Towards a Fast Direct Solver for Maxwell Scattering

Srinath Kailasa [1,2] Manas Rachh [2]

[1] Department of Mathematics, University College London

[2] Centre for Computational Mathematics, Flatiron Institute

September 15, 2022

Table of Contents

Table of Contents

Team

My Research Focus:

1. High Performance & Scientific Computing
2. Fast algorithms for Integral Equations
3. Software Engineering

Find our Software on GitHub:
https://github.com/betckegroup
https://github.com/fastalgorithms

Srinath Kailasa
GitHub: skailasa

Manas Rachh
GitHub: mrachh

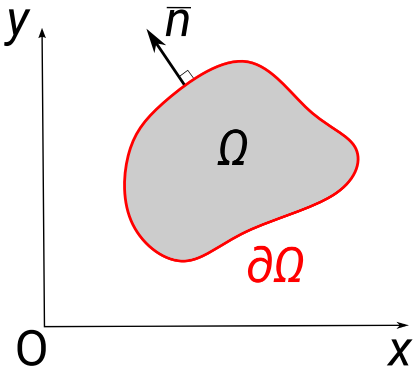
Table of Contents

Motivation I

Consider a simple exterior scattering problem in 2D, we impose a sound hard boundary condition.

$$(\Delta + k^2)u^s = 0$$

$$\frac{\partial u^s}{\partial n} = \frac{\partial u^i}{\partial n}$$

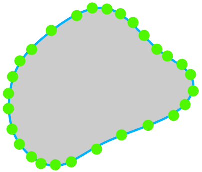


Motivation II

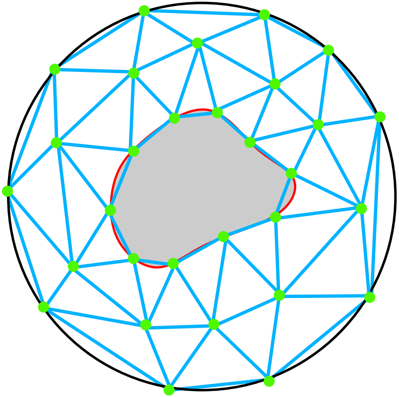
$$(\Delta + k^2)u^s = 0$$

$$\frac{\partial u^s}{\partial n} = \frac{\partial u^i}{\partial n}$$

BEM



FEM



Motivation III

- BEM gives us dense matrices
- Computational cost of naively applying to a vector is $O(N^2)$, matrix-vector product or 'matvec'.
- Computational cost of naively inverting is $O(N^3)$, e.g LU, Gaussian Elimination, QR etc.

Can we take advantage of the properties of our equation to do better than this? Yes! $\rightarrow$ fast algorithms can, in the best case, reduce the application **and** inversion cost to just $O(N)$.

Fast Algorithms I

Fast algorithms enable:

1. Fast particle simulation, e.g. electrostatics and gravitation - the original application!
2. Fast iterative methods for PDEs, e.g. Krylov subspace methods. an $O(N)$ matvec giving a final complexity of $O(N \cdot n_{\text{iter}})$.
3. $O(N)$ Fast direct solvers for matrix inversion, better than iterative methods for problems that involve multiple right hand sides.
4. Time-dependent problems, can solve a fixed geometry at each time step.
5. Can solve a geometry that undergoes low-rank perturbations.

Fast Algorithms II

Late 1980s

- 'Analytic' Fast Multipole Methods - based on analytical multipole series expansions of kernel. $O(N)$ matvec for Laplace, Helmholtz, Stokes (Greengard and Rokhlin, 1987)

1990s/2000s

- Multilevel methods (e.g. $\mathcal{H}$ and $\mathcal{H}^2$ matrices) for fast matvecs and inversion (Hackbusch, 1999), (Hackbusch and Khoromskij 2000).
- 'Semi-Analytic' Kernel Independent Fast Multipole Methods - based on kernel evaluations rather than analytical multipole series expansions of kernel. Easier to write generic software implementations. (Ying et al, 2004)

2010s

- FMM software implementations that can process up to $O(10^6)$ of points per second (Malhotra and Biros, 2015)
- Fast direct solvers and software for 2D and 3D problems (Ambikasaran and Darve, 2014), (Minden et al., 2017)

2020s ...

From Analytic to Algebraic Fast algorithms

Drawbacks of analytic FMM, and analytic fast algorithms. Give sketch of semi-analytic methods, and what might be accomplished by a fully algebraic method.

Algebraic Fast Algorithms for Matrix Inversion

Fast direct solvers, overview of what they are trying to accomplish, and of course the major pros and cons.

Summary

Summarise the motivation for fast algorithms, and briefly discuss their other applications outside of integral equations.

Table of Contents

FMM-LU: A fast direct solver for BIEs

Introduce motivation behind FMM-LU (RS-S).

Proxy Compression I

Proxy Compression II

Laplace

Helmholtz - Sound Hard

Helmholtz - Sound Hard, Numerical Results

Helmholtz - Transmission

Towards Maxwell

Table of Contents

Open Problems & Future Directions

1. Unification of fragmented software landscape for fast-solvers. Offer overview of what's out there and who's working on what, what is parallelized and what's not.
2. FDS for high-frequency problems.
3. Complete a first FDS for maxwell. What kind of problems would this allow us to solve? How close are we to this goal?

Table of Contents

Conclusion

References I