King Abdullah University of Science and Technology KAUST

Hierarchical matrices. Part 1.

Alexander Litvinenko

KAUST, SRI-UQ Center
http://sri-uq.kaust.edu.sa/

www.hlib.org

www.hlibpro.com

December 15, 2013

Content

1. Motivation
2. Low-rank matrices
3. Cluster Tree, Block Cluster Tree and Admissibility condition
4. 1D BEM example
5. Hierarchical matrices: cost and storage
6. Two applications

Used: H-matrices Winter School Script (www.hlib.org), PhD thesis of Ronald Kriemann, preprints from www.mis.mpg.de

$$Ax = b$$

Iterative methods: Jacobi, Gauss- Seidel, SOR, ...
Direct solvers: Gaussian elimination, domain decompositions, LU,...
Cost of A^{-1} is $\mathcal{O}(n^3)$, number of iteration is proportional to $\sqrt{\text{cond(A)}}$.

If A is structured (diagonal, Toeplitz, circulant) then can apply e.g. FFT, but if not ?
What if you need not only $x = A^{-1}b$, but $f(A)$
(e.g. A^{-1}, $\exp A$, sin$A$, sign$A$, ...)?

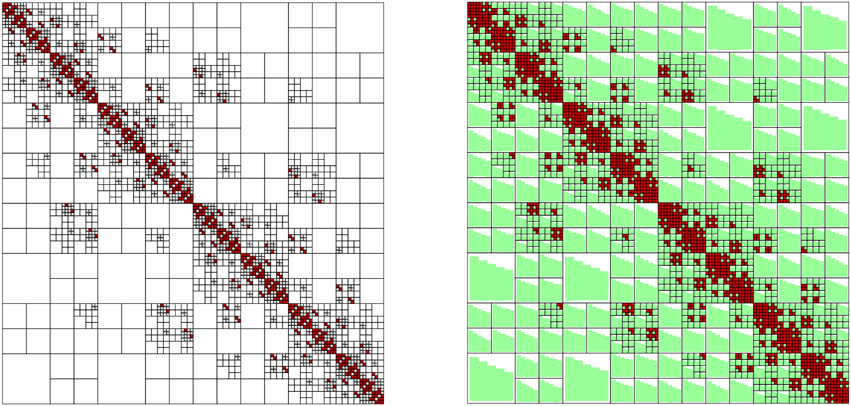


Figure : The $\mathcal{H}$-matrix approximation of the stiffness matrix of the Poisson problem (**left**) and its inverse (**right**). The dark blocks are dense matrices. The light blocks are low-rank matrices with maximal rank $k_{max} = 5$.

$M \in \mathbb{R}^{n\times m}$, $U \approx \tilde{U} \in \mathbb{R}^{n\times k}$, $V \approx \tilde{V} \in \mathbb{R}^{m\times k}$, $k \ll \min(n, m)$. The storage $\tilde{M} = \tilde{U}\tilde{\Sigma}\tilde{V}^T$ is $k(n+m)$ instead of $n \cdot m$ for M represented in the full matrix format.

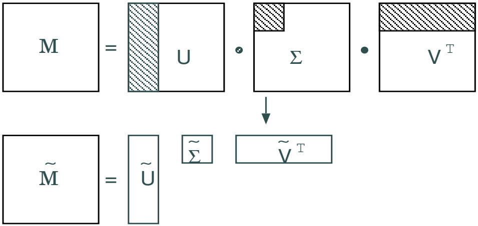
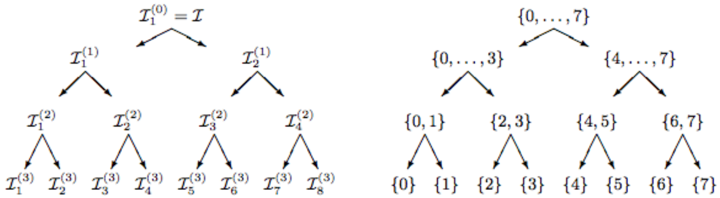
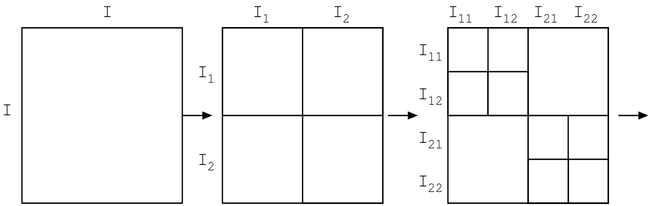


Figure : Reduced SVD, only k biggest singular values are taken.

$\mathcal{H}$-matrices (Hackbusch '99)

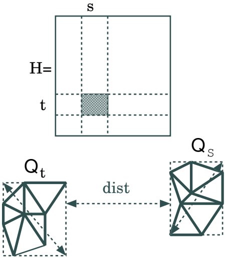
1. Build cluster tree T_I and block cluster tree $T_{I \times I}$.

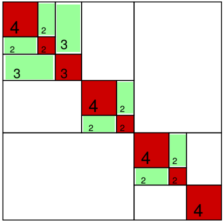




2. For each $(t \times s) \in T_{I \times I}$, $t, s \in T_I$, check admissibility condition
$\min\{diam(Q_t), diam(Q_s)\} \leq \eta \cdot dist(Q_t, Q_s)$.

if(adm=true) then $M|_{t \times s}$ is a rank-k matrix block
if(adm=false) then divide $M|_{t \times s}$ further or define as a dense matrix block, if small enough.





Resume: Grid → cluster tree (T_I) + admissibility condition → block cluster tree ($T_{I \times I}$) → $\mathcal{H}$-matrix → $\mathcal{H}$-matrix arithmetics.

Let B_1, $B_2 \subset \mathbb{R}^d$ are compacts, and $\chi(x, y)$ is defined for $(x, y) \in B_1 \times B_2$ with $x \neq y$.
Let $\mathcal{K}$ be an integral operator with an asymptotic smooth kernel χ in the domain $B_1 \times B_2$:

$$(\mathcal{K}v)(x) = \int_{B_2} \chi(x, y)v(y)dy \quad (x \in B_1).$$

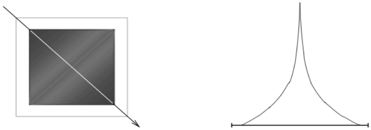
Suppose that $\chi^{(k)}(x, y)$ is an approximation of χ in $B_1 \times B_2$ of the separate form:

$$\chi^{(k)}(x, y) = \sum_{\nu=1}^{k} \varphi_\nu^{(k)}(x)\psi_\nu^{(k)}(y),$$

where k is the rank of separation.
Then $\|\chi - \chi^{(k)}\|_{\infty, B_1 \times B_2} \leq c_1 \left[\frac{c_2 \min\{diam(B_1), diam(B_2)\}}{dist(B_1, B_2)} \right]^k$.

Consider the following integral equation

$$\int_0^1 \log|x-y|U(y)dy = F(x), \qquad x \in (0,1).$$



After discretisation by Galerkin's method we obtain

$$\int_0^1 \int_0^1 \phi_i(x)\log|x-y|U(y)dydx = \int_0^1 \phi_i(x)F(x)dx, \qquad 0 \le i < n,$$

in the space $V_n := span\{\phi_0, ..., \phi_{n-1}\}$, where ϕ_i, $i = 1, ..., n-1$, are some basis functions in BEM. The discrete solution U_n in the space V_n is $U_n := \sum_{j=0}^{n-1} u_j\phi_j$ with u_j being the solution of the linear system

$$Gu = f,\ G_{ij} := \int_0^1 \int_0^1 \phi_i(x) \log|x-y| \phi_j(y) dy dx,\ f_i := \int_0^1 \phi_i(x) F(x) dx. \quad (1)$$

We replace the kernel function $g(x, y) = \log|x - y|$ by a degenerate kernel

$$\tilde{g}(x, y) = \sum_{\nu=0}^{k-1} g_\nu(x) h_\nu(y). \quad (2)$$

Then we substitute $g(x, y) = \log|x - y|$ in (1) for $\tilde{g}(x, y)$

$$\tilde{G}_{ij} := \int_0^1 \int_0^1 \phi_i(x) \sum_{\nu=0}^{k-1} g_\nu(x) h_\nu(y) \phi_j(y) dy dx.$$

After easy transformations

$$\tilde{G}_{ij} := \sum_{\nu=0}^{k-1} (\int_0^1 \phi_i(x) g_\nu(x) dx)(\int_0^1 h_\nu(y) \phi_j(y) dy).$$

Now, all admissible blocks $G|_{(t,s)}$ can be represented in the form

$$G|_{(t,s)} = AB^T, \quad A \in \mathbb{R}^{|t| \times k}, \quad B \in \mathbb{R}^{|s| \times k},$$

where the entries of the factors A and B are

$$A_{i\nu} := \int_0^1 \phi_i(x) g_\nu(x) dx, \quad B_{j\nu} := \int_0^1 \phi_j(y) h_\nu(y) dy.$$

We use the fact that the basis functions are local and obtain for all inadmissible blocks:

$$\tilde{G}_{ij} := \int_{i/n}^{(i+1)/n} \int_{j/n}^{(j+1)/n} \log|x-y| dy dx.$$

Let $\mathcal{H}(T_{I\times J},k) := \{M \in \mathbb{R}^{I\times J} \mid rank(M\mid_{t\times s}) \leq k$ for all admissible leaves $t\times s$ of $T_{I\times J}\}$, $n := \max(|I|,|J|,|K|)$.

Operation	Sequential Compl.	Parallel Complexity (R.Kriemann 2005)
$building(M)$	$N=\mathcal{O}(n\log n)$	$\frac{N}{p}+\mathcal{O}(\|V(T)\setminus\mathcal{L}(T)\|)$
$storage(M)$	$N=\mathcal{O}(kn\log n)$	N
Mx	$N=\mathcal{O}(kn\log n)$	$\frac{N}{p}+\frac{n}{\sqrt{p}}$
$\alpha M'\oplus\beta M''$	$N=\mathcal{O}(k^2 n\log n)$	$\frac{N}{p}$
$\alpha M'\odot M''\oplus\beta M$	$N=\mathcal{O}(k^2 n\log^2 n)$	$\frac{N}{p}+\mathcal{O}(C_{sp}(T)\|V(T)\|)$
M^{-1}	$N=\mathcal{O}(k^2 n\log^2 n)$	$\frac{N}{p}+\mathcal{O}(nn_{min}^2)$
LU	$N=\mathcal{O}(k^2 n\log^2 n)$	N
$\mathcal{H}$-LU	$N=\mathcal{O}(k^2 n\log^2 n)$	$\frac{N}{p}+\mathcal{O}(\frac{k^2 n\log^2 n}{n^{1/d}})$

1. Matrix exponential allows us to solve ODEs

$$\dot{x}(t) = Ax(t), \quad x(0) = x_0, \quad \rightarrow \ x(t) = \exp(tA)x_0$$

2. Other matrix function: use representation by the Cauchy integral

$$f(M) = \frac{1}{2\pi i} \oint_{\Gamma} f(t)(M - tI)^{-1} dt$$

and exponentially convergent quadrature rule

$$f(M) \approx \sum_{j=1}^{k} w_j f(t_j)(M - t_j I)^{-1}$$

to be approximated.

Conclusion

+ Complexity and storage is $\mathcal{O}(k^r n \log^q n)$, $r = 1, 2, 3$, $q = 1, 2$
+ Allow to compute $f(A)$ efficiently for some class of functions f
+ Many examples: FEM 1D, 2D and 3D, BEM 1D,2D and 3D, Lyapunov, Riccati matrix equations
+ Well appropriate to be used as a preconditioner (for iterative methods)
+ There are sequential (www.hlib.org) and parallel (www.hlibpro.com) libraries
+ There are A LOT of implementation details!
- Not so easy implementation
- Can be large constants in 3D

Thanks for your attention!

www.hlib.org

www.hlibpro.com