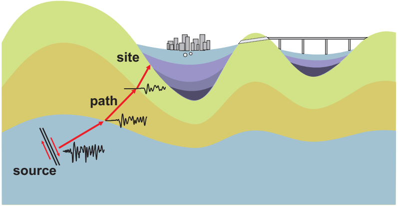
A Fast Direct Solver Based on the Boundary Element Method to Model 3-D Elastic Waves in Large Domains

S. Chaillat-Loseille and P. Ciarlet

POEMS (CNRS-INRIA-ENSTA ParisTech), Université Paris-Saclay, France

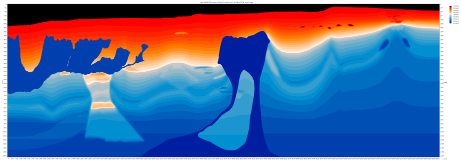
OILWS2: Full Waveform Inversion and Velocity Analysis,
Los Angeles, May 4th 2017

Modelling of elastic wave propagation in large and unbounded domains



- Site effects
- Soil-structure interaction

- Efficient tool to speed up forward solvers for 3D elastic Full Waveform Inversion?



Problems solved in frequency domain (Fourier transform/Convolution Quadrature Method to obtain results in time domain if needed)

$\Rightarrow$ What can we do with the Boundary Element Method?

Principle of the Boundary Element Method

Boundary-value problem over Ω : $\begin{cases} \Delta^* \boldsymbol{u} + \rho\omega^2 \boldsymbol{u} + \mathbf{F} = 0 & \text{in } \Omega, \\ \boldsymbol{u} = g_1 \quad \text{on } \partial\Omega_u, & \mathbf{T}\boldsymbol{u} = g_2 \quad \text{on } \partial\Omega_t. \end{cases}$

- Traction operator: $\mathbf{T} = 2\mu \dfrac{\partial}{\partial \mathbf{n}} + \lambda \mathbf{n}\,\mathrm{div} + \mu\, \mathbf{n} \times \mathbf{curl}$
- $\Delta^* = \mu \boldsymbol{\Delta} + (\lambda + \mu) \boldsymbol{\nabla}\,\mathrm{div}$
- $\mathbf{F}$: body force (thermal effects, initial stresses or strains, ...)

Deriving a Boundary Integral Representation

- Reciprocity identity: $\boldsymbol{u}, \boldsymbol{v}$ 2 elastodynamic states

$$\int_\Omega \Big(\Delta^* \boldsymbol{u}.\boldsymbol{v} - \Delta^* \boldsymbol{v}.\boldsymbol{u} \Big) dV = \int_{\partial\Omega} (\mathbf{T}\boldsymbol{u}.\boldsymbol{v} - \mathbf{T}\boldsymbol{v}.\boldsymbol{u}) dS$$

- Fundamental solution: unit point force at x along k-direction

$$\Delta^* \mathbf{U} + \rho\omega^2 \mathbf{U} + \delta(\mathbf{y} - \mathbf{x}) \mathbf{e}_k = 0$$

- Integral Representation formula: $\mathbf{x} \notin \partial\Omega$

$$\kappa \boldsymbol{u}_k(\mathbf{x}) = \int_\Omega \mathbf{U}_{ki}(\mathbf{x}, \mathbf{y}) F_i(\mathbf{y}) dV_y + \int_{\partial\Omega} \Big(\mathbf{T}_{ki}(\mathbf{x}, \mathbf{y}) \boldsymbol{u}_i(\mathbf{y}) - \mathbf{U}_{ki}(\mathbf{x}, \mathbf{y}) \mathbf{t}_i(\mathbf{y}) \Big) dS_y$$

$$(\kappa = 1 \; if \; \mathbf{x} \in \Omega, \quad \kappa = 0 \; if \; \mathbf{x} \notin \Omega)$$

... Principle of the Boundary Element Method

$$\kappa \boldsymbol{u}_k(\mathbf{x}) = \cancel{\int_{\Omega} \mathbf{U}_{ki}(\mathbf{x},\mathbf{y}) F_i(\mathbf{y}) dV_y} + \int_{\partial\Omega} \big(\mathbf{T}_{ki}(\mathbf{x},\mathbf{y}) \boldsymbol{u}_i(\mathbf{y}) - \mathbf{U}_{ki}(\mathbf{x},\mathbf{y}) \mathbf{t}_i(\mathbf{y}) \big) dS_y$$

Boundary Integral Equation (BIE)

- Representation formula holds only for $\mathbf{x}$ not on the boundary
- $\mathbf{U}$ and $\mathbf{T}$ have a strong singularity for $\mathbf{y} = \mathbf{x}$
- Boundary integral equation relies upon a limiting process
- For surface point load:

$$-\frac{1}{2}\boldsymbol{u}(\mathbf{x}) + \int_{\partial\Omega} \mathbf{T}(\mathbf{x},\mathbf{y}) \boldsymbol{u}(\mathbf{y}) dS_y = \int_{\partial\Omega} \mathbf{U}(\mathbf{x},\mathbf{y}) \mathbf{t}^F(\mathbf{y}) dS_y$$

Two main steps of the boundary element method (BEM)

1. Solve the boundary integral equation on the boundary
   - Linear system to solve
   - Unknowns only on the boundary
2. Integral representation to evaluate the quantities at interior points
   - Cost reduced to a matrix-vector product

M. Bonnet. *Boundary Integral Equation Methods for Solids and Fluids*, 1999.

Pros and cons of BEM for elastic waves

$$-\frac{1}{2}\boldsymbol{u}(\mathbf{x}) + \int_{\partial\Omega} \mathbf{T}(\mathbf{x},\mathbf{y})\boldsymbol{u}(\mathbf{y})dS_y = \int_{\partial\Omega} \mathbf{U}(\mathbf{x},\mathbf{y})\mathbf{t}^F(\mathbf{y})dS_y$$

BEM discretization ⇒ fully-populated system of linear equations

Comparison with volume methods

Domain methods (FEM, SEM, ...)	BEM
→ Domain mesh	→ Surface mesh (i.e. reduced dim.)
→ Approx. radiation conditions	→ Exact radiation conditions
→ Sparse matrix	→ Fully-populated matrix

Limitations and advantages depend on the fundamental solutions used in the BIE

- They are usually defined for homogeneous domains
  ⇒ Analytic expression for an isotropic elastic full-space
  ⇒ All interfaces between homogeneous domains are meshed
  ⇒ Layered medium with non flat surfaces can be considered
- More involved expressions for an elastic (layered) half-space, transverse isotropic elastic full-space

⇒ Adequate for large (unbounded) media, simple (linear) properties. Topography is easily taken into account

Fast BEMs

⇒ Fully-populated BEM influence matrix: severe limiting factor

- Solution with a direct solver (LU factorization)
  - CPU: Factorization $O(N^3)$, Solution $O(N^2)$ and Memory $O(N^2)$
- Solution with an iterative solver (GMRES)
  - CPU: $O(N_{iter}N^2)$, memory: $O(N^2)$; N: $\#$ DOFs on the boundary

⇒ BEM very limited in terms of heterogeneities and frequency range

Acceleration of the BEM with the Fast Multipole Method (FMM)

- Intrinsically based on iterative solvers: $O(N_{iter} \times N \log N)$
- Fast, approximate method evaluation of matrix-vector product
- Number of iterations very large for seismology-oriented problems
- Definition of an efficient preconditioner difficult
- Extension to complex fundamental solutions is involved

S. Chaillat, M. Bonnet, Wave Motion, 2013.

Need of an alternative approach: fast LU factorization of the BEM matrix

⇒ direct solver based on $\mathcal{H}$-matrices and low-rank approximations

⇒ Preliminary study: what can we expect in terms of memory requirements for 3D isotropic elastodynamics?

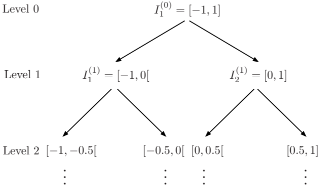
How to reduce memory and CPU costs?

Illustrative simple example: 3D Helmholtz fundamental solution

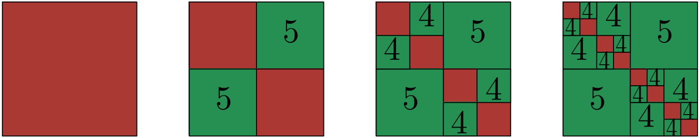
- Matrix: discretization of $[-1,1]$ with $N = 1000$ and $k = 10\pi$

$$\mathbb{G}_{ij} = \frac{\exp(ik|\mathbf{x}_i - \mathbf{y}_j|)}{4\pi|\mathbf{x}_i - \mathbf{y}_j|} \quad \text{with } \mathbf{x},\mathbf{y} \in [-1,1]\times[-1,1]$$

- Complete matrix is full rank: clustering of the discretization points



- Numerical ranks of blocks of $\mathbb{G}$ after various levels of clustering



Entire matrix *1 level* *2 levels* *3 levels*

Hierarchical representation of a BEM matrix

Main idea: approximation of the initial system by a data-sparse one:

redundant informations are removed (similarly to image compression)

if storage of $\mathbb{A}$ is $O(N^2)$, storage of rank-k approx. of $\mathbb{A} = \mathbb{U}\mathbb{V}^t$ is $O(2Nk)$

Representation of the BEM matrix with a $\mathcal{H}$-matrix structure :

- Clustering of degrees of freedom (based on geometry): Clusters of indices in matrix related to distances between nodes
- Definition of blocks $\mathbb{A}_{\sigma\times\tau}$ (σ,τ: 2 nodes of the binary tree): iteratively subdivide a matrix into 4 submatrices if we know a priori that it cannot be approximated by a low-rank matrix

$\Rightarrow$ Reduction of memory costs: low-rank approx. of some blocks

Representation of the 3D elastodynamic BEM matrix

- Is this representation efficient for 3D elastodynamic BEM?
- How can we determine *a priori* which blocks are low-rank?

Theory for Asymptotically smooth kernels

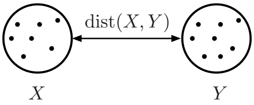
- Taylor series: to build theoretically a low rank approximation
- Asymptotic smoothness: to prove exponential convergence of the Taylor series ($\mathbf{x} \in X$ and $\mathbf{y} \in Y$)

$s(.,.)$ is asymptotically smooth if $\exists c_1, c_2$ and $\sigma \in \mathbb{N}$ (singularity degree) such that $\forall z \in \{x_j, y_j\}$ and $n \in \mathbb{N}$, $\forall \mathbf{x} \neq \mathbf{y}$

$$|\partial_{\mathbf{z}}^n s(\mathbf{x}, \mathbf{y})| \leq n! c_1 (c_2 ||\mathbf{x} - \mathbf{y}||)^{-n-\sigma}$$

Ex: $f(\mathbf{x}, \mathbf{y}) = (4\pi ||\mathbf{x} - \mathbf{y}||)^{-1}$ is asymptotically smooth

- But the exponential convergence is conditioned by the η-admissibility (criterion to determine *a priori* low-rank blocks)



$$\text{min}\Big(\text{diam}\, X, \text{diam}\, Y\Big) < \eta \times \text{dist}(X, Y)$$

η is a parameter and $\eta < c_2$ to obtain the convergence

Hackbusch. Hierarchical Matrices: algorithms and analysis, Springer, 2015.

$\Rightarrow$ Efficiency of $\mathcal{H}$-matrix repres. for asymptotically smooth kernels

What can we expect for the Helmholtz kernel?

- Taylor series: to build theoretically a low rank approximation
- Estimate used to prove exponential convergence of the Taylor series ($\mathbf{x} \in X$ and $\mathbf{y} \in Y$)

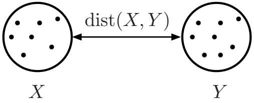
$$G(\mathbf{x},\mathbf{y}) = \exp(ik||\mathbf{x}-\mathbf{y}||) f(\mathbf{x},\mathbf{y})$$

$\exists c_1, c_2$ and $\sigma \in \mathbb{N}$ (singularity degree) such that $\forall z \in \{x_j, y_j\}$ and $n \in \mathbb{N}$

$$|\partial_{\mathbf{z}}^n G(\mathbf{x},\mathbf{y})| \leq n! c_1 (1 + k||\mathbf{x}-\mathbf{y}||)^n (c_2 ||\mathbf{x}-\mathbf{y}||)^{-n-\sigma}$$

because $f(\mathbf{x},\mathbf{y}) = (4\pi||\mathbf{x}-\mathbf{y}||)^{-1}$ is asymptotically smooth

- The exponential convergence is now conditioned by the η_k-admissibility (criterion to determine *a priori* low-rank blocks)



$$\min\Big(\operatorname{diam} X, \operatorname{diam} Y\Big) < \eta(k) \times \operatorname{dist}(X,Y)$$

- Low-frequency regime: η_k-admissibility similar to η-admissibility.
- Higher frequencies: criterion depends linearly on wavenumber.

Adopted point of view for 3D elastodynamics

$$U(\mathbf{x},\mathbf{y}) = \frac{1}{\rho\omega^2}\left(\mathbf{curl}\,\mathbf{curl}_{\mathbf{x}}\,\{G(k_s,\mathbf{x}-\mathbf{y})\,\mathrm{I}_3\} - \boldsymbol{\nabla}_{\mathbf{x}}\,\mathrm{div}_{\mathbf{x}}\,\{G(k_p,\mathbf{x}-\mathbf{y})\mathrm{I}_3\}\right)$$

Similar estimates for elastodynamics (combination of derivatives of G)

Are $\mathcal{H}$-matrix repres. of the 3D elastodynamic BEM matrix efficient?

- Low-frequency regime: η_k-admissibility similar to η-admissibility
  $\Rightarrow$ the method should be as efficient as for asymptotically kernels
- Higher frequencies: criterion depends linearly on ω
  $\Rightarrow$ more involved methods have been proposed to handle this case (directional approaches, ...)

Our approach:

- Standard $\mathcal{H}$-matrices are easier to implement for engineers
- We keep η fixed (similarly to asymptotically smooth kernels)

$$\mathbb{A}_{\sigma\times\tau} \text{ is a priori low-rank} \quad \text{if } \min\Big(\mathrm{diam}(\sigma),\mathrm{diam}(\tau)\Big) < \eta \times \mathrm{dist}(\sigma,\tau)$$

Study of Taylor expansion $\rightarrow$ memory savings will not be optimal.
What can we expect? Frequency range where it can be a useful tool?

Estimated memory requirements

Storage estimate: with N # DOFs

$$N_{st} \leq C \max(r^{\max}, N_{\text{leaf}}) N \log N$$

- $N_{\text{leaf}} (= 100)$: parameter to stop the DOF clustering
- $r^{\max}$: maximum numerical rank observed among all the η-admissible blocks

Estimated maximum numerical rank

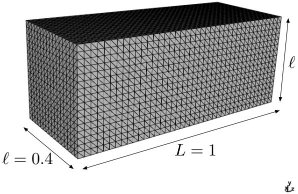
- Study of the Taylor expansion $\rightarrow$ rank depends on ω
- If the circular frequency ω is fixed, the rank should stay constant while N increases

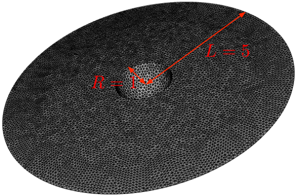
$$N_{st} = O(N \log N)$$

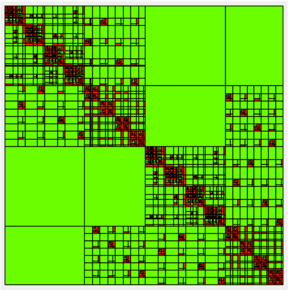
- If the density of points per S-wavelength is fixed, we can show that the maximum rank of the blocks is expected to grow linearly with the frequency until the high-frequency regime is achieved
  - $h = O(\lambda_S) = O(\omega^{-1})$ + BEM mesh $\rightarrow h = O(N^{-2})$
  - The maximum numerical rank is thus expected $O(N^{1/2})$

$$N_{st} \leq O(N^{3/2} \log N)$$

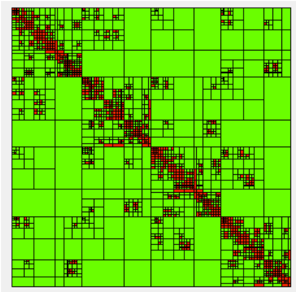
Structure of the matrix depends only on geometry and admissibility criteria ($N_{\text{leaf}} = 100$, $\eta = 3$). Red/green: full/low-rank block







$N = 36\,870$



$N = 27\,903$

Numerical rank for a fixed frequency

$$-\frac{1}{2}\boldsymbol{u}(\mathbf{x}) + \int_{\partial\Omega} \mathbf{T}(\mathbf{x},\mathbf{y})\boldsymbol{u}(\mathbf{y})dS_y = \int_{\partial\Omega} \mathbf{U}(\mathbf{x},\mathbf{y})\mathbf{t}^F(\mathbf{y})dS_y$$

Maximum numerical rank observed for a fixed frequency, i.e. while increasing the density of points per λ_s (obstacle: sphere of diam d)

Single-layer potential $\mathbf{U}$

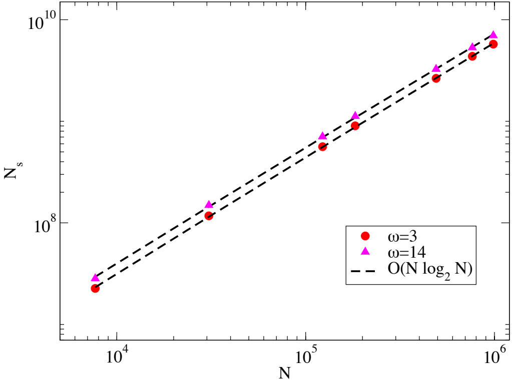
N	7 686	30 726	122 886	183 099	490 629	763 638	985 818
$d=\lambda_s$	39	39	39	37	39	40	39
$d=4\lambda_s$	73	75	76	67	76	76	76

Double-layer potential $\mathbf{T}$

N	7 686	30 726	122 886	183 099	490 629	763 638	985 818
$d=\lambda_s$	37	37	38	36	38	39	39
$d=4\lambda_s$	70	74	74	66	75	75	76

Memory requirements for a fixed frequency

Memory requirements observed for a fixed frequency, i.e. while increasing the density of points per S-wavelength (obstacle: sphere)



$\omega = 3 \rightarrow d = \lambda_s$
$\omega = 14 \rightarrow d = 4\lambda_s$

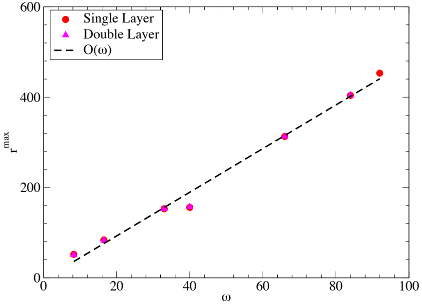
$\Rightarrow$ Numerical results in very good agreement with theoretical estimates.

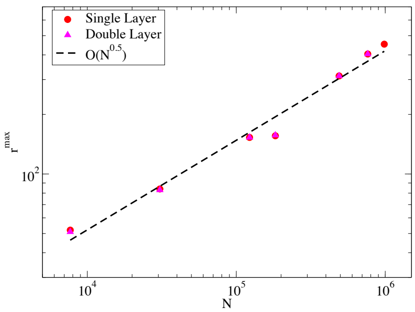
What about the dependance with the circular frequency?

Numerical rank for a fixed density of points

Maximum numerical rank observed for a fixed density of points per λ_s, i.e. while increasing the frequency with N (obstacle: sphere of diam d)

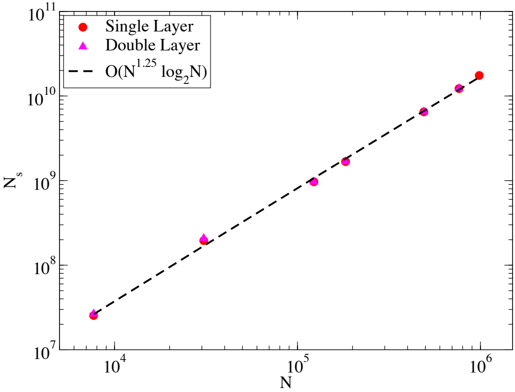
N	7 686	30 726	122 886	183 099	490 629	763 638	985 818
$d = n\lambda_s$	2.6	5.2	10.5	12.6	21	26.6	29.1





$\Rightarrow$ Good agreement between theoretical estimates and numerical results

Memory requirements for a fixed density



$$N_{st} \leq O(N^{3/2} \log N)$$

⇒ Theoretical estimates is overestimated the memory requirements because it uses the maximum rank only

Principle of $\mathcal{H}$-LU factorization

Hierarchical block structure: recursion on 2×2 block matrices

$$\begin{pmatrix} \mathbf{A}_{\tau_1\tau_1} & \mathbf{A}_{\tau_1\tau_2} \\ \mathbf{A}_{\tau_2\tau_1} & \mathbf{A}_{\tau_2\tau_2} \end{pmatrix} = \begin{pmatrix} \mathbf{L}_{\tau_1\tau_1} & \mathbf{0} \\ \mathbf{L}_{\tau_2\tau_1} & \mathbf{L}_{\tau_2\tau_2} \end{pmatrix} \begin{pmatrix} \mathbf{U}_{\tau_1\tau_1} & \mathbf{U}_{\tau_1\tau_2} \\ \mathbf{0} & \mathbf{U}_{\tau_2\tau_2} \end{pmatrix}$$

1. LU decomposition to compute: $\mathbf{L}_{\tau_1\tau_1}$ and $\mathbf{U}_{\tau_1\tau_1}$
2. Compute $\mathbf{U}_{\tau_1\tau_2}$ from $\mathbf{A}_{\tau_1\tau_2} = \mathbf{L}_{\tau_1\tau_1}\mathbf{U}_{\tau_1\tau_2}$
3. Compute $\mathbf{L}_{\tau_2\tau_1}$ from $\mathbf{A}_{\tau_2\tau_1} = \mathbf{L}_{\tau_2\tau_1}\mathbf{U}_{\tau_1\tau_1}$
4. LU decomposition to compute: $\mathbf{A}_{\tau_2\tau_2} - \mathbf{L}_{\tau_2\tau_1}\mathbf{U}_{\tau_1\tau_2} = \mathbf{L}_{\tau_2\tau_2}\mathbf{U}_{\tau_2\tau_2}$

M. Bebendorf. Computing, 2005.

Error estimate to certify the results (fixed density of points)

$$\frac{||b - Ax_0||_2}{||b||_2} \leq \frac{1}{||b||_2}(||b - A_{\mathcal{H}}x_0||_2 + ||A_{\mathcal{H}} - A||_F||x_0||_2) := \mathcal{I}(\delta_{\mathcal{H},F};\delta)$$

# DOFs	ε_{ACA}, ε_{LU}	Single Layer $\mathcal{I}(\delta_{\mathcal{H},F};\delta)$	Single Layer $\frac{\|\|b-Ax_0\|\|_2}{\|\|b\|\|_2}$	Double Layer $\mathcal{I}(\delta_{\mathcal{H},F};\delta)$	Double Layer $\frac{\|\|b-Ax_0\|\|_2}{\|\|b\|\|_2}$
1 926	10^{-6}, 10^{-4}	$3.41\ 10^{-15}$	$1.52\ 10^{-14}$	$3.35\ 10^{-15}$	$3.35\ 10^{-15}$
7 686	10^{-6}, 10^{-4}	$3.99\ 10^{-5}$	$2.21\ 10^{-5}$	$2.20\ 10^{-5}$	$1.09\ 10^{-5}$
30 726	10^{-6}, 10^{-4}	$1.40\ 10^{-4}$	$6.96\ 10^{-5}$	$8.60\ 10^{-5}$	$4.43\ 10^{-5}$
122 886	10^{-6}, 10^{-4}	$9.19\ 10^{-3}$	$8.80\ 10^{-3}$	$3.86\ 10^{-4}$	$1.28\ 10^{-4}$

Half-space subjected to point load

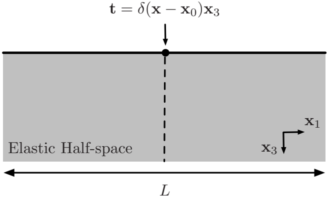
Displacement due to a point force

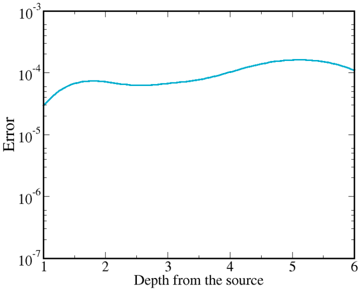
Resulting displacements $u_i(\mathbf{x})$ correspond to fundamental solution $U_i^3(\mathbf{x}, \mathbf{x}_0)$ of the elastic half-space.

S. Chaillat and M. Bonnet, JCP, 2014.

Validation: Comparison with fundamental solution

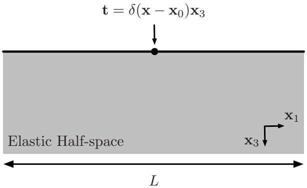
- Density: 10 pts / λ_s
- Accuracy ACA: 10^{-4}
- $N = 21\ 507$, $L = 8.5\lambda_s$





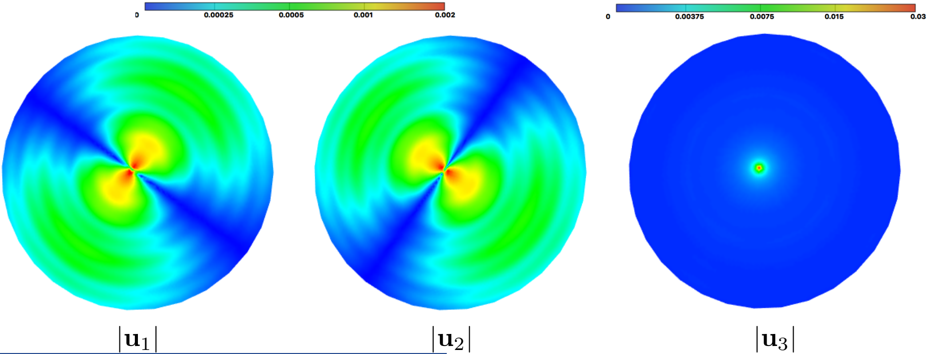
Relative error w.r.t. fundamental solution

Half-space subjected to point load

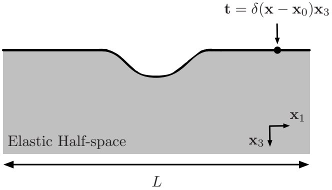


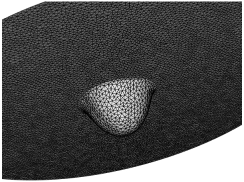
- Density: 10 pts / λ_s
- Accuracy ACA: 10^{-4}
- $N = 21\,507$, $L = 8.5\lambda_s$

Displacement on the free surface



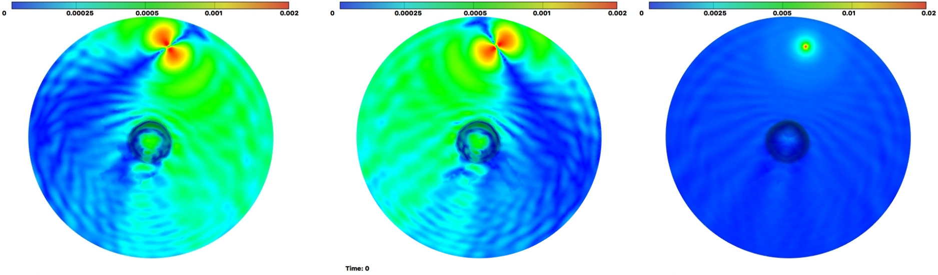
Canyon subjected to a surface Point load





- Density: 10 pts / λ_s
- Accuracy ACA: 10^{-4}
- $L = 10$

Displacement on the free surface for a fixed frequency $L = \lambda_s$



$|\mathbf{u}_1|$ $|\mathbf{u}_2|$ $|\mathbf{u}_3|$

Ongoing works: fast solvers for elastodynamics

Conclusions

- Extension of $\mathcal{H}$-matrix solvers to 3D elastodynamics and parallelization of the computation of the $\mathcal{H}$-matrix
- Accuracy and efficiency checked for 3D elastodynamics
- Error estimate for the direct solver
- First applications to simplified point load problems

Ongoing work

- Extension to piecewise homogeneous domains
- Study of the effect of the low rank approximations on the physics
- Parallelization of the direct solver
- Extension to anisotropic media
- Preconditioning of iterative solvers (low accuracy $\mathcal{H}-$LU factorization based)
- Application to more complex problems

We thank Shell (R.E. Plessix) for funding this exploratory work.

A Fast Direct Solver Based on the Boundary Element Method to Model 3-D Elastic Waves in Large Domains

S. Chaillat-Loseille and P. Ciarlet

POEMS (CNRS-INRIA-ENSTA ParisTech), Université Paris-Saclay, France

OILWS2: Full Waveform Inversion and Velocity Analysis,
Los Angeles, May 4th 2017