Dissertation Defense

Partial FFT Direct Parallel Algorithms for Subsurface Scattering Problems

Ron Gonzales Ph.D. Candidate

Idaho State University
Department of Mathematics and Statistics

April 21, 2022

Introduction

- Novel direct parallel partial FFT-type algorithm for the numerical solutions of the two- and three-dimensional Helmholtz equations.
- Governing equations are discretized by high-order compact finite-difference schemes.
- Accuracy and scalability of the direct parallel method are investigated on scattering problems with realistic ranges of parameters for air, soil and mine-like targets.

Outline

- Model Problem
- Research Chronology
- Discretization
- Direct Parallel FFT Solver
- Direct Parallel Eigenvalue Solver
- Direct Parallel Partial FFT Solver
- Numerical Results

Model Problem

The two- and three-dimensional Helmholtz equations on a rectangular domains. That is

$$\Delta u(\mathbf{x}) + k^2(\mathbf{x})u(\mathbf{x}) = f(\mathbf{x}) \ \text{ in } \Omega.$$

The domain is defined as $\Omega = \left\{\mathbf{x} = (x, y, z) \in \mathbb{R}^3 \mid x_l \leq x \leq x_u,\ y_l \leq y \leq y_u,\ z_l \leq z \leq z_u\right\}$ where $x_l < x_u$, $y_l < y_u$ and $z_l < z_u$. The function k is complex valued. Dirichlet and Sommerfeld-like boundary conditions are considered.

Research Chronology

Construction of Higher-Order Compact Schemes:

Lele (1992),

Nabadi, Siddiqui and Dargahi (2007),

Sutmann (2007),

Turkel, D. Gordon, R. Gordon and Tsynkov (2013).

Second-Order FD and FEM + GMRES + FFT

Elman and O'Leary (1998),

Gryazin, Klibanov and Lucas (2000).

Research Chronology

Parallel Compact Higher-Order Schemes+GMRES+FFT:

Gryazin, Lee and Gonzales (2019),

Gonzales, Gryazin and Lee (2021).

Partial FFT Solver for Compact Schemes:

Toivanen and Wolfmayr (2020).

Parallel Partial FFT Solver for High-Order Compact Schemes:

Gonzales and Gryazin (2022).

Second-Order Compact Scheme

Consider the following notation for the first and second central differences at the (i,j,l)-th grid point

$$\delta_x u_{i,j,l} = \frac{u_{i+1,j,l} - u_{i-1,j,l}}{2h_x}, \qquad \delta_x^2 u_{i,j,l} = \frac{u_{i-1,j,l} - 2u_{i,j,l} + u_{i+1,j,l}}{h_x^2}$$

where $u_{i,j,l} = u(x_i, y_j, z_l)$. Then the standard second-order scheme is given by

$$\frac{1}{h_x^2}u_{i-1,j,l} + \frac{1}{h_y^2}u_{i,j-1,l} + \frac{1}{h_z^2}u_{i,j,l-1} + \left(k_{i,j,l}^2 - \frac{2}{h_x^2} - \frac{2}{h_y^2} - \frac{2}{h_z^2}\right)u_{i,j,l}$$

$$+ \frac{1}{h_x^2}u_{i+1,j,l} + \frac{1}{h_y^2}u_{i,j+1,l} + \frac{1}{h_z^2}u_{i,j,l+1} = f_{i,j,l}.$$

Fourth-Order Compact Scheme

$$\left(1 + h_x^2\frac{\delta_x^2}{12} + h_y^2\frac{\delta_y^2}{12} + h_z^2\frac{\delta_z^2}{12}\right)k_{i,j,l}^2 u_{i,j,l}$$

$$+\left(\delta_x^2 + h_y^2\frac{\delta_x^2\delta_y^2}{12} + h_z^2\frac{\delta_x^2\delta_z^2}{12}\right)u_{i,j,l} + \left(\delta_y^2 + h_x^2\frac{\delta_x^2\delta_y^2}{12} + h_z^2\frac{\delta_y^2\delta_z^2}{12}\right)u_{i,j,l}$$

$$+\left(\delta_z^2 + h_x^2\frac{\delta_x^2\delta_z^2}{12} + h_y^2\frac{\delta_y^2\delta_z^2}{12}\right)u_{i,j,l} = \left(1 + h_x^2\frac{\delta_x^2}{12} + h_y^2\frac{\delta_y^2}{12} + h_z^2\frac{\delta_z^2}{12}\right)f_{i,j,l}$$

Lele (1992)

Sixth-Order Notation

$$\Delta_h u_{i,j,l} = \left(\delta_x^2 + \delta_y^2 + \delta_z^2\right) u_{i,j,l}$$

$$L_h u_{i,j,l} = (\Delta_h + k_{i,j,l}^2) u_{i,j,l}$$

$$\nabla_h u_{i,j,l} = (\delta_x \, , \, \delta_y \, , \, \delta_z)\, u_{i,j,l}$$

$$\nabla_h^{1/2} u_{i,j,l} = \left(\delta_x \delta_y^2 \, , \, \delta_x^2 \delta_y \, , \, \delta_x^2 \delta_z\right) u_{i,j,l} + \left(\delta_x \delta_z^2 \, , \, \delta_y \delta_z^2 \, , \, \delta_y^2 \delta_z\right) u_{i,j,l}$$

$$\nabla^4 u = \left(\frac{\partial^4}{\partial x^4} + \frac{\partial^4}{\partial y^4} + \frac{\partial^4}{\partial z^4}\right) u$$

Sixth-Order Compact Scheme

$$L_h u_{i,j,l} + \frac{h^2}{6}\left(\delta_x^2\delta_y^2 + \delta_x^2\delta_z^2 + \delta_y^2\delta_z^2\right)u_{i,j,l} + \frac{h^2}{20}\left(\Delta k_{i,j,l}^2 - k_{i,j,l}^4\right)u_{i,j,l}$$

$$+ \frac{h^2}{10}\nabla k_{i,j,l}^2 \cdot \nabla_h u_{i,j,l} + \frac{h^4}{60}\nabla k_{i,j,l}^2 \cdot \left(\nabla_h^{1/2} u_{i,j,l} + \nabla_h \left(k^2 u\right)_{i,j,l}\right)$$

$$+ \frac{h^4}{30}\delta_x^2\delta_y^2\delta_z^2 u_{i,j,l} + \frac{h^2}{30}\Delta_h\left(k^2 u\right)_{i,j,l} + \frac{h^4}{90}\left(\delta_x^2\delta_y^2 + \delta_x^2\delta_z^2 + \delta_y^2\delta_z^2\right)\left(k^2 u\right)_{i,j,l}$$

$$= \left(1 - \frac{h^2}{20}k_{i,j,l}^2\right)f_{i,j,l} + \frac{h^2}{12}\Delta f_{i,j,l} + \frac{h^4}{60}\nabla k_{i,j,l}^2 \cdot \nabla f_{i,j,l} + \frac{h^4}{360}\nabla^4 f_{i,j,l}$$

$$+ \frac{h^4}{90}\left(\frac{\partial^4}{\partial x^2 \partial y^2} + \frac{\partial^4}{\partial x^2 \partial z^2} + \frac{\partial^4}{\partial y^2 \partial z^2}\right)f_{i,j,l}$$

Turkel, D. Gordon, R. Gordon, and Tsynkov (2013)

Sommerfeld-Like Boundary Conditions

The Sommerfeld radiation conditions are

$$\lim_{r\to\infty} r^{(d-1)/2}\left(\frac{\partial}{\partial r}u(\mathbf{x}) - ik(\mathbf{x})u(\mathbf{x})\right) = 0$$

with $d = 2$ or $d = 3$ for two- and three-dimensions respectively. Truncating the unbounded domain to a finite domain at the boundary under consideration provides the approximation. That is

$$\nabla u(\mathbf{x})\cdot\mathbf{n} - ik(\mathbf{x})u(\mathbf{x}) = 0$$

for $\mathbf{x}\in\partial\Omega$ where $\mathbf{n}$ is the outward normal vector of the boundary.

Sommerfeld-Like Boundary Conditions Approximation

The ninth-order approximation for the Sommerfeld-like boundary conditions on the x boundaries are

$$u_{\iota\pm1,j,l} = u_{\iota\mp1,j,l} + 2ih_x k_{j,l}\left(1 - \frac{h_x^2 k_{j,l}^2}{3!} + \frac{h_x^4 k_{j,l}^4}{5!} - \frac{h_x^6 k_{j,l}^6}{7!}\right) u_{\iota,j,l}$$

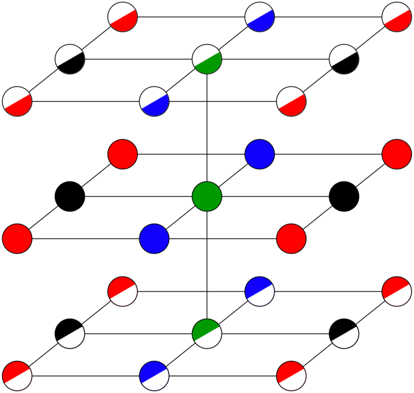
$$= u_{\iota\mp1,j,l} + \alpha_{x,j,l} u_{\iota,j,l}$$

where $\alpha_{x,j,l} = 2ih_x k_{j,l}\left(1 - h_x^2 k_{j,l}^2/3! + h_x^4 k_{j,l}^4/5! - h_x^6 k_{j,l}^6/7!\right)$. The coefficients $\alpha_{y,\iota,l}$ and $\alpha_{z,\iota,j}$ are found in the same way.

Parallel FFT Solver for Compact Schemes

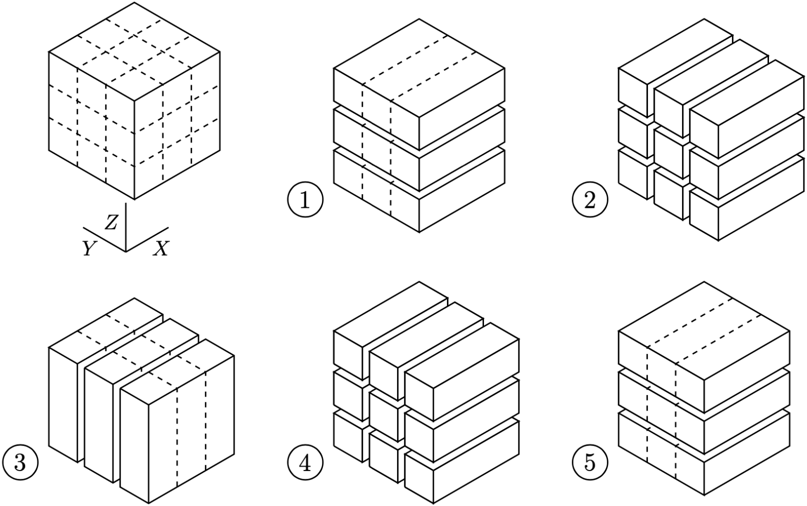
- Assumptions for the parallel FFT solver
  - Sommerfeld-like conditions on the top and bottom boundaries with respect to the vertical direction
  - Dirichlet conditions on the other boundaries
  - The coefficient k varies only vertically, i.e. $k(\mathbf{x}) = k(z)$
- These conditions produce a stencil pattern that ensures the eigenvalues and eigenvectors of the linear system $A\mathbf{u} = \mathbf{f}$ can be represented in terms of real valued functions, i.e. sines and cosines.
- Allows the use of FFT for the diagonalization of the system.
- Solver steps
  - Forward transform (diagonalization)
  - Solution
  - Reverse transform

27 Point Stencil Symmetry for Parallel FFT Solver



Gonzales, Gryazin and Lee (2021)

Parallelization



Parallel FFT Solver

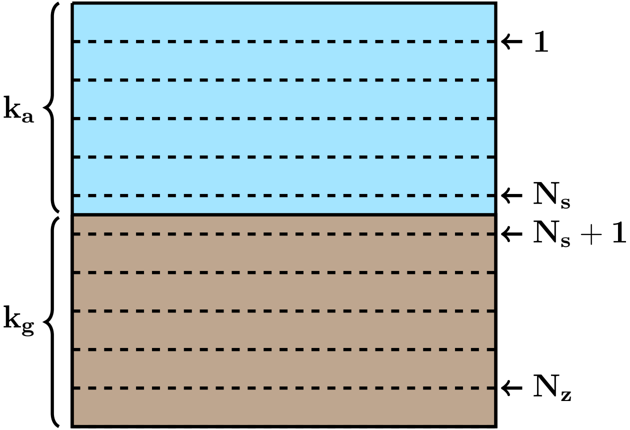
Algorithm Parallel FFT Solver

Find array indexes that evenly split work, i.e. $start_y$, end_y, $start_z$, and end_z
for $l = start_z \ldots end_z$ **do**
    2D forward DST in $x-$ and $y-$ directions via *FFT*
end for
Transfer data via MPI if in a distributed memory environment
for $j = start_y \ldots end_y$; $i = 1 \ldots N_x$ **do**
    Solve the tridiagonal system using LU decomposition
end for
Transfer data via MPI if in a distributed memory environment
for $l = start_z \ldots end_z$ **do**
    2D inverse DST in $x-$ and $y-$ directions via *FFT*
end for

Parallel FFT Solver Versatility

- The parallel FFT solver can solve different PDEs, e.g. the Convection-Diffusion equation.
- This solver is limited to Dirichlet, Neumann and periodic boundary conditions.
- A more generalized solver for the case of absorbing boundary conditions, i.e. Sommerfeld-like, is required.

Domain with Air and Soil Interface



Parallel Eigenvalue Solver Notation

- Sommerfeld-like conditions on all boundaries.
- Define N_s as the vertical index that is the last to contain the constant air coefficient k_a.
- Matrices A_a, B_a, A_g, and $B_g \in \mathbb{C}^{N_x \cdot N_y \times N_x \cdot N_y}$ are generated from the FFT solver 27-point stencil.
- Consider the generalized eigenvalues $A_a S = B_a S \Lambda_a$ and $A_g R = B_g R \Lambda_g$.
- $\mathbf{w}_l = S^{-1}\mathbf{u}_l$ and $\overline{\mathbf{f}}_l = S^{-1} B_a^{-1} \mathbf{f}_l \qquad$ for $l = 1, \ldots, N_s$
- $\mathbf{w}_l = R^{-1}\mathbf{u}_l$ and $\overline{\mathbf{f}}_l = R^{-1} B_g^{-1} \mathbf{f}_l \qquad$ for $l = N_s + 1, \ldots, N_z$

The partially diagonalized system is given by

$$
\begin{aligned}
(\Lambda_a + \alpha_a I)\,\mathbf{w}_1 + (1 + \beta_a)\,\mathbf{w}_2 &= \widehat{\mathbf{f}}_1 \\
\mathbf{w}_{l-1} + \Lambda_a \mathbf{w}_l + \mathbf{w}_{l+1} &= \widehat{\mathbf{f}}_l \qquad \text{for } l = 2, \ldots, N_s - 1 \\
\mathbf{w}_{N_s-1} + \Lambda_a \mathbf{w}_{N_s} + M\mathbf{w}_{N_s+1} &= \widehat{\mathbf{f}}_{N_s} \\
M^{-1}\mathbf{w}_{N_s} + \Lambda_g \mathbf{w}_{N_s+1} + \mathbf{w}_{N_s+2} &= \widehat{\mathbf{f}}_{N_s+1} \\
\mathbf{w}_{l-1} + \Lambda_g \mathbf{w}_l + \mathbf{w}_{l+1} &= \widehat{\mathbf{f}}_l \qquad \text{for } l = N_s + 2, \ldots, N_z - 1 \\
(1 + \beta_g)\,\mathbf{w}_{N_z-1} + (\Lambda_g + \alpha_g I)\,\mathbf{w}_{N_z} &= \widehat{\mathbf{f}}_{N_z}
\end{aligned}
$$

where $M = S^{-1}B_a^{-1}B_gR$ and α_a, α_g, β_a and β_g are coefficients from the approximation of the Sommerfeld-like boundary conditions.

To solve this transformed tridiagonal system assume that

$$\mathbf{w}_l = D_{a,l}\mathbf{w}_{l+1} + P_{a,l} \qquad \text{for } l = 1, \ldots, N_s - 1$$

It follows that

$$D_{a,1} = -2\left(\Lambda_a + \alpha_a I\right)^{-1} \text{ and } P_{a,1} = \left(\Lambda_a + \alpha_a I\right)^{-1}\overline{\mathbf{f}}_1$$

$$D_{a,l} = -\left(\Lambda_a \ + D_{a,l-1}\right)^{-1} \text{ and } P_{a,l} = \left(\Lambda_a \ + D_{a,l-1}\right)^{-1}\left(\overline{\mathbf{f}}_l - P_{a,l-1}\right)$$

for $l = 1, \ldots, N_s - 1$.

In the opposite direction, consider

$$\mathbf{w}_l = D_{g,l}\mathbf{w}_{l-1} + P_{g,l} \qquad \text{for } l = N_z, \ldots, N_s + 2.$$

It follows that

$$D_{g,N_z} = -2\left(\Lambda_g + \alpha_g I\right)^{-1} \text{ and } P_{g,N_z} = \left(\Lambda_g + \alpha_g I\right)^{-1} \overline{\mathbf{f}}_{N_z}$$

$$D_{g,l} = -\left(\Lambda_g + D_{g,l+1}\right)^{-1} \text{ and } P_{g,l} = \left(\Lambda_g + D_{g,l+1}\right)^{-1} \left(\overline{\mathbf{f}}_l - P_{g,l+1}\right)$$

for $l = N_z, \ldots, N_s + 2$.

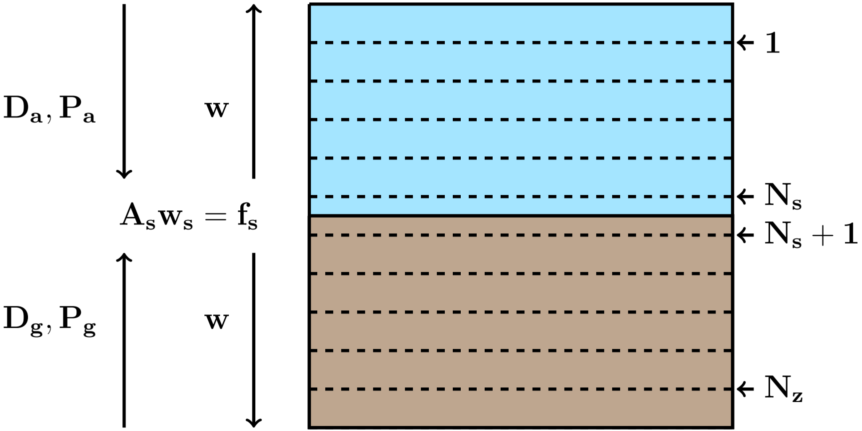
The layers remaining are

$$\mathbf{w}_{N_s-1} + \Lambda_a \mathbf{w}_{N_s} + M\mathbf{w}_{N_s+1} = \widehat{\mathbf{f}}_{N_s}$$
$$M\mathbf{w}_{N_s} + \Lambda_g \mathbf{w}_{N_s+1} + \mathbf{w}_{N_s+2} = \widehat{\mathbf{f}}_{N_s+1}.$$

These layers form the two by two block system $A_s\mathbf{w}_s = \mathbf{f}_s$ solved by LU factorization. The remaining transformed solution is computed by

$$\mathbf{w}_l = D_{a,l}\mathbf{w}_{l+1} + P_{a,l} \qquad \text{for } l = N_s - 1, \ldots, 1$$
$$\mathbf{w}_l = D_{g,l}\mathbf{w}_{l-1} + P_{g,l} \qquad \text{for } l = N_s + 2, \ldots, N_z.$$

Basic Idea



Gonzales and Gryazin (2022)

Parallel Generalized Eigenvalue Solver

Algorithm Parallel Generalized Eigenvalue Solver

1: Preliminarily S, S^{-1}, Λ_a, R, R^{-1}, Λ_g, D_a, D_g, and the LU of A_s.

2: $\widehat{\mathbf{f}}_l = S^{-1}B_a^{-1}\mathbf{f}_l$ **for** $l = 1, \ldots, N_s$

3: $\widehat{\mathbf{f}}_l = R^{-1}B_g^{-1}\mathbf{f}_l$ **for** $l = N_s + 1, \ldots, N_z$

4: Compute P_a and P_g

5: Solve $A_s\mathbf{w}_s = \mathbf{f}_s$ via LU decomposition.

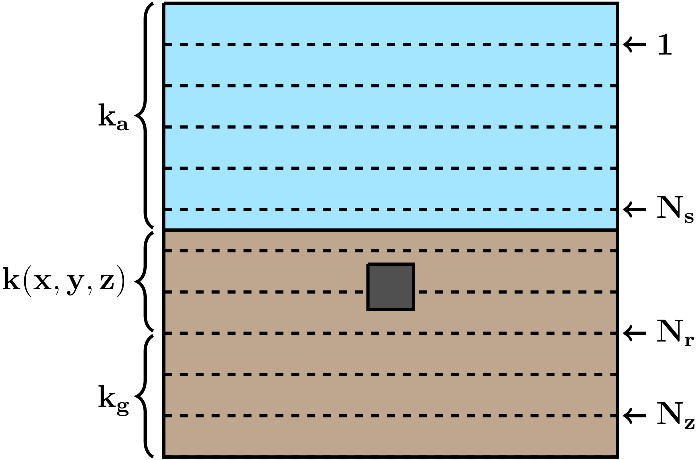
6: $\mathbf{w}_l = D_{a,l}\mathbf{w}_{l+1} + P_{a,l}$ **for** $l = N_s - 1, \ldots, 1$

7: $\mathbf{w}_l = D_{g,l}\mathbf{w}_{l-1} + P_{g,l}$ **for** $l = N_s + 2, \ldots, N_z$

8: $\mathbf{u}_l = S\mathbf{w}_1$ **for** $l = 1, \ldots, N_s$

9: $\mathbf{u}_l = R\mathbf{w}_l$ **for** $l = N_s + 1, \ldots, N_z$

Air and Soil Domain with Subsurface Inclusion



The layers that are not fully diagonalized are

$$
\begin{aligned}
\mathbf{w}_{N_s-1} + \Lambda_a \mathbf{w}_{N_s} + M_a \mathbf{u}_{N_s+1} &= \widehat{\mathbf{f}}_{N_s} \\
C_l \mathbf{u}_{l-1} + A_l \mathbf{u}_l + B_l \mathbf{u}_{l+1} &= \mathbf{f}_l \qquad \text{for } l = N_s+1, \ldots, N_r - 1 \\
M_g \mathbf{u}_{N_r-1} + \Lambda_g \mathbf{w}_{N_r} + \mathbf{w}_{N_r+1} &= \widehat{\mathbf{f}}_{N_r}.
\end{aligned}
$$

where $M_a = S^{-1} B_a^{-1} B_{N_s}$ and $M_g = R^{-1} B_g^{-1} C_{N_r}$. The matrices C_l, A_l and $B_l \in \mathbb{C}^{N_x \cdot N_y \times N_x \cdot N_y}$ are generated from the general 27-point stencil. These layers form the block tridiagonal system $A_s \mathbf{w}_s = \mathbf{f}_s$ solved by LU factorization. The remaining transformed solution is computed by

$$
\begin{aligned}
\mathbf{w}_l &= D_{a,l} \mathbf{w}_{l+1} + P_{a,l} \qquad \text{for } l = N_s - 1, \ldots, 1 \\
\mathbf{w}_l &= D_{g,l} \mathbf{w}_{l-1} + P_{g,l} \qquad \text{for } l = N_r + 1, \ldots, N_z.
\end{aligned}
$$

Parallel Partial FFT Solver

Let A be the matrix produced by the 27-point stencil with Sommerfeld-like conditions on all boundaries. Also, let B be the matrix produced by the 27-point stencil with the pattern for the FFT Solver. The problem at hand is given by

$$A\mathbf{u} = \mathbf{f}.$$

Define $C = B - A$. Consider

$$A\mathbf{y} = A(\mathbf{u} - \mathbf{x}) = \mathbf{f} - A\mathbf{x} = B\mathbf{x} - A\mathbf{x} = C\mathbf{x}$$

where $\mathbf{y} = \mathbf{u} - \mathbf{x}$ and $B\mathbf{x} = \mathbf{f}$. Then

$$A\mathbf{u} = A(\mathbf{y} + \mathbf{x}) = C\mathbf{x} + A\mathbf{x} = B\mathbf{x} = \mathbf{f}.$$

Parallel Partial FFT Solver

The matrix $C = B - A$ is vastly sparse. It is nonzero only at the elements necessary for the approximation of the $x-$ and $y-$ Sommerfeld-like boundary conditions and subsurface inclusion.

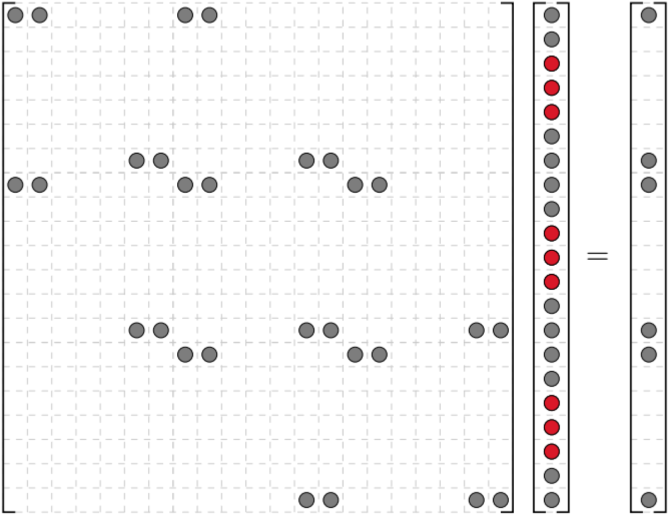
Algorithm Parallel Partial FFT Solver

1: Solve $B\mathbf{x} = \mathbf{f}$ via parallel FFT solver for only the necessary components of $\mathbf{x}$

2: Solve $A\mathbf{y} = C\mathbf{x}$ via parallel eigenvalue solver for only the necessary components of $\mathbf{y}$, here $\mathbf{y} = \mathbf{u} - \mathbf{x}$

3: Solve $B\mathbf{u} = \mathbf{f} + C(\mathbf{x} + \mathbf{y})$ for the entire solution $\mathbf{u}$

2D Example $N_x = 7$, $N_y = 3$



Parallel FFT Solver Test Problem

In the following numerical experiment k is defined by

$$k(z) = a - b\sin(cz)$$

with $a > b \geq 0$. To test the accuracy, consider the analytic solution

$$u(\mathbf{x}) = \sin\left(\beta x\right)\sin\left(\gamma y\right)e^{-k(z)/c}$$

where $\beta^2 + \gamma^2 = a^2 + b^2$. This gives the right-hand side

$$f(\mathbf{x}) = -b(2a + c)\sin(cz)e^{-k(z)/c}\sin(\beta x)\sin(\gamma y).$$

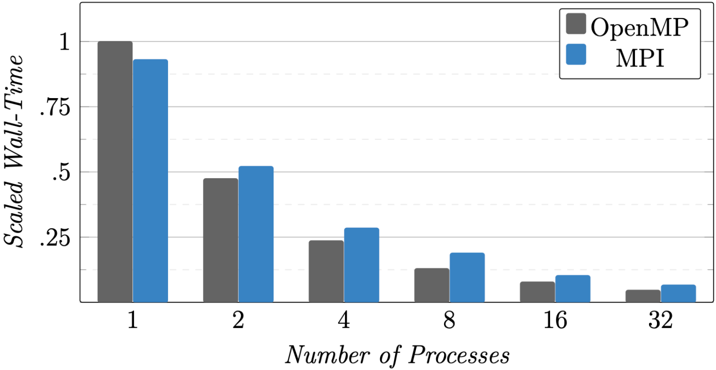
The domain under consideration is $\Omega = [0, \pi] \times [0, \pi] \times [0, \pi]$.

Turkel, D. Gordon, R. Gordon, and Tsynkov (2013)

Parallel FFT Solver Accuracy

	Second-Order		Fourth-Order		Sixth-Order	
Grid	$\|\|\mathbf{u}-\mathbf{u}_a\|\|_\infty$	$\|\|A\mathbf{u}-\mathbf{f}\|\|_2$	$\|\|\mathbf{u}-\mathbf{u}_a\|\|_\infty$	$\|\|A\mathbf{u}-\mathbf{f}\|\|_2$	$\|\|\mathbf{u}-\mathbf{u}_a\|\|_\infty$	$\|\|A\mathbf{u}-\mathbf{f}\|\|_2$
125^3	5.757e-03	4.727e-13	3.449e-05	3.630e-13	2.188e-06	3.158e-13
250^3	1.485e-03	2.585e-12	2.178e-06	1.986e-12	3.494e-08	2.054e-12
500^3	3.745e-04	6.588e-12	1.372e-07	5.083e-12	5.521e-10	5.215e-12

Parallel FFT Solver OpenMP vs MPI on 512^3



Parallel FFT Solver Hybrid on 512^3

n\t	1	2	4	8	16	32
1	39.61	21.73	12.93	8.83	6.84	5.62
2	19.87	10.94	6.48	4.56	3.48	2.92
4	9.99	5.66	3.48	2.52	2.10	1.82
8	5.27	3.11	1.99	1.55	1.37	1.27
16	2.49	1.42	0.85	0.64	0.58	0.58
32	1.85	1.33	1.09	0.93	0.76	0.80

Parallel FFT Solver Hybrid on Large Grids

Grid	Nodes	Processors	Seconds
512^3	1	32	7.794
1024^3	4	128	16.911
2048^3	32	1024	19.418
4096^3	256	8192	27.522

- Turkel's iterative method required 267 minutes with 10000 iterations for a 402^3 grid, roughly 65 million grid points.
- The parallel FFT method solved a problem roughly 103 times larger and approximately 572 times faster.

Parallel Partial FFT Solver Test Problem

Two tests are considered here. First, let $k = \sqrt{439.2}$. Consider the function

$$\phi(x) = \exp\left(ik(x+1)\right) + \exp\left(-ik(x-1)\right) - 2.$$

In the two-dimensional case the analytic solution on the domain $\Omega = [-1, 1] \times [-1, 1]$ is given by $u(x, y) = \phi(x)\phi(y)$. The right-hand side follows

$$f(\mathbf{x}) = -k^2 \left[\phi(x)\phi(y) - 2\phi(x) - 2\phi(y)\right].$$

The next test assumes that $k = 2\pi$ on the domain $\Omega = [0, 1] \times [0, 1]$.

Parallel Partial FFT Solver Accuracy

	Second-Order		Fourth-Order		Sixth-Order	
Grid	$\|\|\mathbf{u}-\mathbf{u}_a\|\|_\infty$	$\|\|A\mathbf{u}-\mathbf{f}\|\|_2$	$\|\|\mathbf{u}-\mathbf{u}_a\|\|_\infty$	$\|\|A\mathbf{u}-\mathbf{f}\|\|_2$	$\|\|\mathbf{u}-\mathbf{u}_a\|\|_\infty$	$\|\|A\mathbf{u}-\mathbf{f}\|\|_2$
33^2	1.63e+01	3.07e-12	1.18e+00	7.17e-11	3.36e-02	6.88e-13
65^2	3.58e+00	3.37e-12	7.07e-02	1.18e-10	4.38e-04	1.41e-12
129^2	8.47e-01	5.49e-12	4.41e-03	3.44e-10	6.57e-06	4.85e-12
257^2	2.09e-01	2.79e-11	2.76e-04	6.87e-10	1.02e-07	1.69e-11
513^2	5.22e-02	9.06e-11	1.72e-05	2.34e-09	1.63e-09	3.89e-11

The numerical and analytic solutions are given by $\mathbf{u}$ and $\mathbf{u}_a$ respectively.

Second Order Parallel Solver Comparison Uniform Grid

$N_x = N_y$	65	129	257	513	1025	2049	4097
Matlab's backslash	0.028	0.064	0.334	1.611	8.198	65.490	1421.4
Fast solver	0.01	0.02	0.06	0.23	1.05	4.58	-
Initialization	0.012	0.066	0.266	1.308	5.340	28.407	211.74
Generalize eigenvalue solver	0.001	0.003	0.014	0.080	0.315	1.850	13.749
Partial FFT solver	0.001	0.004	0.020	0.088	0.251	1.256	5.696

The wall-times are given in seconds.

- The "fast solver" was developed by Toivanen and Wolfmayr (2020).
- The fast solver times come from a Matlab implementation.

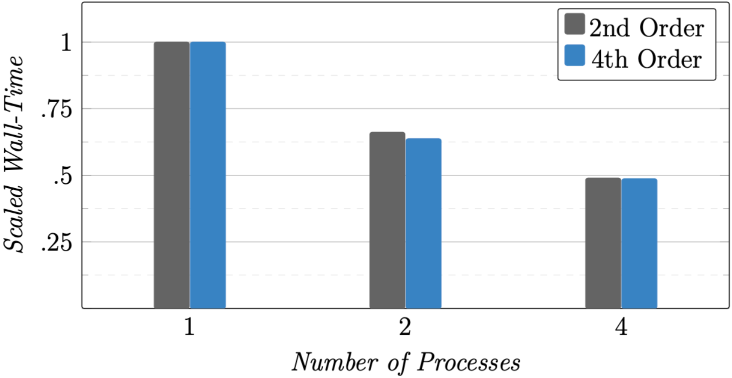
Second Order Parallel Solver Comparison Nonuniform Grid

N_x	65	65	2049	2049	2049	4097	4097
N_y	65	2049	65	2049	4097	2049	4097
Matlab's backslash	0.028	0.655	0.640	65.490	506.62	436.67	1421.4
Fast solver	0.01	0.10	0.12	4.58	-	-	-
Initialization	0.012	0.017	28.899	28.407	28.962	215.57	211.74
Generalize eigenvalue solver	0.001	0.007	0.620	1.850	3.371	7.489	13.749
Partial FFT solver	0.001	0.015	0.571	1.256	2.446	3.780	5.697

The wall-times are given in seconds.

- Faster solution with more grid points in the vertical direction due to parallelism.

Parallel Partial FFT Solver Time Reduction OpenMP



- Grid size of 4097^2.
- An extension to variable k, beyond the capability of the "fast solver" by Toivanen and Wolfmayr (2020).

Inclusion Test Problem

The function $k(\mathbf{x})$ is defined as

$$k(\mathbf{x}) = \begin{cases} 439.2 & \text{if } y < 0 \\ 1273 + 31i & \text{if } y \geq 0 \text{ and } y \notin S \\ 1050 + 2.26i & \text{if } y \in S \end{cases}$$

where $S = \{\mathbf{x} \in \Omega \mid |x - .3| \leq .15 \text{ and } |y - .2| < .04\}$ is the set of points within the rectangular inclusion with width .15, height .04 and center (.03,.2). The air and soil interface is at $y = 0$.

Inclusion Test Problem Continued

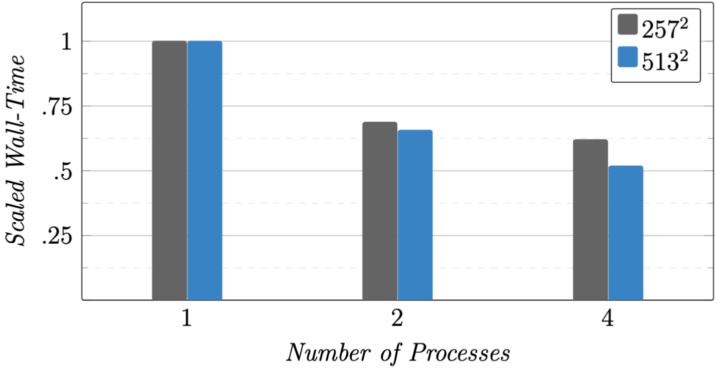
The domain considered is $\Omega = [-1,1] \times [-1,1]$ and right-hand side given by

$$f(\mathbf{x}) = \begin{cases} 0, & \text{outside inclusions,} \\ g(\mathbf{x}), & \text{inside inclusions} \end{cases}$$

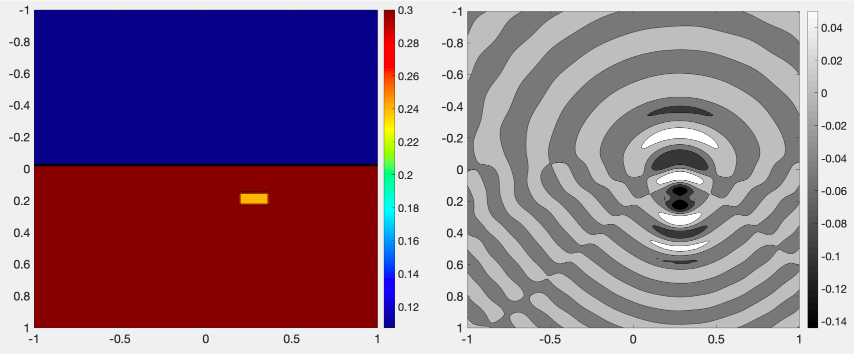
where $g(\mathbf{x})$ is a function corresponding to the electromagnetic signal produced by ground penetrating radar.

Subsurface Inclusion Accuracy and Parallel Time Reduction

Grid	Second-Order	Fourth-Order
129^2	5.64e-14	2.47e-14
257^2	1.63e-13	4.69e-14
513^2	3.83e-13	1.19e-13



Subsurface Inclusion and Solution Image



Conclusion

- Presented a novel direct parallel partial FFT-type algorithm for the numerical solutions of the two- and three-dimensional Helmholtz equations.
- Governing equations were discretized by high-order compact finite-difference schemes.
- Presented numerical results demonstrating the accuracy and scalability of the direct parallel method.

Thank you

- Family and friends
- Yury Gryazin, Ph.D.
- Yun Teck Lee, M.S.
- Idaho State University Department of Mathematics and Statistics

References

Sanjiva Lele. *Compact finite difference schemes with spectral-like resolution.* Journal of Computational Physics 103 (1992) 16-42.

Majid Nabavi, M.H. Kamran Siddiqui, and Javad Dargahi. *A new 9-point sixth-order accurate compact finite-difference method for the Helmholtz equation.* Journal of Sound and Vibration 307 (2007) 972-982.

Godehard Sutmann. *Compact finite difference schemes of sixth order for the Helmholtz equation.* Journal of Computational and Applied Mathematics 203 (2007) 15-31.

Eli Turkel, Dan Gordon, Rachel Gordon, and Semyon Tsynkov. *Compact 2D and 3D Sixth Order Schemes for the Helmholtz Equation with Variable Wave Number.* Journal of Computational Physics 232 (2013) 272-287.

- Howard Elman and Dianne O'Leary *Efficient iterative solution of the three-dimensional Helmholtz equation.* Journal of Computational Physics 142 (1998) 163-181.
- Yury Gryazin, Michael Klibanov, and Thomas Lucas. *GMRES Computation of High Frequency Electrical Field Propagation in Land Mine Detection.* Journal of Computational Physics 158 (2000) 98-115.
- Yun Teck Lee, Yury Gryazin, and Ronald Gonzales. *Scalable high-resolution algorithms for land-mine imaging problem.* SPIE Proceedings 11012 (2019) 241-250.
- Ronald Gonzales, Yury Gryazin, and Yun Teck Lee. *Parallel FFT algorithms for high-order approximations on three-dimensional compact stencils.* Parallel Computing 103 (2021) 102757.

Jari Toivanen and Monika Wolfmayr. *A fast Fourier transform based direct solver for the Helmholtz problem.* Numerical Linear Algebra with Applications, arXiv:1809.03808 (2020).

Ronald Gonzales and Yury Gryazin. *Parallel High-Resolution Compact Partial FFT-Type Direct Algorithms for Subsurface Scattering Problems.* SPIE Proceedings (2022)