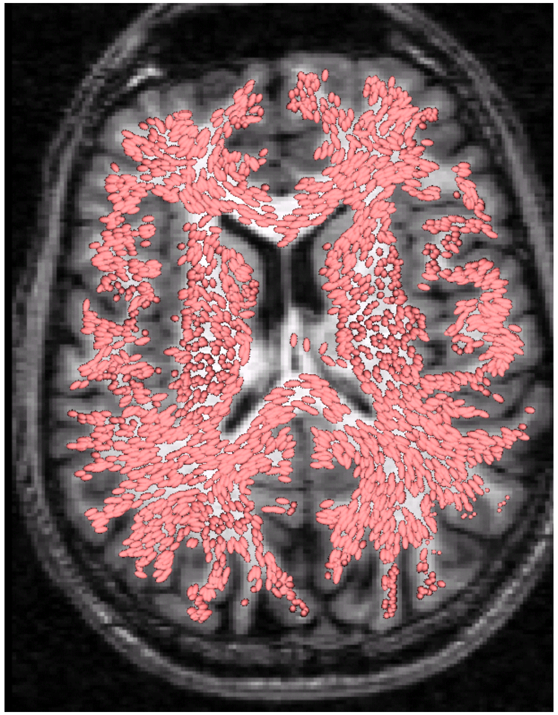
What is the current distribution in the brain ? Where are the sources ?

PDE: $-\text{div}\,\sigma(x)\nabla u(x) = f(x), \quad x \in \Omega \subset \mathbb{R}^3$, $\partial_n u = 0$ on $\partial\Omega$.

The boundary value problem has to be solved for $\approx$ 400 right-hand sides



Triangulation with
$N = 147287$ tetraeder



conductivity σ

- Galerkin discretisation $\rightsquigarrow Ax = b$
- The system has to be solved for $\approx$ 400 right-hand sides b
- Stopping criterion: $\|Ax - b\| / \|b\| \leq 10^{-8}$
- Machine: SUNFire, 900 MHz, single processor

	Pardiso[†]	$\mathrm{LU}_{\mathcal{H}}$, $\varepsilon = 10^{-6}$	PEBBLES[‡]
Setup	237	468	13
Solve	2.4	1.0	10
Total	1197	868	4013

[†] Pardiso (Schenk & Co)

[‡] PEBBLES (Langer/Haase)

7 $\mathcal{H}^2$-Matrices

Two hierarchies are involved:

1. Hierarchy given by the cluster tree T.

2. The involved rank-k-matrices do not use arbitrary row and column vectors, but vectors from special subspaces V_τ $(\tau \in T)$, i.e., the matrix blocks belong to tensor spaces $V_\tau \otimes V_\sigma$

3. The basis of V_τ is connected with the bases of $V_{\tau'}$ for $\tau' \in S(\tau)$. This leads to hierarchically defined bases: $V_\tau|_{\tau'} \subset V_{\tau'}$.

Since, in the end, the bases need not be stored directly, the log-factor disappears:

$$storage(A),\ cost(A * x),\ cost(A + B),\ cost(A * B) = O(n)$$

and smaller constants.

S. Börm: Efficient Numerical Methods for Non-local Operators. EMS, Zürich (2013)

8 Matrix Equations

$$\begin{array}{lrcl} \text{Lyapunov:} & AX + XA^\top & = & C \\ \text{Sylvester} & AX - XB & = & C \\ \text{Riccati:} & AX + XA^\top - XFX & = & C \end{array}$$

Given: A, B, C, F; desired matrix-valued solution: X.

Applications: optimal control problems for elliptic / parabolic PDEs.

- Low rank $C, F \Rightarrow$ low rank X

- $\mathcal{H}$-matrix C, low rank $F \Rightarrow \mathcal{H}$-matrix X

Computation via $\mathcal{H}$-arithmetic, possibly combined with multi-grid methods.

Matrix-Riccati Equation

$$A^\top X + XA - XFX + G = O \qquad (A < O).$$

Lemma 8.1 *The solution X satisfies*

$$X = -(M^\top M)^{-1} M^\top N,$$

where

$$\begin{bmatrix} M & N \end{bmatrix} := \mathsf{sign}\left(\begin{bmatrix} A^\top & G \\ F & -A \end{bmatrix}\right) - \begin{bmatrix} I & O \\ O & I \end{bmatrix}.$$

$S = T\,\mathsf{diag}\{\lambda_1, \ldots\} T^{-1} \Rightarrow \mathsf{sign}(S) := T\,\mathsf{diag}\{\mathsf{sign}(\lambda_1), \ldots\} T^{-1}.$

Lemma 8.2 *Assume that $\Re e\lambda \neq 0$ for all eigenvalues $\lambda \in \sigma(S)$.*

Start: $S^{(0)} := S$. Then the iteration

$$S^{(i+1)} := \frac{1}{2}\left(S^{(i)} + \left(S^{(i)}\right)^{-1}\right)$$

converges quadratically to $\mathsf{sign}(S)$.

Example of a matrix-Riccati equation by L. Grasedyck

Choice of A by $A = \Delta_h$ (1D-Laplacian).

The following table shows the relative error $\left\|\tilde{X} - X\right\|_2 / \|X\|_2$.

	$n =$ 101	256	1024	65 536
$k = 1$	8.8_{10}-3	1.5_{10}-1	1.3_{10}-1	-
$k = 2$	2.4_{10}-4	2.6_{10}-4	4.2_{10}-4	6.7_{10}-4
$k = 4$	7.7_{10}-8	9.1_{10}-8	1.1_{10}-7	6.2_{10}-7
$k = 6$	1.9_{10}-10	3.7_{10}-10	2.4_{10}-10	1.7_{10}-9
Number of iterations	12	14	17	26
time* [sec]	2.2	8.5	67	18263

*) k=2, Sun Quasar 450 MHz

In the last case, the (full) matrix X has $4,294,967,296$ entries.

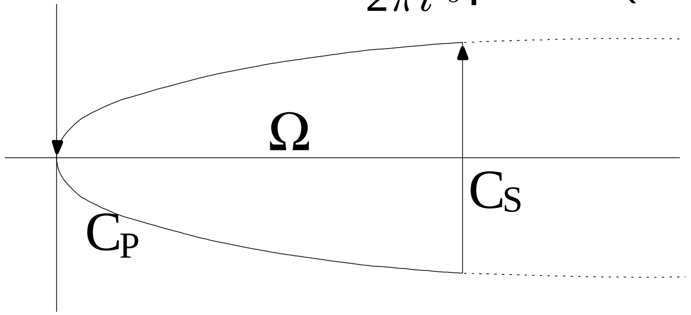
L. Grasedyck, W. Hackbusch, B. Khoromskij: *Solution of large scale algebraic matrix Riccati equations by the use of hierarchical matrices*. Computing **70**, 121-165 (2003)

9 Matrix-Valued Functions $f(A)$

EXAMPLE: Matrix-exponential function e^{-tA}.

Cauchy-Dunford representation: $e^{-tA} = \frac{1}{2\pi i} \int_{\Gamma} e^{-zt} (zI - A)^{-1} \, dt$

using a parabola Γ :



After parametrisation and quadrature:

$$T_N(t) := \sum_{\ell=-N}^{N} \gamma_\ell e^{-z_\ell t} (z_\ell I - A)^{-1}, \qquad z_\ell \in \Gamma.$$

Error estimate for $t \geq t_0 > 0$:

$$\left\| T_N(t) - e^{-tA} \right\| \lesssim e^{-cN^{2/3}}.$$

$\Rightarrow N \sim \log n \;\Rightarrow$ Total cost: $O(n \log^* n)$.

I.P. Gavrilyuk, W. Hackbusch, B. Khoromskij: *$\mathcal{H}$-matrix approximation for the operator exponential with applications*. Numer. Math. **92**, 83-111 (2002).

10 Higher dimensional analogue: Tensor systems

The analogue of rank-k-matrices are sums of k tensor products.

Tensor space:

$$\mathbb{V} := V_1 \otimes V_2 \otimes \ldots \otimes V_d.$$

Example: $V_i = \mathbb{R}^{I_i}$ for index sets I_i. Then the entries of $v \in \mathbb{V}$ are

$$v_{i_1, i_2, \ldots, i_d} \qquad \text{with } i_j \in I_j.$$

$\mathbb{V}$ is isomorphic to $\mathbb{R}^I$ with the product index set $I := I_1 \times I_2 \times \ldots \times I_d$.

DEFINITION: A rank-k-tensor is of the form

$$\sum_{\mu=1}^{k} v_1^{(\mu)} \otimes v_2^{(\mu)} \otimes \ldots \otimes v_d^{(\mu)} \qquad \text{with } v_j^{(\mu)} \in V_j.$$

DEFINITION: A rank-k-tensor is of the form

$$\sum_{\mu=1}^{k} v_1^{(\mu)} \otimes v_2^{(\mu)} \otimes \ldots \otimes v_d^{(\mu)} \qquad \text{with } v_j^{(\mu)} \in V_j. \tag{*}$$

QUESTION: Given $v \in \mathbb{V}$, are there best rank-k-approximations (*) ?
How can they be computed?

REMARK: Tools like the singular-value decomposition do not exist for $d \geq 3$.
Non-existence of best-approximations and numerical instability possible.

A trust-region Newton method is described by Espig (Diss. 2008).

Example from the electronic Schrödinger equation

Hartree-Fock equation $F_\psi \, \psi_b(\mathbf{y}) = \epsilon_b \, \psi_b(\mathbf{y})$ involves the Hartree potential

$$V_H(\mathbf{x}) = 2 \sum_{b=1}^{N/2} \int \frac{\psi_b^*(\mathbf{y})\psi_b(\mathbf{y})}{|\mathbf{x}-\mathbf{y}|} d\mathbf{y} = \int \frac{\rho(\mathbf{y})}{|\mathbf{x}-\mathbf{y}|} \, d\mathbf{y}, \tag{1}$$

where $\rho(\mathbf{y}) = 2 \sum_{b=1}^{N/2} \psi_b(\mathbf{y})\psi_b^*(\mathbf{y})$ is the electron density.

Standard approach uses Gaussians $g_k^{(j)}(y_j) = (y_j - A_k^{(j)})^{\ell_k} \, e^{-\alpha_k (y_j - A_k^{(j)})^2}$ to represent the orbital (wavefunction) by

$$\psi_b(\mathbf{y}) \approx \sum_{k=1}^{K_\psi} g_k^{(1)}(y_1) \, g_k^{(2)}(y_2) \, g_k^{(3)}(y_3). \tag{2}$$

Here, $K_\psi =$ tensor rank. We start with a representation (2) produced by the MOLPRO program package using the MATROP program for matrix operations. Eq. (2) yields $\rho(\mathbf{y}) = \psi_b^*(\mathbf{y})\psi_b(\mathbf{y})$ with $K := K_\psi(K_\psi + 1)/2$ terms.

Optimising the tensor representation reduces the tensor rank to a much smaller rank κ while almost keeping the same order of accuracy:

$$\rho(\mathbf{y}) \approx \sum_{k=1}^{\kappa} \varrho_k^{(1)}(y_1)\, \varrho_k^{(2)}(y_2)\, \varrho_k^{(3)}(y_3), \qquad \kappa \ll K.$$

The computational work for evaluating the Hartree potential (1) depends essentially on the tensor rank.

EXAMPLE CH_4: The MOLPRO program yields $K = 1540$, which can be reduced by our approach to $\kappa = 45$. The computing time for evaluating V_H for the tensor representation with $\kappa = 45$ is 8 hours, while the estimated time for $K = 1540$ is 190 hours.

molecule	initial rank $K$of $\rho(y)$	final rank κ	relative error	error in energy (hartree)
CH_4	1540	45	9.0×10^{-6}	6.0×10^{-5}
C_2H_2	2346	50	1.3×10^{-4}	5.0×10^{-4}
C_2H_6	4656	55	8.8×10^{-5}	4.0×10^{-4}

see Rao Chinnamsetty - Espig - Khoromskij - Hackbusch - Flad: J. Chem. Physics 127 (2007) and Rao Chinnamsetty, Diss. 2008.

Kronecker-Tensor Products

$V_j \ = \mathbb{R}^{I_j \times J_j}$ vector spaces of matrices. Then

$$\mathbb{V} := V_1 \otimes V_2 \otimes \ldots \otimes V_d \cong \mathbb{R}^{I \times J}$$

$$\text{with} \quad I := I_1 \times I_2 \times \ldots \times I_d \quad \text{and} \quad J := J_1 \times J_2 \times \ldots \times J_d.$$

Notation for $d = 2$: $\quad A \otimes B = \begin{bmatrix} A_{11}B & A_{12}B & \ldots \\ A_{21}B & \ddots & \\ \vdots & & \end{bmatrix}$

REMARK: a) For $d = 2$ the approximation of a matrix M by a Kronecker-rank-k expression $\sum_{\mu=1}^{k} A^{(\mu)} \otimes B^{(\mu)}$ is equivalent to a certain standard rank-k approximation of a related matrix $\tilde{M}$.

b) For $d \geq 3$ the search for rank-k approximations is more involved.

If the matrix is the discretisation of a continuous operator with a kernel function $\varkappa(\mathbf{x}, \mathbf{y})$, $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$, analytical methods may help (see next example).

Separable PDE in $[0,1]^d$, d large

Let $\Omega = (0,1)^d \subset \mathbb{R}^d$.
Equidistant grid: $\Omega_h = (h, 2h, \ldots, nh)$ with

$$(n+1)\,h = 1 \qquad \text{(here } n = 1024\text{).}$$

Separable PDE: $L = \sum_{\nu=1}^{d} a_\nu(x_v) \frac{\partial^2}{\partial x_v^2}$, e.g., $L = \Delta$.

Discretisation of $-L$ by usual difference formula:

$$\begin{aligned} A = -L_h &= -\sum_{\nu=1}^{d} a_\nu(x_v) D^h_{x_\nu x_\nu} \qquad (D^h_{x_\nu x_\nu}\text{: 2nd difference}) \\ &= A_1 \otimes I \otimes \ldots \otimes I + I \otimes A_2 \otimes \ldots \otimes I + \ldots + I \otimes I \otimes \ldots \otimes A_d \end{aligned}$$

Goal: Approximation of L_h^{-1}.

Numerical result (Grasedyck 2004):
For $d = 2048$, accuracy 10^{-5} to 10^{-6}: 5 min computer time

Related dimension:

$$N = 1024^{2048} = 1.24 \times 10^{6165}.$$

Underlying method

$1/x$ can be approximated by exponential sums $\sum_{\nu=1}^{k} \omega_\nu \exp(\alpha_\nu x)$:

$$\begin{aligned}
\min_{\omega_\nu,\alpha_\nu} \max_{x\in[x_0,x_1]} \left| \frac{1}{x} - \sum_{\nu=1}^{k} \omega_\nu \exp(\alpha_\nu x) \right| &\leq O(e^{-ck}), \quad c > 0, \\
\min_{\omega_\nu,\alpha_\nu} \max_{x\in[x_0,\infty)} \left| \frac{1}{x} - \sum_{\nu=1}^{k} \omega_\nu \exp(\alpha_\nu x) \right| &\leq O(e^{-ck^{1/2}}), \quad c > 0.
\end{aligned}$$

Let $[x_0, x_1]$ or $[x_0, \infty)$ contain the spectrum of L_h. Then

$$L_h^{-1} \approx \sum_{\nu=1}^{k} \omega_\nu \exp(\alpha_\nu L_h).$$

The special tensor structure

$$L_h = \sum_{\mu=1}^{d} I \otimes \ldots \otimes I \otimes L_{h,\mu} \otimes I \otimes \ldots \otimes I$$

implies $\exp(\alpha_\nu L_h) = \bigotimes_{\mu=1}^{d} \exp(\alpha_\nu L_{h,\mu})$.
Approximation of $\exp(\alpha_\nu L_{h,\mu})$ by $\mathcal{H}$-matrices (see above). Finally:

$$L_h^{-1} \approx \sum_{\nu=1}^{k} \omega_\nu \bigotimes_{\mu=1}^{d} \exp_{\mathcal{H}}(\alpha_\nu L_{h,\mu}) \qquad \text{(rank-}k\text{-tensor)}.$$

- For scientific purpose the software library HLib is freely available (ask for a licence form)