Nonlocal UFL: Finite elements for Helmholtz equations with a nonlocal boundary condition

Robert Kirby[1] Andreas Klöckner[2] Ben Sepanski[3]
[1]Baylor University
[2]University of Illinois at Urbana-Champaign
[3]University of Texas at Austin
25 March 2021

Order of Presentation

Motivating Problem: Helmholtz scattering

A nonlocal boundary condition

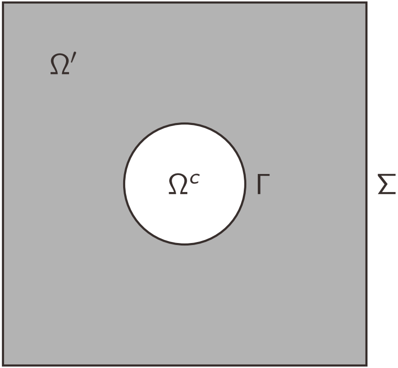
Nonlocal UFL

Numerical Results

Thanks to...

- NSF 1525697, 1909176
- The U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, Department of Energy Computational Science Graduate Fellowship under Award Number DE-SC0021110
- Luke Olson (UIUC)

Exterior scattering[1]



- Model waves reflecting off of obstacle Γ

$$\begin{cases} -\Delta u + \kappa^2 u = 0, & \mathbb{R}^d \setminus \Omega^c \\ -\frac{\partial u}{\partial n} = f, & \Gamma \end{cases}$$

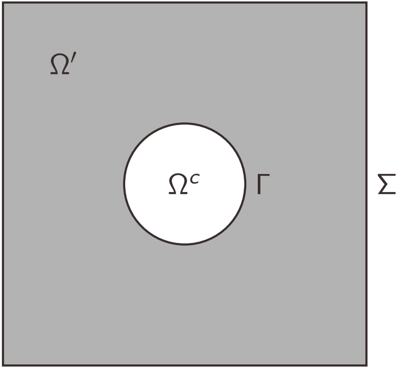
- *Without* any spurious reflections from infinity

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

- In some finite domain of interest $\Omega' \subseteq \mathbb{R}^d \setminus \Omega^c$ bounded by Σ.

[1] Colton and Kress 1998; Kress 1999.

Exterior scattering[1]



- Model waves reflecting off of obstacle Γ

$$\begin{cases} -\Delta u + \kappa^2 u = 0, & \mathbb{R}^d \setminus \Omega^c \\ -\frac{\partial u}{\partial n} = f, & \Gamma \end{cases}$$

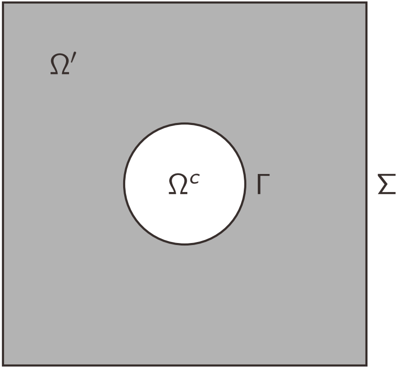
- *Without* any spurious reflections from infinity

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

- In some finite domain of interest $\Omega' \subseteq \mathbb{R}^d \setminus \Omega^c$ bounded by Σ.

[1] Colton and Kress 1998; Kress 1999.

Exterior scattering[1]



- Model waves reflecting off of obstacle Γ

$$\begin{cases} -\Delta u + \kappa^2 u = 0, & \mathbb{R}^d \setminus \Omega^c \\ -\frac{\partial u}{\partial n} = f, & \Gamma \end{cases}$$

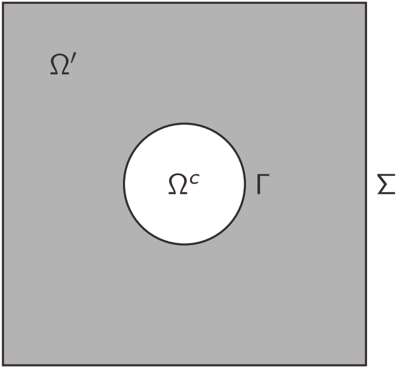
- *Without* any spurious reflections from infinity

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

- In some finite domain of interest $\Omega' \subseteq \mathbb{R}^d \setminus \Omega^c$ bounded by Σ.

[1] Colton and Kress 1998; Kress 1999.

Exterior scattering: computational problem



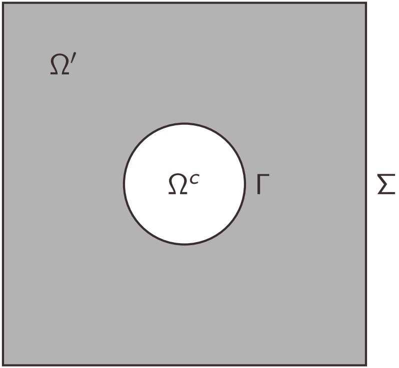
- Problem we want to solve

$$\begin{cases} -\Delta u + \kappa^2 u = 0, \\ -\frac{\partial u}{\partial n} = f, \\ \lim_{r\to\infty} r^{(d-1)/2}\left(\frac{\partial\ u}{\partial r} - i\kappa u\right) = 0 \end{cases}$$

- Problem we can actually solve

$$\begin{cases} -\Delta u + \kappa^2 u = 0, & \Omega' \\ -\frac{\partial u}{\partial n} = f, & \Gamma \\ ?????, & \Sigma \end{cases}$$

Exterior scattering: computational problem



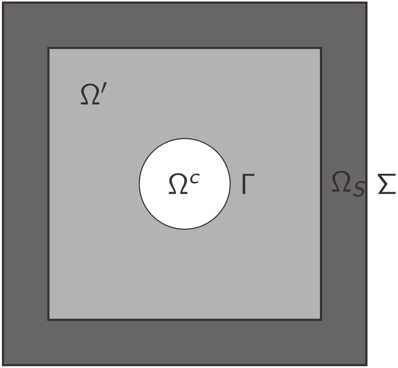
- Problem we want to solve

$$\begin{cases} -\Delta u + \kappa^2 u = 0, \\ -\frac{\partial u}{\partial n} = f, \\ \lim_{r\to\infty} r^{(d-1)/2} \left(\frac{\partial u}{\partial r} - i\kappa u\right) = 0 \end{cases}$$

- Problem we can actually solve

$$\begin{cases} -\Delta u + \kappa^2 u = 0, & \Omega' \\ -\frac{\partial u}{\partial n} = f, & \Gamma \\ ?????, & \Sigma \end{cases}$$

Exterior scattering: Perfectly Matched Layers (PML)[3]

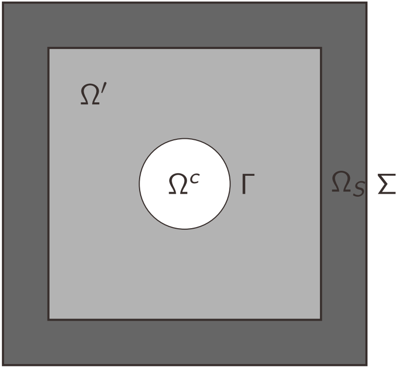


$$\begin{cases} -\nabla \cdot \beta(x)\nabla u + \kappa^2 u = 0, & \Omega' \\ -\frac{\partial u}{\partial n} = f, & \Gamma \\ u = 0, & \Sigma \end{cases}$$

- Ω': $\beta = I$, satisfies original equation
- Ω_S: β is a complex-valued coordinate transform to cause exponential decay in oscillating waves
- **Preconditioning is difficult!**[2]

[2] Engquist and Ying 2011; Safin, Minkoff, and Zweck 2018.
[3] Berenger 1994; Erlangga 2006; Bermudez et al. 2006.

Exterior scattering: Perfectly Matched Layers (PML)[3]

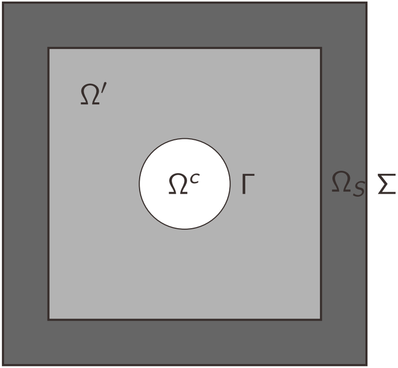


$$\begin{cases} -\nabla \cdot \beta(x) \nabla u + \kappa^2 u = 0, & \Omega' \\ -\frac{\partial u}{\partial n} = f, & \Gamma \\ u = 0, & \Sigma \end{cases}$$

- Ω': $\beta = I$, satisfies original equation
- Ω_S: β is a complex-valued coordinate transform to cause exponential decay in oscillating waves
- **Preconditioning is difficult!**[2]

[2]Engquist and Ying 2011; Safin, Minkoff, and Zweck 2018.
[3]Berenger 1994; Erlangga 2006; Bermudez et al. 2006.

Exterior scattering: Perfectly Matched Layers (PML)[3]

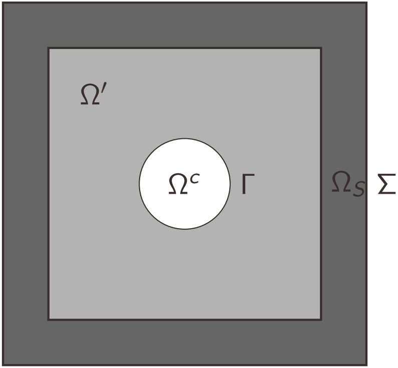


$$\begin{cases} -\nabla \cdot \beta(x)\nabla u + \kappa^2 u = 0, & \Omega' \\ -\frac{\partial u}{\partial n} = f, & \Gamma \\ u = 0, & \Sigma \end{cases}$$

- Ω': $\beta = I$, satisfies original equation
- Ω_S: β is a complex-valued coordinate transform to cause exponential decay in oscillating waves
- **Preconditioning is difficult!**[2]

[2] Engquist and Ying 2011; Safin, Minkoff, and Zweck 2018.
[3] Berenger 1994; Erlangga 2006; Bermudez et al. 2006.

Exterior scattering: Perfectly Matched Layers (PML)[3]



$$
\begin{cases}
-\nabla \cdot \beta(x)\nabla u + \kappa^2 u = 0, & \Omega' \\
-\frac{\partial u}{\partial n} = f, & \Gamma \\
u = 0, & \Sigma
\end{cases}
$$

- Ω': $\beta = I$, satisfies original equation
- Ω_S: β is a complex-valued coordinate transform to cause exponential decay in oscillating waves
- **Preconditioning is difficult!**[2]

[2] Engquist and Ying 2011; Safin, Minkoff, and Zweck 2018.
[3] Berenger 1994; Erlangga 2006; Bermudez et al. 2006.

Integral form of the solution

Using the Helmholtz Green's function $\mathcal{K}(\mathrm{x}) = \dfrac{i}{4\pi\,|\mathrm{x}|} e^{i\kappa|\mathrm{x}|}$,

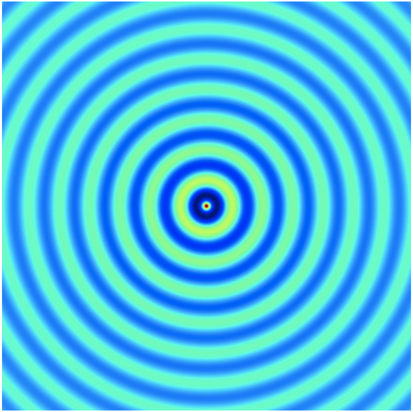


Figure: $\mathcal{K}$ in 2D

Integral form of the solution

Using the Helmholtz Green's function

$$\mathcal{K}(\mathrm{x}) = \frac{i}{4\pi\,|\mathrm{x}|} e^{i\kappa|\mathrm{x}|},$$

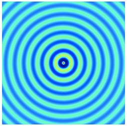


Figure: $\mathcal{K}$ in 2D

the *true solution* satisfies[4]

$$u(x) = D(u)(x) - S\left(\frac{\partial u}{\partial n}\right)(x), \quad x \in \Omega'$$

where

$$D(u)(x) = \int_\Gamma \left(\frac{\partial}{\partial n}\mathcal{K}(x-y)\right) u(y)\,dy,$$

$$S(u)(x) = \int_\Gamma \mathcal{K}(x-y)u(y)\,dy$$

[4] Colton and Kress 1998; Kress 1999.

Integral form of the solution

Using the Helmholtz Green's function

$$\mathcal{K}(\mathrm{x}) = \frac{i}{4\pi\,|\mathrm{x}|} e^{i\kappa|\mathrm{x}|},$$

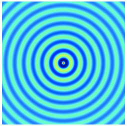


Figure: $\mathcal{K}$ in 2D

the *true solution* satisfies[4]

$$u(x) = D(u)(x) - S(\tfrac{\partial u}{\partial n})(x), \quad x \in \Omega'$$

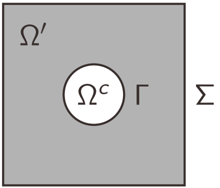
where

$$D(u)(x) = \int_\Gamma \left(\tfrac{\partial}{\partial n}\mathcal{K}(x-y)\right) u(y)\,dy,$$

$$S(u)(x) = \int_\Gamma \mathcal{K}(x-y)u(y)\,dy$$

[4] Colton and Kress 1998; Kress 1999.

Exterior scattering



Exact boundary conditions

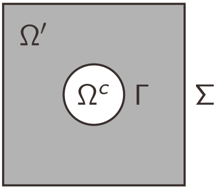
$$\begin{cases} -\Delta u + \kappa^2 u = 0, & \Omega' \\ -\frac{\partial u}{\partial n} = f, & \Gamma \\ \left(i\kappa - \frac{\partial}{\partial n}\right)\left(u - D(u) + S(f)\right) = 0, & \Sigma \end{cases}$$

Variational Form:

For all $v \in H^1(\Omega')$

$$\begin{aligned} (\nabla u, \nabla v) - \kappa^2 (u, v) - i\kappa\langle u, v\rangle_\Sigma + \langle \left(i\kappa - \tfrac{\partial}{\partial n}\right) D(u), v\rangle_\Sigma \\ = \langle f, v\rangle_\Gamma + \langle \left(i\kappa - \tfrac{\partial}{\partial n}\right) S(f), v\rangle_\Sigma \end{aligned}$$

Exterior scattering



Exact boundary conditions

$$\begin{cases} -\Delta u + \kappa^2 u = 0, & \Omega' \\ -\frac{\partial u}{\partial n} = f, & \Gamma \\ \left(i\kappa - \frac{\partial}{\partial n}\right)\left(u - D(u) + S(f)\right) = 0, & \Sigma \end{cases}$$

Variational Form:

For all $v \in H^1(\Omega')$

$$\begin{aligned} (\nabla u, \nabla v) - \kappa^2 (u, v) - i\kappa\langle u, v\rangle_\Sigma + \langle \left(i\kappa - \tfrac{\partial}{\partial n}\right) D(u), v\rangle_\Sigma \\ = \langle f, v\rangle_\Gamma + \langle \left(i\kappa - \tfrac{\partial}{\partial n}\right) S(f), v\rangle_\Sigma \end{aligned}$$

Theory

- a is a bounded bilinear form on $H^1 \times H^1$
- F is a bounded linear functional on H^1
- Gårding inequality. There exist M and an $\alpha > 0$ such that

$$\mathrm{Re}(a(u,u)) + M \|u\|^2 \geq \alpha \|u\|^2_{H^1(\Omega)}.$$

- For $h \leq h_0$, we have optimal-order H^1 and L^2 error estimates.[5]

[5] Kirby, Klöckner, and Sepanski 2021.

Theory

- a is a bounded bilinear form on $H^1 \times H^1$
- F is a bounded linear functional on H^1
- Gårding inequality. There exist M and an $\alpha > 0$ such that

$$\mathrm{Re}(a(u, u)) + M \|u\|^2 \geq \alpha \|u\|^2_{H^1(\Omega)} .$$

- For $h \leq h_0$, we have optimal-order H^1 and L^2 error estimates.[5]

[5] Kirby, Klöckner, and Sepanski 2021.

Theory

- a is a bounded bilinear form on $H^1 \times H^1$
- F is a bounded linear functional on H^1
- Gårding inequality. There exist M and an $\alpha > 0$ such that

$$\mathrm{Re}(a(u,u)) + M \|u\|^2 \geq \alpha \|u\|^2_{H^1(\Omega)} .$$

- For $h \leq h_0$, we have optimal-order H^1 and L^2 error estimates.[5]

[5] Kirby, Klöckner, and Sepanski 2021.

Theory

- a is a bounded bilinear form on $H^1 \times H^1$
- F is a bounded linear functional on H^1
- Gårding inequality. There exist M and an $\alpha > 0$ such that

$$\operatorname{Re}(a(u,u)) + M\|u\|^2 \geq \alpha \|u\|^2_{H^1(\Omega)}.$$

- For $h \leq h_0$, we have optimal-order H^1 and L^2 error estimates.[5]

[5] Kirby, Klöckner, and Sepanski 2021.

Theory

- a is a bounded bilinear form on $H^1 \times H^1$
- F is a bounded linear functional on H^1
- Gårding inequality. There exist M and an $\alpha > 0$ such that

$$\mathrm{Re}(a(u,u)) + M \|u\|^2 \geq \alpha \|u\|^2_{H^1(\Omega)}.$$

- For $h \leq h_0$, we have optimal-order H^1 and L^2 error estimates.[5]

[5] Kirby, Klöckner, and Sepanski 2021.

Nonlocal operations in UFL

- *Recall:*

$$D(u)(x) = \int_\Gamma \frac{\partial}{\partial n}\mathcal{K}(x-y)u(y)\,dy, \qquad x \in \Sigma$$

- *Problem:* Nonlocal operations have large support (all of Σ!)
  - This makes our stiffness matrix dense, especially in 3D
  - *Solution:* Firedrake's matrix-free evaluation
- *Problem:* Naive evaluation of layer potentials is slow:
  - $\text{ndof}(\Gamma) \cdot \text{ndof}(\Sigma)$
  - *Solution:* Fast multipole methods (FMM)[6]: use the structure of $\mathcal{K}$ to compute the potential in *linear time* with low-rank approximations

[6] Carrier, Greengard, and Rokhlin 1988.

Nonlocal operations in UFL

- *Recall:*

$$D(u)(x) = \int_\Gamma \frac{\partial}{\partial n}\mathcal{K}(x-y)u(y)\,dy, \qquad x \in \Sigma$$

- *Problem:* Nonlocal operations have large support (all of Σ!)
  - This makes our stiffness matrix dense, especially in 3D
  - *Solution:* Firedrake's matrix-free evaluation

- *Problem:* Naive evaluation of layer potentials is slow:
  - $\mathrm{ndof}(\Gamma) \cdot \mathrm{ndof}(\Sigma)$
  - *Solution:* Fast multipole methods (FMM)[6]: use the structure of $\mathcal{K}$ to compute the potential in *linear time* with low-rank approximations

[6] Carrier, Greengard, and Rokhlin 1988.

Nonlocal operations in UFL

- *Recall:*

$$D(u)(x) = \int_\Gamma \frac{\partial}{\partial n}\mathcal{K}(x-y)u(y)\,dy, \qquad x \in \Sigma$$

- *Problem:* Nonlocal operations have large support (all of Σ!)
  - This makes our stiffness matrix dense, especially in 3D
  - *Solution:* Firedrake's matrix-free evaluation
- *Problem:* Naive evaluation of layer potentials is slow:
  - $\mathrm{ndof}(\Gamma)\cdot\mathrm{ndof}(\Sigma)$
  - *Solution:* Fast multipole methods (FMM)[6]: use the structure of $\mathcal{K}$ to compute the potential in *linear time* with low-rank approximations

[6] Carrier, Greengard, and Rokhlin 1988.

Nonlocal operations in UFL

- *Recall:*

$$D(u)(x) = \int_\Gamma \frac{\partial}{\partial n}\mathcal{K}(x-y)u(y)\,dy, \qquad x \in \Sigma$$

- *Problem:* Nonlocal operations have large support (all of Σ!)
  - This makes our stiffness matrix dense, especially in 3D
  - *Solution:* Firedrake's matrix-free evaluation

- *Problem:* Naive evaluation of layer potentials is slow:
  - $\mathrm{ndof}(\Gamma) \cdot \mathrm{ndof}(\Sigma)$
  - *Solution:* Fast multipole methods (FMM)[6]: use the structure of $\mathcal{K}$ to compute the potential in *linear time* with low-rank approximations

[6] Carrier, Greengard, and Rokhlin 1988.

Nonlocal operations in UFL

- *Recall:*

$$D(u)(x) = \int_\Gamma \frac{\partial}{\partial n}\mathcal{K}(x-y)u(y)\,dy, \qquad x \in \Sigma$$

- *Problem:* Nonlocal operations have large support (all of Σ!)
  - This makes our stiffness matrix dense, especially in 3D
  - *Solution:* Firedrake's matrix-free evaluation
- *Problem:* Naive evaluation of layer potentials is slow:
  - $\mathrm{ndof}(\Gamma) \cdot \mathrm{ndof}(\Sigma)$
  - *Solution:* Fast multipole methods (FMM)[6]: use the structure of $\mathcal{K}$ to compute the potential in *linear time* with low-rank approximations

[6] Carrier, Greengard, and Rokhlin 1988.

Nonlocal operations in UFL

- *Recall:*

$$D(u)(x) = \int_{\Gamma} \frac{\partial}{\partial n}\mathcal{K}(x-y)u(y)\,dy, \qquad x \in \Sigma$$

- *Problem:* Nonlocal operations have large support (all of Σ!)
  - This makes our stiffness matrix dense, especially in 3D
  - *Solution:* Firedrake's matrix-free evaluation
- *Problem:* Naive evaluation of layer potentials is slow:
  - $\mathrm{ndof}(\Gamma) \cdot \mathrm{ndof}(\Sigma)$
  - *Solution:* Fast multipole methods (FMM)[6]: use the structure of $\mathcal{K}$ to compute the potential in *linear time* with low-rank approximations

[6] Carrier, Greengard, and Rokhlin 1988.

Nonlocal operations in UFL

- *Recall:*

$$D(u)(x) = \int_{\Gamma} \frac{\partial}{\partial n}\mathcal{K}(x-y)u(y)\,dy, \qquad x \in \Sigma$$

- *Problem:* Nonlocal operations have large support (all of Σ!)
  - This makes our stiffness matrix dense, especially in 3D
  - *Solution:* Firedrake's matrix-free evaluation
- *Problem:* Naive evaluation of layer potentials is slow:
  - $\mathrm{ndof}(\Gamma) \cdot \mathrm{ndof}(\Sigma)$
  - *Solution:* Fast multipole methods (FMM)[6]: use the structure of $\mathcal{K}$ to compute the potential in *linear time* with low-rank approximations

[6] Carrier, Greengard, and Rokhlin 1988.

Nonlocal operations in UFL

- *Recall:*

$$D(u)(x) = \int_\Gamma \frac{\partial}{\partial n}\mathcal{K}(x-y)u(y)\,dy, \qquad x \in \Sigma$$

- *Problem:* Nonlocal operations have large support (all of Σ!)
  - This makes our stiffness matrix dense, especially in 3D
  - *Solution:* Firedrake's matrix-free evaluation

- *Problem:* Naive evaluation of layer potentials is slow:
  - $\mathrm{ndof}(\Gamma) \cdot \mathrm{ndof}(\Sigma)$
  - *Solution:* Fast multipole methods (FMM)[6]: use the structure of $\mathcal{K}$ to compute the potential in *linear time* with low-rank approximations

[6] Carrier, Greengard, and Rokhlin 1988.

Nonlocal operations in UFL: Marshalling pytential[8]

- Build `LayerPotential` as a UFL External Operator[7]
  - ✓ Build pytential representation of domain of interest
  - ✓ Build pytential representation of function space
  - ✓ Build efficient converter between pytential and firedrake representations
  - – Fully support automatic differentiation
- Evaluation of $\langle (i\kappa - \frac{\partial}{\partial n})D(u), v\rangle_\Sigma$
  - ✓ `LayerPotential` evaluates $D(u)$ (automatically uses pytential, which employs FMM to compute the potential)
  - ✓ Firedrake evaluates inner product

[7] N. Bouziani, *External Operators*: https://fenics2021.com/talks/bouziani.html
[8] Klöckner 2020.

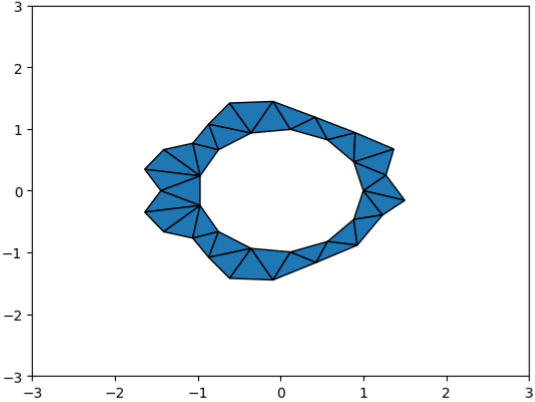
Nonlocal operations in UFL: Marshalling pytential[8]

- Build `LayerPotential` as a UFL External Operator[7]
  - ✓ Build pytential representation of domain of interest
  - ✓ Build pytential representation of function space
  - ✓ Build efficient converter between pytential and firedrake representations
  - – Fully support automatic differentiation
- Evaluation of $\langle (i\kappa - \frac{\partial}{\partial n}) D(u), v \rangle_\Sigma$
  - ✓ `LayerPotential` evaluates $D(u)$ (automatically uses pytential, which employs FMM to compute the potential)
  - ✓ Firedrake evaluates inner product

[7] N. Bouziani, *External Operators*: https://fenics2021.com/talks/bouziani.html
[8] Klöckner 2020.

Nonlocal operations in UFL: Marshalling pytential[8]

- Build `LayerPotential` as a UFL External Operator[7]
  - ✓ Build pytential representation of domain of interest



[7] N. Bouziani, *External Operators*: `https://fenics2021.com/talks/bouziani.html`
[8] Klöckner 2020.

Nonlocal operations in UFL: Marshalling pytential[8]

- Build `LayerPotential` as a UFL External Operator[7]
  - ✓ Build pytential representation of domain of interest
  - ✓ Build pytential representation of function space
  - ✓ Build efficient converter between pytential and firedrake representations
  - – Fully support automatic differentiation
- Evaluation of $\langle (i\kappa - \frac{\partial}{\partial n}) D(u), v \rangle_\Sigma$
  - ✓ `LayerPotential` evaluates $D(u)$ (automatically uses pytential, which employs FMM to compute the potential)
  - ✓ Firedrake evaluates inner product

[7] N. Bouziani, *External Operators*: https://fenics2021.com/talks/bouziani.html
[8] Klöckner 2020.

Nonlocal operations in UFL: Marshalling pytential[8]

- Build `LayerPotential` as a UFL External Operator[7]
  - ✓ Build pytential representation of domain of interest
  - ✓ Build pytential representation of function space
  - ✓ Build efficient converter between pytential and firedrake representations
  - – Fully support automatic differentiation
- Evaluation of $\langle (i\kappa - \frac{\partial}{\partial n}) D(u), v \rangle_\Sigma$
  - ✓ `LayerPotential` evaluates $D(u)$ (automatically uses pytential, which employs FMM to compute the potential)
  - ✓ Firedrake evaluates inner product

[7] N. Bouziani, *External Operators*: https://fenics2021.com/talks/bouziani.html
[8] Klöckner 2020.

Nonlocal operations in UFL: Marshalling pytential[8]

- Build `LayerPotential` as a UFL External Operator[7]
  - ✓ Build pytential representation of domain of interest
  - ✓ Build pytential representation of function space
  - ✓ Build efficient converter between pytential and firedrake representations
  - – Fully support automatic differentiation
- Evaluation of $\langle (i\kappa - \frac{\partial}{\partial n}) D(u), v \rangle_\Sigma$
  - ✓ `LayerPotential` evaluates $D(u)$ (automatically uses pytential, which employs FMM to compute the potential)
  - ✓ Firedrake evaluates inner product

[7] N. Bouziani, *External Operators*: https://fenics2021.com/talks/bouziani.html
[8] Klöckner 2020.

Nonlocal operations in UFL: Marshalling pytential[8]

- Build `LayerPotential` as a UFL External Operator[7]
  - ✓ Build pytential representation of domain of interest
  - ✓ Build pytential representation of function space
  - ✓ Build efficient converter between pytential and firedrake representations
  - – Fully support automatic differentiation
- Evaluation of $\langle (i\kappa - \frac{\partial}{\partial n}) D(u), v \rangle_\Sigma$
  - ✓ `LayerPotential` evaluates $D(u)$ (automatically uses pytential, which employs FMM to compute the potential)
  - ✓ Firedrake evaluates inner product

[7] N. Bouziani, *External Operators*: https://fenics2021.com/talks/bouziani.html
[8] Klöckner 2020.

Nonlocal operations in UFL: Marshalling pytential[8]

- Build `LayerPotential` as a UFL External Operator[7]
  - ✓ Build pytential representation of domain of interest
  - ✓ Build pytential representation of function space
  - ✓ Build efficient converter between pytential and firedrake representations
  - – Fully support automatic differentiation
- Evaluation of $\langle (i\kappa - \frac{\partial}{\partial n}) D(u), v \rangle_\Sigma$
  - ✓ `LayerPotential` evaluates $D(u)$ (automatically uses pytential, which employs FMM to compute the potential)
  - ✓ Firedrake evaluates inner product

[7] N. Bouziani, *External Operators*: https://fenics2021.com/talks/bouziani.html
[8] Klöckner 2020.

Nonlocal operations in UFL: Marshalling pytential[8]

- Build `LayerPotential` as a UFL External Operator[7]
  - ✓ Build pytential representation of domain of interest
  - ✓ Build pytential representation of function space
  - ✓ Build efficient converter between pytential and firedrake representations
  - – Fully support automatic differentiation
- Evaluation of $\left\langle (i\kappa - \frac{\partial}{\partial n})D(u), v \right\rangle_\Sigma$
  - ✓ `LayerPotential` evaluates $D(u)$ (automatically uses pytential, which employs FMM to compute the potential)
  - ✓ Firedrake evaluates inner product

[7] N. Bouziani, *External Operators*: https://fenics2021.com/talks/bouziani.html
[8] Klöckner 2020.

Solving the system with Firedrake

Extend UFL:

$$a(u,v) = (\nabla u, \nabla v) - \kappa^2 (u,v) - i\kappa\langle u,v\rangle_\Sigma + \langle \left(i\kappa - \tfrac{\partial}{\partial n}\right) D(u) v\rangle_\Sigma$$

Solving the system with Firedrake

Extend UFL:

$$a(u, v) = (\nabla u, \nabla v) - \kappa^2 (u, v) - i\kappa \langle u, v \rangle_\Sigma + \langle \left(i\kappa - \tfrac{\partial}{\partial n} \right) D(u) v \rangle_\Sigma$$

Will be written as:

```
Du = DoubleLayerPotential(u, HelmholtzKernel(dim=2),
                          source=gamma, target=sigma)
a = inner(grad(u), grad(v)) * dx - \
    kappa**2 * inner(u, v) * dx - \
    i * kappa * inner(u, v) * ds(sigma) + \
    i * kappa * inner(Du, v) * ds(sigma) - \
    inner(dot(grad(Du), n), v) * ds(sigma)
```

Solving the system with Firedrake

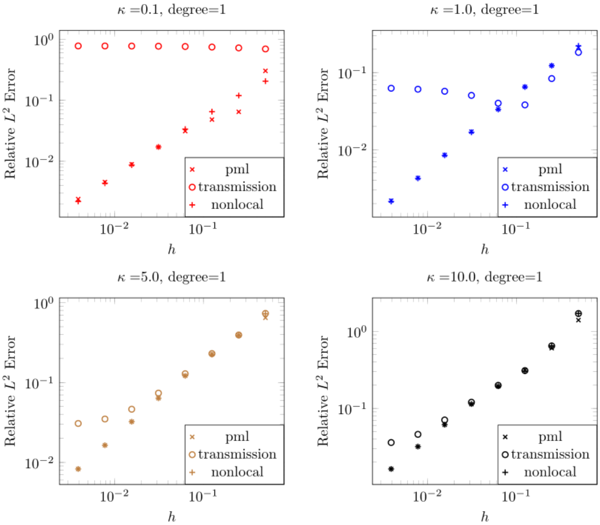
Extend UFL:

$$a(u, v) = (\nabla u, \nabla v) - \kappa^2 (u, v) - i\kappa \langle u, v \rangle_\Sigma + \langle \left(i\kappa - \frac{\partial}{\partial n} \right) D(u) v \rangle_\Sigma$$

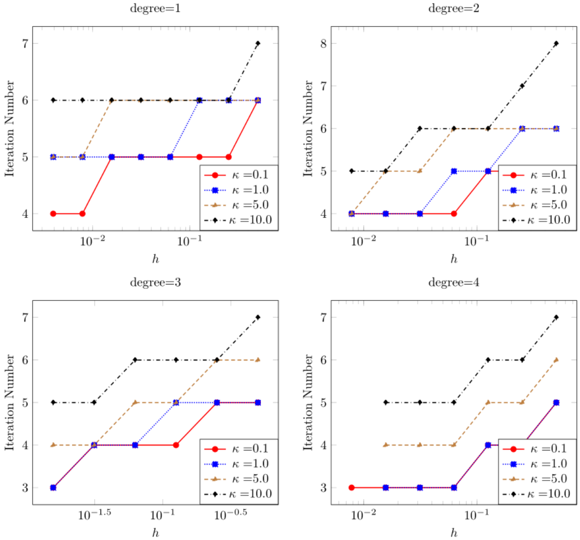
Will be written as:

```
Du = DoubleLayerPotential(u, HelmholtzKernel(dim=2),
                          source=gamma, target=sigma)
a = inner(grad(u), grad(v)) * dx - \
    kappa**2 * inner(u, v) * dx - \
    i * kappa * inner(u, v) * ds(sigma) + \
    i * kappa * inner(Du, v) * ds(sigma) - \
    inner(dot(grad(Du), n), v) * ds(sigma)
```

Numerical results: 2D

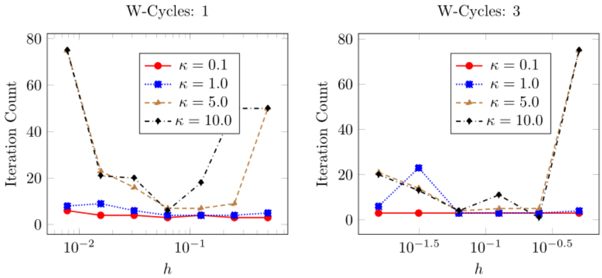


Preconditioning: LU of local part



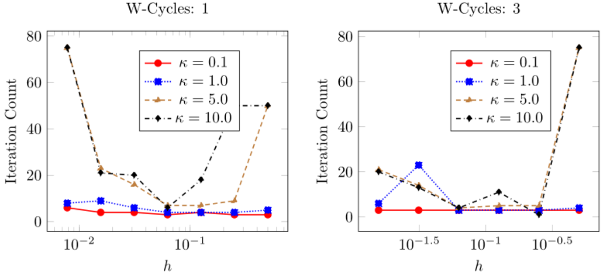
Preconditioning: PyAMG

- If we can find a good preconditioner for the local problem, we get a good preconditioner for the nonlocal problem
- PyAMG: precondition with plane waves



Preconditioning: PyAMG

- If we can find a good preconditioner for the local problem, we get a good preconditioner for the nonlocal problem
- PyAMG: precondition with plane waves



Conclusion

Results

- Novel nonlocal boundary condition
  - Error estimates [9]
- Extension of UFL to efficiently handle nonlocal operators
- Numerical experiments demonstrating optimal-order convergence
- Investigation into preconditioners

Coming Soon

- Full implementation of `LayerPotentials` and `VolumePotential`[10]s in UFL as External Operator[11]s
- General theory for this method and application to more problems

[10] Kirby, Klöckner, and Sepanski 2021.
[11] X. Wei, *IEM-FEM Coupling*: `https://fenics2021.com/talks/wei.html`
[12] N. Bouziani, *External Operators*: `https://fenics2021.com/talks/bouziani.html`

References I

- Berenger, Jean-Pierre (1994). "A perfectly matched layer for the absorption of electromagnetic waves". In: *Journal of Computational Physics* 114.2, pp. 185–200. ISSN: 0021-9991. DOI: `https://doi.org/10.1006/jcph.1994.1159`. URL: `https://www.sciencedirect.com/science/article/pii/S0021999184711594`.
- Bermudez, Alfredo et al. (Jan. 2006). "An optimal finite-element/pml method for the simulation of acoustic wave propagation phenomena". In: *Variational Formulations in Mechanics: Theory and Applications*. Citation Key: Bermudez:2006.

References II

- Carrier, J., Leslie Greengard, and Vladimir Rokhlin (July 1988). "A Fast Adaptive Multipole Algorithm for Particle Simulations". In: *SIAM Journal on Scientific and Statistical Computing* 9.4, pp. 669–686. ISSN: 0196-5204, 2168-3417. DOI: 10.1137/0909044.
- Colton, David and Rainer Kress (Jan. 1998). *Inverse Acoustic and Electromagnetic Scattering Theory*. 2nd ed. Springer. ISBN: 3-540-62838-X.

References III

Engquist, Björn and Lexing Ying (2011). "Sweeping preconditioner for the Helmholtz equation: Hierarchical matrix representation". In: *Communications on Pure and Applied Mathematics* 64.5, pp. 697–735. DOI: https://doi.org/10.1002/cpa.20358. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/cpa.20358. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/cpa.20358.

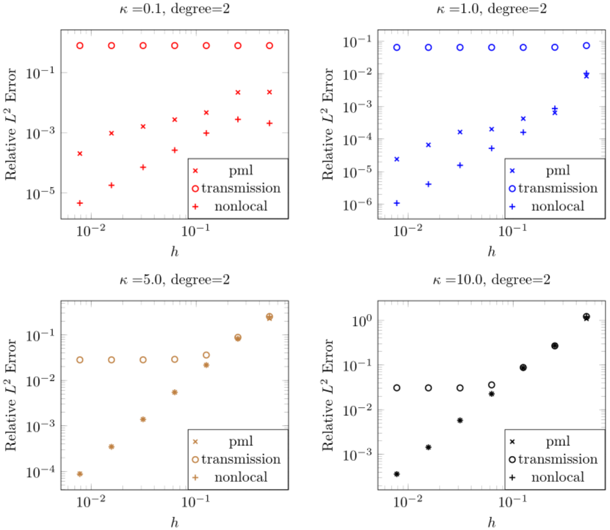
Erlangga, Y. A. (2006). "A preconditioner for the Helmholtz equation with perfectly matched layer". In: URL: https://repository.tudelft.nl/islandora/object/uuid%3A241032bc-7485-41cb-8f2a-e36ad88fbc08.

References IV

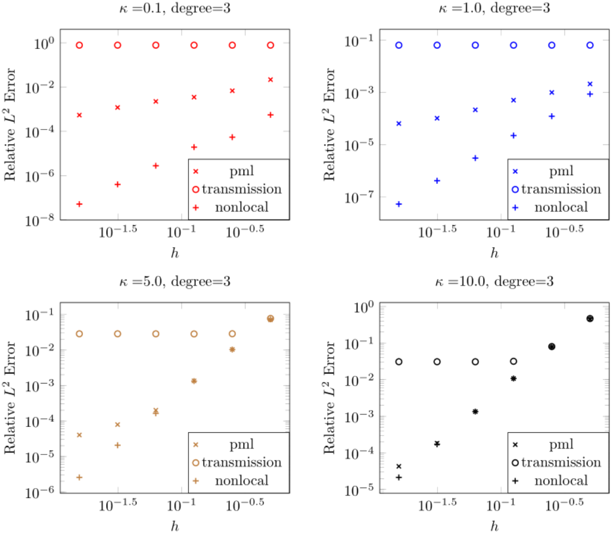
- Kirby, Robert C., Andreas Klöckner, and Ben Sepanski (2021). *Finite elements for Helmholtz equations with a nonlocal boundary condition*. arXiv: 2009.08493 [math.NA].
- Klöckner, Andreas (2020). "pytential Source Code Repository". In: URL: https://github.com/inducer/pytential.
- Kress, Rainer (Jan. 1999). *Linear Integral Equations*. Springer. ISBN: 978-0-387-98700-2.
- Safin, Artur, Susan Minkoff, and John Zweck (Jan. 2018). "A Preconditioned Finite Element Solution of the Coupled Pressure-Temperature Equations Used to Model Trace Gas Sensors". In: *SIAM Journal on Scientific Computing* 40.5, B1470–B1493. ISSN: 1064-8275, 1095-7197. DOI: 10.1137/17M1145823.

Backup Slides

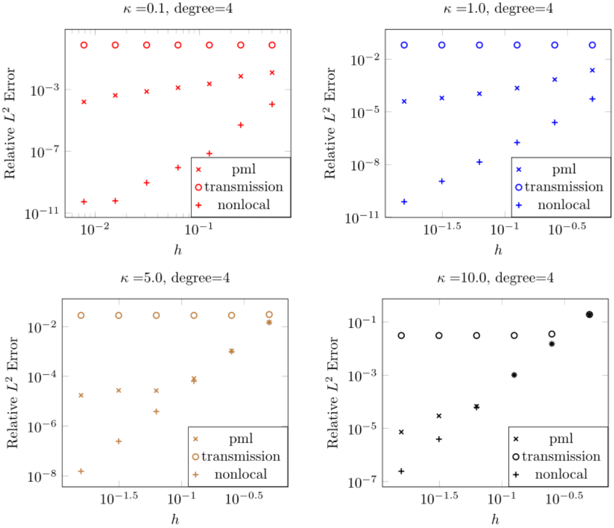
Numerical results: 2D, degree 2



Numerical results: 2D, degree 3



Numerical results: 2D, degree 4



Numerical results: 3D, degree 1