Synthesis of Translation Operators and Execution Plans for the Fast Multipole Method

Isuru Fernando

May 23, 2022

University of Illinois at Urbana-Champaign

Problem

Problem: When an integral equation representation of a system of constant coefficient linear PDEs is given synthesize low complexity translation operators and an optimized execution plan.

Low complexity - $\mathrm{O}(p^d)$ or below for translation operations for a d dimensional order p expansion.

Problem

For example when Stokes equation

$$\mu\nabla^2\mathbf{u} - \boldsymbol{\nabla} p + \mathbf{F}\delta(r) = \mathbf{0} \tag{1}$$

$$\boldsymbol{\nabla}\cdot\mathbf{u} = 0 \tag{2}$$

and the Stokeslet representation with 9 Stokeslets

$$\mathbf{u} = \frac{1}{8\pi\mu}\mathbf{F}.\left(\frac{\mathbb{I}}{\mathbf{r}} + \frac{\mathbf{rr}}{|\mathbf{r}|^3}\right)$$

is given, we want:

- Automatic execution plan that uses minimal number of FMMs. eg:
  - 4 harmonic FMMs for Stokes or
  - 3 biharmonic FMMs for Stokes
- Automatically generate FMM with near-optimal cost, i.e.
  - near-optimal complexity translations
  - low constant factor

Automatic Execution Plan

How to obtain an optimized execution plan?
(eg: Convert 9 Stokeslets $\rightarrow$ 3 biharmonic FMMs)

Outline:

- Find a scalar PDE for each component.
  (eg: Biharmonic PDE for Stokes)
- Write the Green's functions as linear combination of derivatives of a base kernel. (eg: 9 Stokeslets $\rightarrow$ 12 biharmonic FMMs)
- Reduce the number of FMMs needed.
  (eg: 9 Stokeslets $\rightarrow$ 12 biharmonic FMMs $\rightarrow$ 3 biharmonic FMMs)

Low Complexity Translation Operators

Low complexity translation operators achieved by compressed Taylor series or spherical harmonic series.

Need a scalar PDE for each component in the system of PDEs.

In Stokes equation we have,

$$\begin{aligned}
(1) \implies \boldsymbol{\nabla}.(\mu\nabla^2\mathbf{u} - \boldsymbol{\nabla} p) &= 0, \\
\mu\nabla^2\boldsymbol{\nabla}.\mathbf{u} - \nabla^2 p &= 0, \\
(2) \implies \nabla^2 p &= 0, && (3) \\
(1) \implies \nabla^2(\mu\nabla^2\mathbf{u} - \boldsymbol{\nabla} p) &= \mathbf{0}, && (4) \\
(3) \implies \nabla^4\mathbf{u} &= \mathbf{0}. && (5)
\end{aligned}$$

Can we go from system of PDEs to a scalar PDE for each component automatically?

Finding a Scalar PDE

Write the system of PDEs as a linear system with derivative operators.

$$\begin{bmatrix} \mu\nabla^2 & 0 & 0 & -\frac{\partial}{\partial x_1} \\ 0 & \mu\nabla^2 & 0 & -\frac{\partial}{\partial x_2} \\ 0 & 0 & \mu\nabla^2 & -\frac{\partial}{\partial x_3} \\ \frac{\partial}{\partial x_1} & \frac{\partial}{\partial x_2} & \frac{\partial}{\partial x_3} & 0 \end{bmatrix} \begin{bmatrix} u_0 \\ u_1 \\ u_2 \\ p \end{bmatrix} = \mathbf{0}$$

$$\begin{bmatrix} \mathbf{w}_0 & \mathbf{w}_1 & \mathbf{w}_2 & \mathbf{w}_3 \end{bmatrix} \begin{bmatrix} u_0 & u_1 & u_2 & p \end{bmatrix}^T = \mathbf{0}$$

To find a scalar PDE for u_0, find a vector $\mathbf{v}_0$ such that

$$\mathbf{v}_0^T \begin{bmatrix} \mathbf{w}_1 & \mathbf{w}_2 & \mathbf{w}_3 \end{bmatrix} = \mathbf{0}$$

$\mathbf{v}_0^T \mathbf{w}_0 u_0 = 0$ implies that the derivative operator $\mathbf{v}_0^T \mathbf{w}_0$ is a scalar PDE for u_0.

$\mathbf{v}_i, \mathbf{w}_i \in \mathbb{R}[\partial_{x_1}, \partial_{x_2}, \partial_{x_3}]^4$.

Some bits of abstract algebra

An **ideal** I of a ring R is,

- an additive subgroup of R
- multiplying an element of I by an element in R is in I.

eg: All polynomials generated by a finite number of polynomials.

A **Gröbner basis** G of an ideal I of a polynomial ring R is a

- a subset of I
- the ideal generated by the leading monomial of all elements of G is equal to the ideal generated by the leading monomial of all elements of I.

A **syzygy** of a generating set $x_1, \ldots, x_k$ of an ideal I in ring R is a

- a sequence of elements $a_1, \ldots, a_k$
- $a_1 x_1 + \cdots + a_k x_k = 0$.

A **minimal generating set** of an ideal is a generating set such that a subset does not generate the ideal.

Finding a Scalar PDE

Since we have $\mathbf{v}_0^T \mathbf{w}_3 = 0$, in algebra terms $\mathbf{v}_0$ is a syzygy of $\mathbf{w}_3$ or in other terms $\mathbf{v}_0$ is a linear combination of the syzygy module of $\mathbf{w}_3$.

- Find $\mathbf{v}_0$ by calculating the syzygy modules of $\mathbf{w}_1, \mathbf{w}_2, \mathbf{w}_3$ and intersecting the modules.
- Calculate a minimal generating set of the syzygy module and order the set into a matrix M_0 and we have

$$\mathbf{v}_0 = \begin{bmatrix} \partial_{x_2} & -\partial_{x_1} & 0 & 0 \\ \partial_{x_3} & 0 & -\partial_{x_1} & 0 \\ 0 & \partial_{x_3} & -\partial_{x_2} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{q} = M_0 \mathbf{q}$$

  for $\mathbf{q} \in \mathbb{R}[\partial_{x_1}, \partial_{x_2}, \partial_{x_3}]^4$.
- $v_0 = \min_{\deg} (\mathit{groebner}(M_0 \mathbf{w}_0))$ where *groebner* is a Groebner basis of the ideal generated by the vector $M_0 \mathbf{w}_0$ and $\min_{\deg}$ is the minimum degree polynomial.

Rewriting kernels using a base kernel

Now that we know the scalar PDE, we need to re-write the kernel to be a linear combination of derivatives of a base kernel so that we can use existing FMM machinery like spherical harmonics or Taylor series expansions.

In the case of the Mindlin solution we have,

$$u_1^B = \frac{\sigma_1}{x_1 + x_3} + \frac{x_1 \sigma_3}{r(r + x_3)} - \frac{x_1(\sigma_1 x_1 + \sigma_2 x_2)}{r(r + x_3)^2}$$

where $r = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + (x_3 - y_3)^2}$ and Gimbutas et al proved that

$$u_1^B = B_{x_1 x_1} \sigma_1 + B_{x_1 x_2} \sigma_2 + B_{x_1 x_3} \sigma_3$$

where

$$B = (x_3 - y_3) \log(r + x_3 - y_3) - r.$$

Can we do this automatically?

Rewriting kernels using a base kernel

Goal: Let G_1, G_2 be two kernels and we want to find a base kernel G_0 defined by,

$$G_1 = LG_0$$
$$G_2 = MG_0$$

where L, M are derivative operators. Solve

$$MG_1 - LG_2 = 0$$
$$\begin{bmatrix} M & -L \end{bmatrix} \begin{bmatrix} G_1 \\ G_2 \end{bmatrix} = 0$$

for L, M.

Rewriting kernels using a base kernel

Instead of solving

$$\begin{bmatrix} M & -L \end{bmatrix} \begin{bmatrix} G_1 & G_2 \end{bmatrix}^T = 0,$$

we can solve

$$\begin{bmatrix} M_1 & M_2 & M_3 & \dots & -L_1 & -L_2 & -L_3 & \dots \end{bmatrix}$$
$$\begin{bmatrix} G_1 & \frac{\partial G_1}{\partial x_1} & \frac{\partial G_1}{\partial x_2} & \dots & \frac{\partial^p G_1}{\partial x_3^p} & G_2 & \frac{\partial G_2}{\partial x_1} & \frac{\partial G_2}{\partial x_2} & \dots & \frac{\partial^p G_2}{\partial x_3^p} \end{bmatrix}^T = 0$$

$M_i, L_i \in \mathbb{C}$s are the coefficients of the derivative operator.

- Start with $p = 1$
- Sample values of G_1, G_2 and their derivatives upto order p
- If matrix of derivatives is not rank deficient, increase p and start over
- Solve the equation

Rewriting kernels using a base kernel

Recall

$$G_1 = LG_0$$
$$G_2 = MG_0$$

We now know G_1, G_2, L, M but not G_0.

For Taylor series based expansions

- No need of G_0.
- Only need G_1, G_2 and all of their derivatives

for expansions, translations and evaluations.

Still too expensive. eg: Stokes 9 Stokeslets $\rightarrow$ 12 biharmonic FMMs.

$$\frac{x_2 x_3}{r} = -\partial_{x_2 x_3} r$$
$$\frac{x_1 x_1}{r} = \left(\partial_{x_2 x_2} + \partial_{x_3 x_3}\right) r$$

Reducing the Number of FMMs

Let us look at the following simpler example.

Assume we want to calculate the following outputs

$$S\sigma + D\mu$$
$$S_{x_1}\sigma + D_{x_1}\mu$$

where S, D are single layer and double layer potentials and S_{x_1}, D_{x_1} are target derivatives of those.

Naively requires 4 FMMs, but only 1 FMM is enough.

Reducing the Number of FMMs

More complicated example from Mindlin solution for half-space elasticity.

$$\begin{aligned} u_1^B &= B_{x_1 x_1}\sigma_1 + B_{x_1 x_2}\sigma_2 + B_{x_1 x_3}\sigma_3 \\ u_2^B &= B_{x_2 x_1}\sigma_1 + B_{x_2 x_2}\sigma_2 + B_{x_2 x_3}\sigma_3 \\ u_3^B &= B_{x_3 x_1}\sigma_1 + B_{x_3 x_2}\sigma_2 + B_{x_3 x_3}\sigma_3 \end{aligned}$$

Gimbutas et al 2016 showed that this can be rewritten as

$$\begin{aligned} u_1^B &= -(B_{y_1}\sigma_1 + B_{y_2}\sigma_2 + B_{y_3}\sigma_3)_{x_1} \\ u_2^B &= -(B_{y_1}\sigma_1 + B_{y_2}\sigma_2 + B_{y_3}\sigma_3)_{x_2} \\ u_3^B &= -(B_{y_1}\sigma_1 + B_{y_2}\sigma_2 + B_{y_3}\sigma_3)_{x_3} \end{aligned}$$

Reducing the Number of FMMs

Rewrite the derivative operators as polynomials and factorize. For example, rewriting the example from Mindlin solution before, we have

$$\begin{bmatrix} \partial_{x_1^2} & \partial_{x_1 x_2} & \partial_{x_1 x_3} \\ \partial_{x_2 x_1} & \partial_{x_2^2} & \partial_{x_2 x_3} \\ \partial_{x_3 x_1} & \partial_{x_3 x_2} & \partial_{x_3^2} \end{bmatrix} \begin{bmatrix} B\sigma_1 \\ B\sigma_2 \\ B\sigma_3 \end{bmatrix} = \begin{bmatrix} \partial_{x_1} \\ \partial_{x_2} \\ \partial_{x_3} \end{bmatrix} \begin{bmatrix} \partial_{x_1} & \partial_{x_2} & \partial_{x_3} \end{bmatrix} \begin{bmatrix} B\sigma_1 \\ B\sigma_2 \\ B\sigma_3 \end{bmatrix}$$

$$= \underbrace{\begin{bmatrix} \partial_{x_1} \\ \partial_{x_2} \\ \partial_{x_3} \end{bmatrix}}_{\text{evaluate at target}} \underbrace{\begin{bmatrix} -\partial_{y_1} & -\partial_{y_2} & -\partial_{y_3} \end{bmatrix} \begin{bmatrix} B\sigma_1 \\ B\sigma_2 \\ B\sigma_3 \end{bmatrix}}_{\text{evaluate at source}}$$

How to do a "rank" revealing factorization of a matrix with polynomials as elements?

Assumptions

- Only target and source derivative transformations
- Only one kernel and it is translation invariant
- No composition of layer potentials involved

Reducing the Number of FMMs

Can we obtain a "rank" revealing factorization of a matrix $M = LR$ with

- elements of L, R in the ring $\mathbb{R}[\partial_{x_1}, \partial_{x_2}, \partial_{x_3}]$
- the number of rows of R is minimal?

Non-optimal method:

$$\mathrm{syz}(M)M = 0$$
$$\implies M^T \mathrm{syz}(M)^T = 0$$
$$\implies M^T \in \mathrm{syz}(\mathrm{syz}(M)^T)$$

Here $\mathrm{syz}(M)$ is a minimal generating set of the syzygy module of M.
Then $R = \mathrm{syz}(\mathrm{syz}(M)^T)$.

- Does not guarantee that the number of rows of R is minimal.
- Found R matching the solution by Gimbutas et al for the Mindlin solution.

Recap on Systems of PDEs

- Convert the system to a scalar PDE for each component
- Rewrite the kernels as a linear combination of derivatives of a base kernel
- Reduce the number of FMMs

Examples

Integral equation	Automated algorithm	Literature
Stokeslet and stresslet	3 biharmonic	4 harmonic
Elasticity - Kelvin	3 biharmonic	4 harmonic
Elasticity - Mindlin	3 biharmonic + 1 harmonic	8 harmonic

A biharmonic FMM costs roughly 50% more than a harmonic FMM.

Expansions for Fast Multipole Method

New goal: given a scalar PDE synthesize an FMM such that

- Expansion can be done without knowing the "base" kernel explicitly. i.e. Only need the derivative operators L, M in

$$G_1 = LG_0$$
$$G_2 = MG_0.$$

- Low complexity translation operators for any kernel given symbolically.

Taylor Series based FMM

Let ψ be a translation invariant potential satisfying a constant coefficient linear PDE. Let $\mathbf{s} \in \mathbb{R}^d$ a source point and $\mathbf{t} \in \mathbb{R}^d$ be a target point.

Local expansion:

$$\psi(\mathbf{t}, \mathbf{s}) = \sum_{|m| \le k} \underbrace{\frac{D_{\mathbf{t}}^m \psi(\mathbf{t}, \mathbf{s})\Big|_{\mathbf{t}=\mathbf{c}}}{m!}}_{\text{depends on src/ctr}} \underbrace{(\mathbf{t} - \mathbf{c})^m}_{\text{depends on tgt/ctr}} \qquad \text{converges when } |t - c| < R.$$

Multipole expansion:

$$\psi(\mathbf{t}, \mathbf{s}) = \sum_{|m| \le k} \underbrace{\frac{D_{\mathbf{s}}^m \psi(\mathbf{t}, \mathbf{s})\Big|_{\mathbf{s}=\mathbf{c}}}{m!}}_{\text{depends on tgt/ctr}} \underbrace{(\mathbf{s} - \mathbf{c})^m}_{\text{depends on src/ctr}} \qquad \text{converges when } |t - c| > R.$$

Too costly: $\mathcal{O}(p^{2d})$ translations and $\mathcal{O}(p^d)$ storage.

Compressed Multipole Expansion

Recall the multipole expansion

$$\psi(\mathbf{t}, \mathbf{s}) = \sum_{|m| \le p} \underbrace{\frac{D_{\mathbf{s}}^m \psi(\mathbf{t}, \mathbf{s})\Big|_{\mathbf{s}=\mathbf{c}}}{m!}}_{\text{depends on tgt/ctr}} \underbrace{(\mathbf{s} - \mathbf{c})^m}_{\text{depends on src/ctr}} .$$

Let $G, G_x, G_y, \ldots$ be the derivatives of ψ w.r.t $\mathbf{s}$ evaluated at center $\mathbf{c}$.

When the potential ψ satisfies the Helmholtz equation we have

$$G_{xx} + G_{yy} + \kappa^2 G = 0.$$

From the PDE we have

$$\begin{aligned}
\psi &= c_1 G + c_2 G_{xx} + c_3 G_{yy} + \cdots \\
&= c_1 G + c_2 G_{xx} + c_3(-G_{xx} - \kappa^2 G) + \cdots \\
&= (c_1 - \kappa^2 c_3) G + (c_2 - c_3) G_{xx} + 0 G_{yy} + \cdots .
\end{aligned}$$

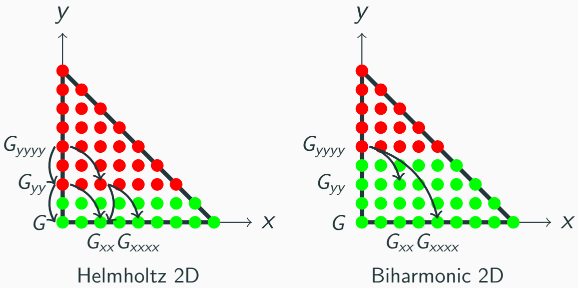
We call this a compressed Taylor series.

Compressed Multipole Expansion

For Helmholtz equation we also have

$$G_{xxyy} + G_{yyyy} + \kappa^2 G_{yy} = 0,$$

$$G_{xxxx} + G_{xxyy} + \kappa^2 G_{xx} = 0.$$



All the coefficients represented by red dots get lumped into "green" coefficients.

Count of expansion coefficients goes from $\mathcal{O}(p^d)$ to $\mathcal{O}(p^{d-1})$.

Optimizations for Faster Translations

- Use recurrence relations for the Green's functions to get amortized $\mathcal{O}(1)$ time for a derivative when calculating all derivatives.
- Use a Fast Fourier Transform to calculate M2L which is a recursive block Toeplitz matrix.
  - $\mathcal{O}(p^{d-1}\log(p))$ for elliptic PDEs
  - $\mathcal{O}(p^{d}\log(p))$ for other PDEs
- Use temporaries to reduce the time of L2L and M2M from $\mathcal{O}(p^{2d-1})$ to $\mathcal{O}(p^{d})$.

Time Complexities

	P2L/M2P	P2M/L2P	M2M/L2L	M2L
Taylor Series	$\mathbf{p}^3$	$\mathbf{p}^3$	$\mathbf{p}^6$	$\mathbf{p}^6$
Improved Taylor Series	$\mathbf{p}^3$	$\mathbf{p}^3$	$\mathbf{p}^4$	$\mathbf{p}^3 \log(\mathbf{p})$
Compressed Taylor Series	$\mathbf{p}^2$	$\mathbf{p}^3$	$\mathbf{p}^3$	$\mathbf{p}^2 \log(\mathbf{p})$
Spherical Harmonic Series	$\mathbf{p}^2$	$\mathbf{p}^2$	$\mathbf{p}^2 \log(\mathbf{p})$	$\mathbf{p}^2 \log(\mathbf{p})$

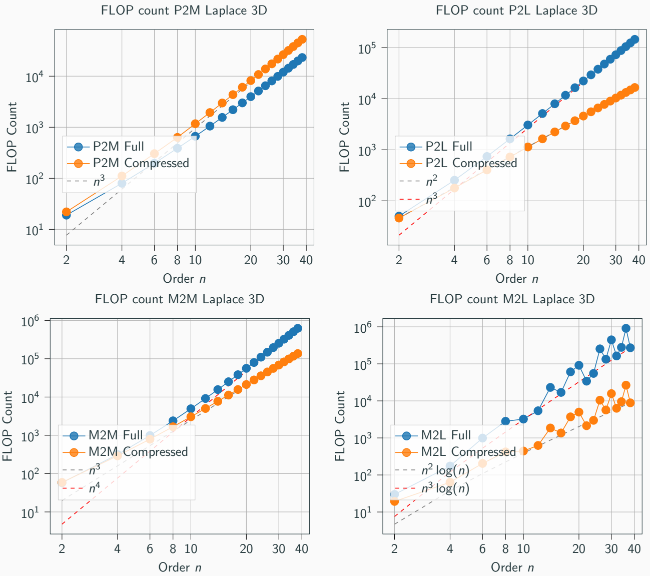
Table 1: Time complexities for expansions, translations and evaluations

All operations are exact except for M2M in Compressed Taylor and M2L operations with FFT.

Works for all constant coefficient elliptic PDEs with translation invariant kernels.

Needs only the PDE, Green's function and optionally a fast way to calculate derivatives of the Green's function.

Time Complexities



Taylor Series Expansions for System of PDEs

- Can generate low complexity translation operators for any kernel.
- No need to know the base kernel. Only need the derivative operators L, M in

$$
\begin{aligned}
G_1 &= LG_0 \\
G_2 &= MG_0.
\end{aligned}
$$

- PDE is only used for compression and does not change the FMM. For example, even though Stokeslet FMM is converted to a biharmonic FMM, the biharmonic relation is only used for compression.
- Relative error with uncompressed Taylor series is either zero or has the same order as the truncation error.

Summary

- Kernel-generic method for elliptic constant coefficient linear system of PDEs with non-oscillatory kernels.
- Automatically reduces the number of FMM calls needed.
- Only needs the PDE and the Green's functions for the PDE.

Ackowledgements:

- Andreas Klöckner, Xiaoyu Wei
- NSF grants 19-11019 and 16-54756