$\mathcal{DH}^2$-matrix compression for Helmholtz problems

Steffen Börm with Christina Börst

Christian-Albrechts-Universität zu Kiel

Oberwolfach Mini-Workshop
"Fast Solvers for Highly Oscillatory Problems", 2016

Overview

Overview

Helmholtz equation

Goal: Solve the Helmholtz equation

$$\begin{aligned} \Delta u(x) + \kappa^2 u(x) &= 0 && \text{for all } x \in \Omega, \\ u(x) &= f(x) && \text{for all } x \in \partial\Omega. \end{aligned}$$

Challenge: Indefinite operator, pollution effect.

Approach: Boundary integral formulation

$$u(x) = \int_{\partial\Omega} g(x,y) \frac{\partial u}{\partial n}(y)\,dy - \int_{\partial\Omega} \frac{\partial g}{\partial n(y)}(x,y) u(y)\,dy \quad \text{for all } x \in \Omega$$

with fundamental solution

$$g(x,y) = \frac{\exp(\iota\kappa\|x-y\|)}{4\pi\|x-y\|}.$$

Challenge: Standard discretization leads to dense matrix.

Fast summation

Standard approach: Approximate g by degenerate kernel,

$$g(x,y) \qquad \approx \sum_{\nu=1}^{k} v_\nu(x) \qquad \sum_{\mu=1}^{k} s_{\nu\mu} \overline{w_\mu(y)}$$

Fast summation

Standard approach: Approximate g by degenerate kernel,

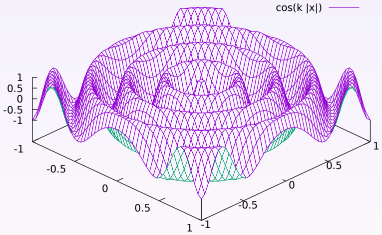
$$\int_{\partial\Omega} g(x,y)u(y)\,dy \approx \sum_{\nu=1}^{k} v_\nu(x) \int_{\partial\Omega} \sum_{\mu=1}^{k} s_{\nu\mu} \overline{w_\mu(y)} u(y)\,dy.$$

Fast summation

Standard approach: Approximate g by degenerate kernel,

$$\int_{\partial\Omega} g(x,y)u(y)\,dy \approx \sum_{\nu=1}^{k} v_\nu(x) \int_{\partial\Omega} \sum_{\mu=1}^{k} s_{\nu\mu}\overline{w_\mu(y)}u(y)\,dy.$$

Challenge: Helmholtz kernel oscillates rapidly if κ large.



$$g(x,y) = \frac{\exp(\iota\kappa\|x-y\|)}{\|x-y\|}.$$

Consequence: Standard approximation schemes require $k \gg 1$.

Experiment: $\mathcal{H}^2$-matrix compression

Helmholtz matrix approximated by quasi-optimal $\mathcal{H}^2$-matrix.

n	κ	k	k/κ	M/n
2048	8	33	4.1	12.5
4608	12	47	3.9	15.4
8192	16	61	3.8	18.3
18432	24	95	4.0	22.2
32768	32	132	4.1	25.4
73728	48	263	5.5	30.1
131072	64	333	5.2	36.0

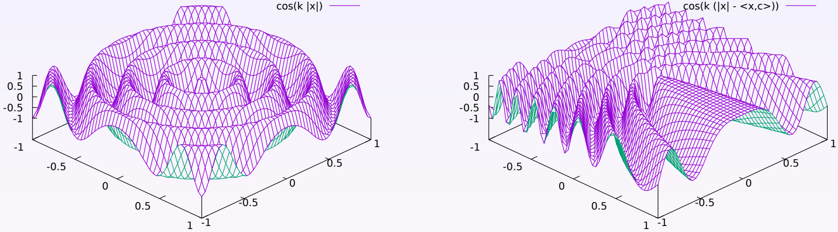
Observation: Maximal rank appears proportional to wave number κ, storage looks like $\mathcal{O}(n \log^2 \kappa)$.

Overview

Directional approximation

Idea: Divide spherical wave by plane wave in direction c, $\|c\| = 1$:

$$g(x,y) = \exp(\iota\kappa\langle x-y, c\rangle)\underbrace{\frac{\exp(\iota\kappa(\|x-y\| - \langle x-y, c\rangle))}{4\pi\|x-y\|}}_{=g_c(x,y)}.$$



✎ Brandt (1991), Engquist/Ying (2007), Messner/Schanz/Darve (2012), Bebendorf/Kuske/Venn (2015)

Directional interpolation

Approach: Apply interpolation to smoothed function g_c.

$$\tilde{g}_{\tau\sigma,c}(x,y) := \sum_{\nu,\mu=1}^{k} \mathcal{L}_{\tau,\nu}(x) g_c(\xi_{\tau,\nu}, \xi_{\sigma,\mu}) \mathcal{L}_{\sigma,\mu}(y).$$

Directional approximation in domain $\tau \times \sigma$ given by

$$\begin{aligned} g(x,y) &= \exp(\iota\kappa\langle x-y, c\rangle) g_c(x,y) \approx \exp(\iota\kappa\langle x-y, c\rangle) \tilde{g}_{\tau\sigma,c}(x,y) \\ &= \sum_{\nu,\mu=1}^{k} \underbrace{\exp(\iota\kappa\langle x, c\rangle)\mathcal{L}_{\tau,\nu}(x)}_{=:\mathcal{L}_{\tau c,\nu}(x)} g_c(\xi_{\tau,\nu}, \xi_{\sigma\mu}) \underbrace{\exp(-\iota\kappa\langle y, c\rangle)\mathcal{L}_{\sigma,\mu}(y)}_{=:\overline{\mathcal{L}_{\sigma c,\mu}(y)}} \\ &= \sum_{\nu,\mu=1}^{k} \mathcal{L}_{\tau c,\nu}(x) g_c(\xi_{\tau,\nu}, \xi_{\sigma,\mu}) \overline{\mathcal{L}_{\sigma c,\mu}(y)} =: \tilde{g}_{\tau\sigma}(x,y). \end{aligned}$$

Matrix factorization

Goal: Approximate Galerkin matrix $G \in \mathbb{C}^{\mathcal{I}\times\mathcal{I}}$ given by

$$g_{ij} = \int_{\partial\Omega} \varphi_i(x) \int_{\partial\Omega} g(x,y)\varphi_j(y)\,dy\,dx.$$

Idea: If $g|_{\tau\times\sigma} \approx \tilde{g}_{\tau\sigma}$, choose

$$\widehat{\tau} := \{i \in \mathcal{I} \;:\; \operatorname{supp}\varphi_i \subseteq \tau\}, \qquad \widehat{\sigma} := \{j \in \mathcal{I} \;:\; \operatorname{supp}\varphi_j \subseteq \sigma\}$$

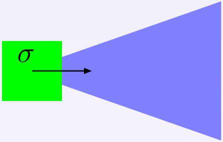
and approximate $G|_{\hat{\tau}\times\hat{\sigma}}$ by discretizing $\tilde{g}_{\tau\sigma}$ instead of g.

Result: Factorization $G|_{\hat{\tau}\times\hat{\sigma}} \approx V_{\tau c} S_{\tau\sigma} V_{\sigma c}^*$ with

$$v_{\tau c,i\nu} = \int_{\partial\Omega} \varphi_i(x)\mathcal{L}_{\tau c,\nu}(x)\,dx, \qquad s_{\tau\sigma,\nu\mu} = g_c(\xi_{\tau,\nu}, \xi_{\sigma,\mu}).$$

Directions

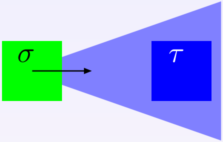
Kernel function $g_c(\cdot, y)$ for $y \in \sigma$
is smooth in a cone with axis c.



Directions

Kernel function $g_c(\cdot, y)$ for $y \in \sigma$ is smooth in a cone with axis c.

Interactions with clusters τ inside the cone can be approximated.

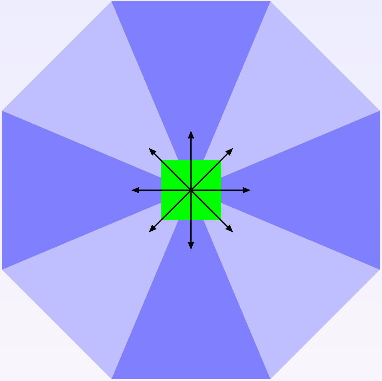


Directions

Kernel function $g_c(\cdot, y)$ for $y \in \sigma$ is smooth in a cone with axis c.

Interactions with clusters τ inside the cone can be approximated.

Multiple cones have to be used to cover the entire domain.

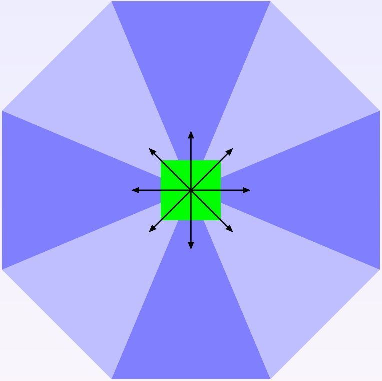


Directions

Kernel function $g_c(\cdot, y)$ for $y \in \sigma$ is smooth in a cone with axis c.

Interactions with clusters τ inside the cone can be approximated.

Multiple cones have to be used to cover the entire domain.



Approach: Choose fixed set of directions for each σ.
Analysis shows that $\mathcal{O}(1 + \kappa^2 \operatorname{diam}^2(\tau))$ directions are sufficient.

Nested basis

Problem: Storing $V_{\tau c}$ for all τ, c requires $\sim n^2$ units of memory.

Solution: Organize domains τ in a cluster tree such that

$$\operatorname{sons}(\tau) \neq \emptyset \qquad \Longrightarrow \qquad \hat{\tau} = \bigcup_{\tau' \in \operatorname{sons}(\tau)} \hat{\tau}',$$

and approximate Lagrange polynomials by weighted interpolation

$$\mathcal{L}_{\tau c,\nu} \approx \sum_{\nu'=1}^{k} e_{\tau'\tau c,\nu'\nu} \, \mathcal{L}_{\tau' c',\nu'}.$$

Result: Nested representation $V_{\tau c}|_{\hat{\tau}'} \approx V_{\tau' c'} E_{\tau'\tau c}$ if $\tau' \in \operatorname{sons}(\tau)$.

✎ Messner/Schanz/Darve (2012), B./Melenk (2015)

$\mathcal{DH}^2$-matrix

Approach:

- Fix set of directions $\mathcal{D}_\tau$ for all clusters τ.
- Choose admissible blocks $(\tau, \sigma, c) \in \mathcal{L}^+_{\mathcal{I}\times\mathcal{I}}$.
- Compute leaf matrices $V_{\tau c} \in \mathbb{C}^{\hat\tau \times k}$ for all leaves τ and all $c \in \mathcal{D}_\tau$.
- Find transfer matrices $E_{\tau'\tau c} \in \mathbb{C}^{k\times k}$ for all $\tau' \in \operatorname{sons}(\tau)$, $c \in \mathcal{D}_\tau$.
- Create coupling matrices $S_{\tau\sigma} \in \mathbb{C}^{k\times k}$ for all $(\tau, \sigma, c) \in \mathcal{L}^+_{\mathcal{I}\times\mathcal{I}}$.

Results:

- Storage $\mathcal{O}(kn + k^2\kappa^2 \log n)$.
- Matrix-vector multiplication in $\mathcal{O}(kn + k^2\kappa^2 \log n)$ operations.
- High-frequency case: $\kappa^2 \sim n$, complexity $\mathcal{O}(nk^2 \log n)$.

✎ B. (2015)

Overview

Approximation of general matrices

Given: Matrix $G \in \mathbb{C}^{\mathcal{I}\times\mathcal{I}}$ and cluster/block structure.

Goal: Find $\mathcal{DH}^2$-matrix approximation of G,
i.e., directional cluster basis $(V_{\tau c})$ and coupling matrices $(S_{\tau\sigma})$.

Orthogonal basis: If we ensure $V_{\tau c}^* V_{\tau c} = I$, the best coupling matrices are given by orthogonal projection

$$G|_{\hat\tau\times\hat\sigma} \approx V_{\tau c} \underbrace{V_{\tau c}^* G|_{\hat\tau\times\hat\sigma} V_{\sigma c}}_{=:S_{\tau\sigma}} V_{\sigma c}^*,$$

Row and column projections can be analyzed separately,

$$\begin{aligned}\|G|_{\hat\tau\times\hat\sigma} - V_{\tau c} S_{\tau\sigma} V_{\sigma c}^*\|_2^2 \leq \|G|_{\hat\tau\times\hat\sigma} - V_{\tau c} V_{\tau c}^* G|_{\hat\tau\times\hat\sigma}\|_2^2 \\ + \|G|_{\hat\tau\times\hat\sigma} - G|_{\hat\tau\times\hat\sigma} V_{\sigma c} V_{\sigma c}^*\|_2^2.\end{aligned}$$

Low-rank structure

Question: What has to be approximated by $V_{\tau c}$?

Answer: All blocks $G|_{\hat{\tau}\times\hat{\sigma}}$ with $(\tau, \sigma, c) \in \mathcal{L}^+_{\mathcal{I}\times\mathcal{I}}$.

Low-rank structure

Question: What has to be approximated by $V_{\tau c}$?

Partial answer: All blocks $G|_{\hat{\tau}\times\hat{\sigma}}$ with $(\tau, \sigma, c) \in \mathcal{L}^+_{\mathcal{I}\times\mathcal{I}}$.

Important: Directional cluster basis is nested, $V_{\tau c}|_{\hat{\tau}'} = V_{\tau' c'} E_{\tau' \tau c}$.
Anything that is not approximated in the son clusters cannot be approximated in the father.

Low-rank structure

Question: What has to be approximated by $V_{\tau c}$?

Partial answer: All blocks $G|_{\hat{\tau}\times\hat{\sigma}}$ with $(\tau, \sigma, c) \in \mathcal{L}^+_{\mathcal{I}\times\mathcal{I}}$.

Important: Directional cluster basis is nested, $V_{\tau c}|_{\hat{\tau}'} = V_{\tau' c'} E_{\tau' \tau c}$.
Anything that is not approximated in the son clusters cannot be approximated in the father.

Complete answer: All blocks $G|_{\hat{\tau}\times\hat{\sigma}^*}$ where

- $(\tau^*, \sigma^*, c^*) \in \mathcal{L}^+_{\mathcal{I}\times\mathcal{I}}$,
- τ is a descendant of τ^*, and
- c approximates c^*.

We collect these tuples in sets $\mathcal{R}_{\tau c} \subseteq \mathcal{L}^+_{\mathcal{I}\times\mathcal{I}}$.

Leaf matrices

Assume that τ is a leaf cluster and that $\mathcal{R}_{\tau c}$ is available for all $c \in \mathcal{D}_\tau$.

Goal: Find orthogonal $V_{\tau c}$ with k columns such that

$$G|_{\hat{\tau}\times\hat{\sigma}^*} \approx V_{\tau c} V_{\tau c}^* G|_{\hat{\tau}\times\hat{\sigma}^*} \qquad \text{for all } (\tau^*, \sigma^*, c^*) \in \mathcal{R}_{\tau c}.$$

Leaf matrices

Assume that τ is a leaf cluster and that $\mathcal{R}_{\tau c}$ is available for all $c \in \mathcal{D}_\tau$.

Goal: Find orthogonal $V_{\tau c}$ with k columns such that

$$G|_{\hat\tau\times\hat\sigma^*} \approx V_{\tau c} V_{\tau c}^* G|_{\hat\tau\times\hat\sigma^*} \qquad \text{for all } (\tau^*, \sigma^*, c^*) \in \mathcal{R}_{\tau c}.$$

Reformulation: Define the submatrix

$$G_{\tau c} := G|_{\hat\tau\times\mathcal{F}_{\tau c}}, \qquad \mathcal{F}_{\tau c} := \bigcup\{\hat\sigma^* \ :\ (\tau^*, \sigma^*, c^*) \in \mathcal{R}_{\tau c}\}$$

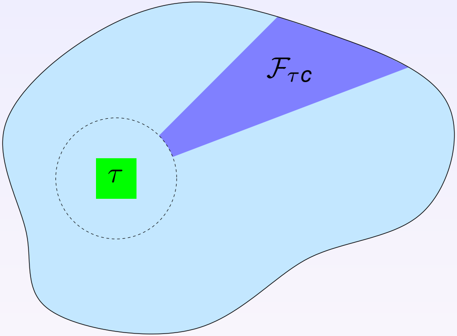
with farfield matrix $G_{\tau c}$ and consider only

$$G_{\tau c} \approx V_{\tau c} V_{\tau c}^* G_{\tau c}.$$

$\to$ Compute SVD of $G_{\tau c}$, remove small singular values.

Directional farfield

Question: What does $\mathcal{F}_{\tau c}$ represent?



Answer: $\mathcal{F}_{\tau c}$ corresponds to basis functions whose supports are

- sufficiently far from τ (i.e., $\mathrm{dist} \geq \mathrm{diam}(\tau), \kappa\, \mathrm{diam}(\tau)^2$), and
- contained in a cone around the direction c.

Transfer matrices

Assume $\operatorname{sons}(\tau) = \{\tau_1, \tau_2\}$, $c_1 \in \mathcal{D}_{\tau_1}$, $c_2 \in \mathcal{D}_{\tau_2}$ approximate $c \in \mathcal{D}_\tau$.

Recursion: Ensure that $V_{\tau_1 c_1}$, $V_{\tau_2 c_2}$ have already been computed.

Reduced matrices given by

$$\widehat{G}_{\tau c} = \begin{pmatrix} V_{\tau_1 c_1}^* G|_{\hat\tau_1 \times \mathcal{F}_{\tau c}} \\ V_{\tau_2 c_2}^* G|_{\hat\tau_2 \times \mathcal{F}_{\tau c}} \end{pmatrix}, \qquad \widehat{V}_{\tau c} = \begin{pmatrix} E_{\tau_1 \tau c} \\ E_{\tau_2 \tau c} \end{pmatrix}$$

lead to

$$\begin{aligned} \|G_{\tau c} - V_{\tau c} V_{\tau c}^* G_{\tau c}\|_2^2 \leq\ & \|G|_{\hat\tau_1 \times \mathcal{F}_{\tau c}} - V_{\tau_1 c_1} V_{\tau_1 c_1}^* G|_{\hat\tau_1 \times \mathcal{F}_{\tau c}}\|_2^2 \\ & + \|G|_{\hat\tau_2 \times \mathcal{F}_{\tau c}} - V_{\tau_2 c_2} V_{\tau_2 c_2}^* G|_{\hat\tau_2 \times \mathcal{F}_{\tau c}}\|_2^2 \\ & + \|\widehat{G}_{\tau c} - \widehat{V}_{\tau c} \widehat{V}_{\tau c}^* \widehat{G}_{\tau c}\|_2^2. \end{aligned}$$

The first two terms correspond to the sons' influence, the last can be controlled via SVD and provides us with the transfer matrices.

$\mathcal{DH}^2$-matrix compression algorithm

Recursion following the cluster tree.

- Set up $\mathcal{R}_{\tau c}$ during the downward pass.
- Construct $V_{\tau c}$, $E_{\tau' \tau c}$, and $V_{\tau c}^* G_{\tau c}$ during the upward pass.

Compared to $\mathcal{H}^2$-matrices:

- Similar complexity estimates.
- Loss of sparsity, some error estimates no longer work.
- Parallelization tricky, large clusters very time-consuming.
- Complexity $\mathcal{O}(n^2 k)$.

Overview

Approximation of $\mathcal{DH}^2$-matrices

Challenge: Compression of a general matrix takes $\mathcal{O}(n^2 k)$ operations.

Idea: Obtain intermediate $\mathcal{DH}^2$-matrix by interpolation and recompress to reduce the rank.

Approximation of $\mathcal{DH}^2$-matrices

Challenge: Compression of a general matrix takes $\mathcal{O}(n^2 k)$ operations.

Idea: Obtain intermediate $\mathcal{DH}^2$-matrix by interpolation and recompress to reduce the rank.

Nested structure: We have

$$(V_{\tau c} S_{\tau\sigma} V_{\sigma c}^*)|_{\hat{\tau}'} = V_{\tau' c'} E_{\tau' \tau c} S_{\tau\sigma} V_{\sigma c}^*,$$

and induction yields

$$G|_{\hat{\tau}\times\hat{\sigma}^*} \approx V_{\tau c} E_{\tau\tau^* c^*} S_{\tau^*\sigma^*} V_{\sigma^* c^*}^* \qquad \text{for all } (\tau^*, \sigma^*, c^*) \in \mathcal{R}_{\tau c}$$

with suitable "long-range" transfer matrices $E_{\tau\tau^* c^*}$.

Approximation of $\mathcal{DH}^2$-matrices

Challenge: Compression of a general matrix takes $\mathcal{O}(n^2 k)$ operations.

Idea: Obtain intermediate $\mathcal{DH}^2$-matrix by interpolation and recompress to reduce the rank.

Nested structure: We have

$$(V_{\tau c} S_{\tau\sigma} V_{\sigma c}^*)|_{\hat{\tau}'} = V_{\tau' c'} E_{\tau'\tau c} S_{\tau\sigma} V_{\sigma c}^*,$$

and induction yields

$$G|_{\hat{\tau}\times\hat{\sigma}^*} \approx V_{\tau c} E_{\tau\tau^* c^*} S_{\tau^*\sigma^*} V_{\sigma^* c^*}^* \qquad \text{for all } (\tau^*, \sigma^*, c^*) \in \mathcal{R}_{\tau c}$$

with suitable “long-range” transfer matrices $E_{\tau\tau^* c^*}$.

Result: Intermediate low-rank approximation $G_{\tau c} \approx V_{\tau c} B_{\tau c}^*$.

Weighted truncation

Goal: Find optimized low-rank factorization of intermediate approximation

$$G_{\tau c} \approx V_{\tau c} B_{\tau c}^*.$$

Idea: Since approximation depends only on singular values and left singular vectors, use skinny Householder factorization

$$B_{\tau c} = P_{\tau c} Z_{\tau c}$$

with weight matrix $Z_{\tau c} \in \mathbb{C}^{k \times k}$ and work with

$$G_{\tau c} P_{\tau c}^* \approx V_{\tau c} Z_{\tau c}^*.$$

Result: If all weights $(Z_{\tau c})_{\tau, c}$ are available, compression requires only $\mathcal{O}(nk^2 \log n)$ operations instead of $\sim n^2 k$.

Weight matrices

Goal: Compute weight matrices with $B_{\tau c} = P_{\tau c} Z_{\tau c}$.

Idea: Take advantage of nested structure, $\tau \in \mathrm{sons}(\tau^+)$.

$$B_{\tau c} = \begin{pmatrix} V_{\sigma_1 c} S^*_{\tau\sigma_1} \\ \vdots \\ V_{\sigma_s c} S^*_{\tau\sigma_s} \\ B_{\tau^+ c_1^+} E^*_{\tau\tau^+ c_1^+} \\ \vdots \\ B_{\tau^* c_\ell^+} E^*_{\tau\tau^+ c_\ell^+} \end{pmatrix}$$

with blocks $(\tau, \sigma_1, c), \dots, (\tau, \sigma_s, c) \in \mathcal{L}^+_{\mathcal{I}\times\mathcal{I}}$, and the father's directions $c_1^+, \dots, c_\ell^+ \in \mathcal{D}_{\tau^+}$ approximating c.

Weight matrices

Goal: Compute weight matrices with $B_{\tau c} = P_{\tau c} Z_{\tau c}$.

Idea: Take advantage of nested structure, $\tau \in \mathrm{sons}(\tau^+)$.

$$B_{\tau c} = \begin{pmatrix} V_{\sigma_1 c} S^*_{\tau\sigma_1} \\ \vdots \\ V_{\sigma_s c} S^*_{\tau\sigma_s} \\ B_{\tau^+ c_1^+} E^*_{\tau\tau^+ c_1^+} \\ \vdots \\ B_{\tau^* c_\ell^+} E^*_{\tau\tau^+ c_\ell^+} \end{pmatrix} = \begin{pmatrix} Q_{\sigma_1 c} R_{\sigma_1 c} S^*_{\tau\sigma_1} \\ \vdots \\ Q_{\sigma_s c} R_{\sigma_s c} S^*_{\tau\sigma_s} \\ P_{\tau^+ c_1^+} Z_{\tau^+ c_1^+} E^*_{\tau\tau^+ c_1^+} \\ \vdots \\ P_{\tau^+ c_\ell^+} Z_{\tau^+ c_\ell^+} E^*_{\tau\tau^+ c_\ell^+} \end{pmatrix}$$

with blocks $(\tau, \sigma_1, c), \ldots, (\tau, \sigma_s, c) \in \mathcal{L}^+_{\mathcal{I}\times\mathcal{I}}$, and the father's directions $c_1^+, \ldots, c_\ell^+ \in \mathcal{D}_{\tau^+}$ approximating c.

Weight matrices

Goal: Compute weight matrices with $B_{\tau c} = P_{\tau c} Z_{\tau c}$.

Idea: Take advantage of nested structure, $\tau \in \mathrm{sons}(\tau^+)$.

$$B_{\tau c} = \begin{pmatrix} V_{\sigma_1 c} S^*_{\tau\sigma_1} \\ \vdots \\ V_{\sigma_s c} S^*_{\tau\sigma_s} \\ B_{\tau^+ c_1^+} E^*_{\tau\tau^+ c_1^+} \\ \vdots \\ B_{\tau^* c_\ell^+} E^*_{\tau\tau^+ c_\ell^+} \end{pmatrix} = \begin{pmatrix} Q_{\sigma_1 c} R_{\sigma_1 c} S^*_{\tau\sigma_1} \\ \vdots \\ Q_{\sigma_s c} R_{\sigma_s c} S^*_{\tau\sigma_s} \\ P_{\tau^+ c_1^+} Z_{\tau^+ c_1^+} E^*_{\tau\tau^+ c_1^+} \\ \vdots \\ P_{\tau^+ c_\ell^+} Z_{\tau^+ c_\ell^+} E^*_{\tau\tau^+ c_\ell^+} \end{pmatrix}$$

with blocks $(\tau, \sigma_1, c), \ldots, (\tau, \sigma_s, c) \in \mathcal{L}^+_{\mathcal{I}\times\mathcal{I}}$, and the father's directions $c_1^+, \ldots, c_\ell^+ \in \mathcal{D}_{\tau^+}$ approximating c.

Result: Construction of all $(Z_{\tau c})_{\tau,c}$ in $\mathcal{O}(nk^2 \log n)$ operations.

Recompression algorithm

Directional interpolation yields intermediate $\mathcal{DH}^2$-matrix approximation that is used implicitly and not stored completely.

Farfield matrices $G_{\tau c}$ are replaced by low-rank factorizations $V_{\tau c} Z_{\tau c}^*$ with suitable weight matrices.

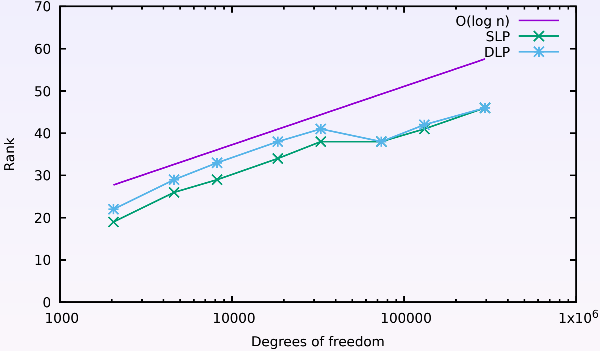
Weight matrices constructed by top-down recursion and Householder factorization.

Result: Compression in $\mathcal{O}(nk^2 \log n)$ operations.

Overview

Experiment: Single and double layer potential

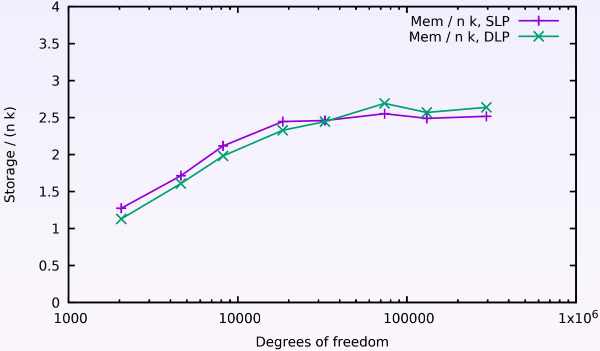
Approximate the matrix G corresponding to SLP and DLP operators for the unit sphere.



Result: Maximal rank grows like $\mathcal{O}(\log n)$.

Experiment: Single and double layer potential

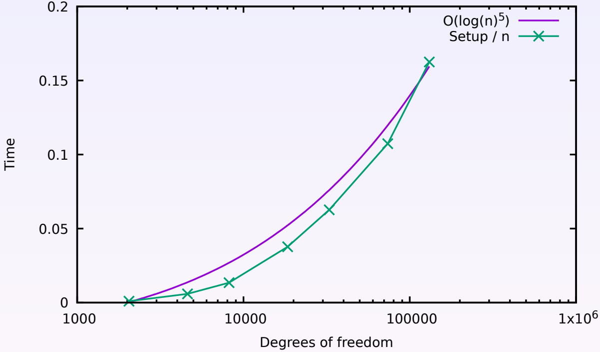
Approximate the matrix G corresponding to SLP and DLP operators for the unit sphere.



Result: Storage grows like $\mathcal{O}(nk)$.

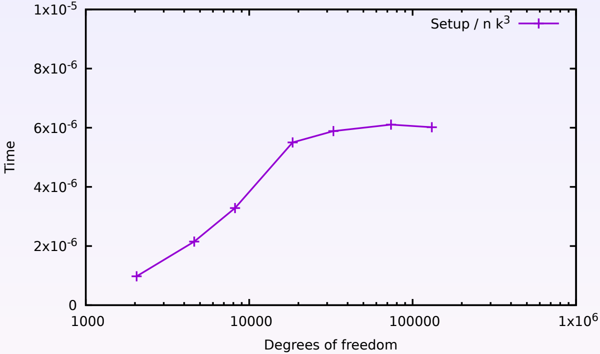
Experiment: Recompression

Approximate the matrix G corresponding to the SLP operator by directional interpolation, followed by algebraic recompression.



Experiment: Recompression

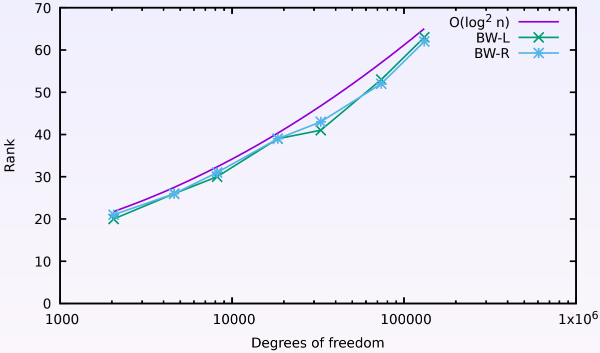
Approximate the matrix G corresponding to the SLP operator by directional interpolation, followed by algebraic recompression.



Result: Time appears to grow like $\mathcal{O}(nk^3)$.

Experiment: Preconditioner

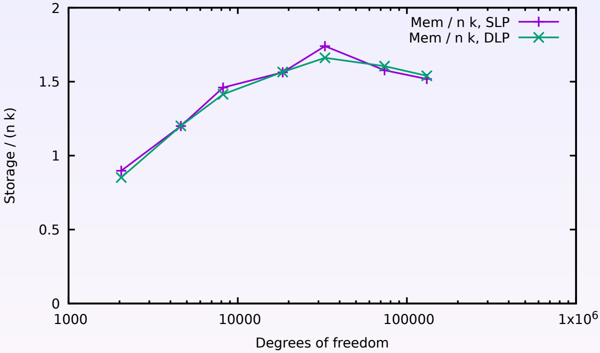
Approximate the LR factorization of the Brakhage-Werner operator.



Result: Maximal rank grows like $\mathcal{O}(\log^2 n)$.

Experiment: Preconditioner

Approximate the LR factorization of the Brakhage-Werner operator.



Result: Storage grows like $\mathcal{O}(nk)$.

Conclusion

Challenge: Approximate high-frequency Helmholtz operators.

Approach: Directional approximation leads to $\mathcal{DH}^2$-matrix.
$\rightarrow$ Complexity $\mathcal{O}(nk^2 \log n)$.

Compression: Recursive algorithm, $\mathcal{O}(n^2 k)$ operations.

Recompression: Combine directional interpolation with on-the-fly compression, $\mathcal{O}(nk^2 \log n)$ operations.

Work in progress: Arithmetic operations, inclusion in H2Lib package

`http://www.h2lib.org`