Méthode de Décomposition de Domaines Quasi Optimale pour l’Équation d’Helmholtz

X. Antoine, Y. Boubendir, C. Geuzaine et B. Thierry

Université de Liège, Belgique
Université de Nancy
NJIT, USA

ANR MicroWave, 22 Février 2012
Outline

Introduction to DDM

Transmission boundary operators

Numerical examples

Improvement in GetDP

Conclusion and perspectives
Introduction to DDM

Transmission boundary operators

Numerical examples

Improvement in GetDP

Conclusion and perspectives
Introduction to DDM

Sound-soft acoustic scattering problem

\[
\begin{aligned}
\Delta u + k^2 u &= 0 \\
u &= f \\
\lim_{|x| \to \infty} |x| (\partial_x u - iku) &= 0.
\end{aligned}
\]

- \(K\): bounded open subset of \(\mathbb{R}^3\) (scatterer),
- \(f\): fixed by a plane wave: \(f = -e^{ik\alpha \cdot x}\) with \(x = (x_1, x_2, x_3) \in \mathbb{R}^3\),
- \(i = \sqrt{-1}\),
- \(\alpha\): incidence angle normalized on the unit sphere (\(|\alpha| = 1\)),
- \(k\): wavenumber related to the wavelength \(\lambda\) by: \(k = 2\pi/\lambda\).

Last equation: Sommerfeld radiation condition at infinity (the scattered wave is outgoing).
Introduction to DDM

Approximation by truncation

System (4) is approximated by

\[
\begin{aligned}
\Delta u + k^2 u &= 0 \quad \text{in } \Omega, \\
u &= f \quad \text{on } \Gamma, \\
\partial_n u + \mathcal{B} u &= 0 \quad \text{on } \Gamma^\infty.
\end{aligned}
\]

- \( n \): outwardly directed unit normal to \( \Gamma^\infty \),
- \( \Omega \): bounded domain enclosed by the fictitious boundary \( \Gamma^\infty \) and \( \Gamma \),
- The operator \( \mathcal{B} \) represents an approximation of the DtN operator (for example \( \mathcal{B} = -ik \)) on \( \Gamma^\infty \).
Main difficulties

- Finite element method leads to large, complex-valued and highly indefinite sparse matrix.
- This is worst when considering large wavenumbers (high frequency regime).
- Convergence breakdown of Krylov subspace solvers (GMRES): requires a robust and efficient preconditioner: currently not available (see the work of Ibrahim).
- **Alternative**: Domain Decomposition Methods (since about 20 years) where (small) subdomain problems can be solved efficiently by direct solvers on parallel computers.
Introduction to DDM

Figure: Example of a two-dimensional non-overlapping domain decomposition method.
Introduction to DDM

Lions-Desprès DDM: step 1

Splitting $\Omega$ into $N_{\text{dom}}$ subdomains $\Omega_i$, $i = 1, \ldots, N_{\text{dom}}$, such that (see Figure 1):

- $\overline{\Omega} = \bigcup_{i=1}^{N_{\text{dom}}} \overline{\Omega}_i$
- $\Omega_i \cap \Omega_j = \emptyset$, if $i \neq j$, ($i, j = 1, \ldots, N_{\text{dom}}$)
- $\partial \Omega_i \cap \partial \Omega_j = \Sigma_{ij} = \Sigma_{ji}$ ($i, j = 1, \ldots, N_{\text{dom}}$) is the artificial interface separating $\Omega_i$ and $\Omega_j$ as long as its interior $\Sigma_{ij}$ is not empty.
- $\Gamma_i = \Gamma \cap \partial \Omega_i$ and $\Gamma_{\infty} = \Gamma_{\infty} \cap \partial \Omega_i$ for $i = 1, \ldots, N_{\text{dom}}$. 
Introduction to DDM

Lions-Desprès DDM: step 2

Reducing the solution of the initial problem (5) by solving the local \textit{transmission} problems

\[
\begin{aligned}
\Delta u_i^{(n+1)} + k^2 u_i^{(n+1)} &= 0 & \text{in } \Omega_i, \\
u_i^{(n+1)} &= f_i & \text{on } \Gamma_i, \\
\partial_n u_i^{(n+1)} + B u_i^{(n+1)} &= 0 & \text{on } \Gamma_i^\infty, \\
\partial_n u_i^{(n+1)} + S u_i^{(n+1)} &= g_{ij}^{(n)} & \text{on } \Sigma_{ij}.
\end{aligned}
\]  

(1.1a)
Introduction to DDM

Lions-Desprès DDM: step 2

...and then in forming the quantities to be transmitted through the interfaces

$$g_{ji}^{(n+1)} = -\partial_n u_i^{(n+1)} + Su_i^{(n+1)} = -g_{ij}^{(n)} + 2Su_i^{(n+1)} \quad \text{on } \Sigma_{ij},$$

(1.2)

where $u_i = u|_{\Omega_i}$, $n_i$ (resp. $n_j$) is the outward unit normal to the boundary of $\Omega_i$ (resp. $\Omega_j$), $i = 1, \ldots, N_{\text{dom}}$, $j = 1, \ldots, N_{\text{dom}}$, $\Gamma_i = \partial\Omega_i \cap \Gamma$, $\Gamma_i^\infty = \partial\Omega_i \cap \Gamma^\infty$.

Lions-Desprès DDM: crucial point:

$S$: an invertible operator to choose.
Introduction to DDM

Remarks:

▶ The boundary condition on \( \Gamma_i \) (resp. \( \Gamma_i^\infty \)) does not take place if the interior of \( \partial \Omega_i \cap \Gamma \) (resp. \( \partial \Omega_i \cap \Gamma^\infty \)) is the empty set

▶ We will assume in all that follows that the DDM is well-posed, in the sense that each subproblem (1.1a)-(1.1b) is itself well-posed, i.e., away from interior resonances.
Introduction to DDM

Iteration operator (Jacobi version)

Solving at each step all the local transmission problems through (1.1)-(1.2) may be recast as one application of the iteration operator $A : \times_{i,j=1}^{N_{\text{dom}}} L^2(\Sigma_{ij}) \rightarrow \times_{i,j=1}^{N_{\text{dom}}} L^2(\Sigma_{ij})$ defined by

$$g^{(n+1)} = Ag^{(n)} + b,$$

(1.3)

where $g^{(n)}$: set of boundary data $(g^{(n)}_{ij})_{1 \leq i,j \leq N_{\text{dom}}}$, $b$ given by the Dirichlet boundary condition. Hence, (1.1)-(1.2) can be seen as an iteration of the Jacobi method (or fixed point iteration) applied to: $(I - A)g = b$, where $I$ is the identity operator.

Krylov solvers version

Following this idea, any Krylov solver could be applied: GMRES vs. successive approximations procedure.
Introduction to DDM

Transmission boundary operators

Numerical examples

Improvement in GetDP

Conclusion and perspectives
Transmission boundary operators

What is the impact of correctly choosing $S$?
The choice of $S$ directly impacts the spectral properties of the operator $A$ and may lead to convergence breakdown (e.g. eigenvalues with modulus larger than 1 for Jacobi methods). On the other side, a suitable choice of $S$ would lead to fast converging algorithms (e.g. large clusters of eigenvalues for GMRES)

Després (Ph.D. 1991):
Sommerfeld transmission boundary condition (IBC(0))

\[ S^0 u = -iku. \]  (2.1)
Transmission boundary operators

Improved operators

- **OO0 and OO2 (Optimized Order 0 and 2):** Gander, Magoulès and Nataf (2002)

  \[ S^{OO0}u = au \quad \text{and} \quad S^{OO2}u = au - b\Delta_\Sigma u, \]  
  \[ (2.2) \]

  complex numbers \( a \) and \( b \) obtained by solving a min-max optimization problem, \( \Delta_\Sigma \) is the Laplace-Beltrami operator on the interface \( \Sigma \).

- **Evanescent Modes Damping Algorithm (EMDA):** Bendali & Boubendir (2008) (IBC(\( \mathcal{X} \)))

  \[ S^\mathcal{X}u = -iku + \mathcal{X}u, \]  
  \[ (2.3) \]

  \( \mathcal{X} \): self-adjoint positive operator (usually a real-valued positive coefficient).
Transmission boundary operators

Antoine, Geuzaine, Boubendir (2011)

Generalized Impedance Boundary Condition - GIIBC(sq,\(\varepsilon\)):

\[
S^{sq,\varepsilon} u = -ik \sqrt{1 + \text{div}_{\Sigma} \left( \frac{1}{k_{\varepsilon}^2} \nabla_{\Sigma} \right)} u,
\]  

(2.4)

where

\[
k_{\varepsilon} = k + i\varepsilon,
\]

(2.5)

is a complexified wavenumber and where...

- \text{div}_{\Sigma}: surface divergence of a tangent vector field on \(\Sigma\)
- \(\nabla_{\Sigma}\): tangential gradient of a surface field
- The square-root \(\sqrt{A}\) of an operator \(A\) is classically defined through the spectral decomposition of \(A\)
- \(\sqrt{z}\) designates the principal determination of the square-root of a complex number \(z\) with branch-cut along the negative real axis
Transmission boundary operators

Formal construction of the square-root operator

Consider the half-space, with straight interface $\Sigma$:

\[
\begin{cases}
\Delta u + k^2 u = 0, & \text{in } \mathbb{R}^3_+ = \{ x \in \mathbb{R}^3; x_1 > 0 \}, \\
u = g, & \text{on } \Sigma, \\
u \text{ is outgoing},
\end{cases}
\] (2.6)

With Fourier variable $\xi$ along $\Sigma$, we have

\[
\partial_n u(0, x_\perp) = \mathcal{F}_\xi^{-1}(\sigma_1(k, \xi) \hat{u}(x_1, \xi))|_{\Sigma}.
\] (2.7)

with $\sigma_1(k, \xi) = ik\sqrt{1 - |\xi|^2/k^2}$. In terms of pseudodifferential operators, another way of writing this equation is

\[
\partial_n u(0, x_\perp) = \Lambda(u(0, x_\perp)) = \text{Op}(\sigma_1)u(0, x_\perp), \quad \text{on } \Sigma,
\] (2.8)

with $\Lambda := \text{Op}(\sigma_1)$ the pseudodifferential operator with symbol $\sigma_1$. 
Transmission boundary operators

Formal construction of the square-root operator

Since the Helmholtz equation (2.6) has constant coefficients

\[ \Lambda := \text{Op}(\sigma_1) = \text{Op} \left( ik \sqrt{1 - \frac{|\xi|^2}{k^2}} \right) = ik \sqrt{1 + \frac{\Delta \Sigma}{k^2}}, \quad (2.9) \]

The transmitting operator \( S \) is thus simply taken to be equal to

\[ S^{sq,0} u = -\Lambda u = -ik \sqrt{1 + \frac{\Delta \Sigma}{k^2}} u. \quad (2.10) \]

For curved \( \Sigma \), a regularization is needed for grazing rays \( |\xi| \approx k \) :

\( k \to k_\varepsilon \), leading to our definition of \( S^{sq,\varepsilon} \).
Transmission boundary operators

Special features

1. The nonlocal operator $S_{\text{sq}, \varepsilon}^{s}$ can be accurately localized using complex Padé approximants

$$\sqrt{1 + z} \approx R_{N_p}^{\alpha}(z) = C_0 + \sum_{\ell=1}^{N_p} \frac{A_{\ell} z}{1 + B_{\ell} z},$$

and suitably combined with finite element methods

2. The convergence of the resulting DDM is quasi-optimal: this means that you would consider a mode expansion of the solution, the convergence is almost optimal for both the evanescent, propagating and grazing modes: see next slides.

3. This results in an effective solution where the iterative procedure is quasi independent of both the wavenumber and spatial resolution (mesh refinement).
Transmission boundary operators

Convergence analysis for a model problem

Figure: Model problem with two subdomains and a circular interface.

Study the iteration operator $\mathcal{A} = \sum_{m=-\infty}^{+\infty} A_m e^{im\theta}$ in terms of the spectrum of the modal matrix $A_m$. 
Transmission boundary operators

(a) Spectral radius of the modal iteration operator $A_m$ vs. the mode $m$.

(b) Spectral radius of the iteration operator $A$ vs. $\chi$ (for IBC), $\Delta k$ (for OO2) and the damping parameter $\varepsilon$ (for GIBC).
Transmission boundary operators

Figure: Eigenvalues distribution in the complex plane for \((I - A)\) and different transmission operators.
Transmission boundary operators

Figure: Eigenvalue distribution in the complex plane for the exact and Padé-localized square-root transmission operator of order 4 (left) and 8 (right).
Transmission boundary operators

Practical considerations

1. A complete description of the finite element approximation with localization by complex Padé approximant is detailed in our paper JCP2011

2. In particular, we show that the implementation is easy in a basic finite element solver and does not require any new development (similar to OO2). More or less, all is done during the assembly process (see next slide)

3. Finally, the numerical solution for each iteration does not require any significant extra computational cost compared with OO2
Transmission boundary operators

An idea of the (small size) FEM systems for the local subproblems

\[
\begin{cases}
(S_{\Omega_h} - k^2M_{\Omega_h} - ikC_0M_{\partial\Omega_h})u + ik \sum_{\ell=1}^{N_p} A_{\ell} S_{k\varepsilon}^{-2} \varphi_\ell = -M_{\partial\Omega_h} g_{in}, \\
-M_{\partial\Omega_h} u - (B_{\ell} S_{k\varepsilon}^{-2} - M_{\partial\Omega_h}) \varphi_\ell = 0, \quad \ell = 1, \ldots, N_p.
\end{cases}
\]

\(S_{\Omega_h}\) and \(M_{\Omega_h}\): stiffness/mass matrices for linear elements for \(\Omega_h\).
\(S_{\partial\Omega_h}\) and \(M_{\partial\Omega_h}\): stiffness/mass matrices for the transmitting surface \(\partial\Omega_h\).
\(S_\beta\): generalized stiffness matrix for a surface function \(\beta\).
\(u \in \mathbb{C}^{n_v}\): local unknown vector.
\(\varphi_\ell \in \mathbb{C}^{n_\ell}\): local surface unknown auxiliary vectors.
Introduction to DDM

Transmission boundary operators

Numerical examples

Improvement in GetDP

Conclusion and perspectives
Numerical examples

Parameters

- In all the calculations, the parameters of the square-root transmission boundary operator are fixed (interesting point: no numerical optimization procedure).
- For the other approaches, we consider the optimized parameters in the reference papers (sometimes unclear if it is really optimal for complex configurations).
Numerical examples

(a) “circle-concentric” DD    
(b) “circle-pie” DD

Figure: Two-dimensional test cases: reconstruction of the scattered field on the global domain after a DDM computation with $k = 4\pi$ and $N_{\text{dom}} = 5$. 
### Numerical examples

<table>
<thead>
<tr>
<th>$N_{\text{dom}}$</th>
<th>Jacobi</th>
<th>GMRES</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>12</td>
<td>9</td>
</tr>
<tr>
<td>5</td>
<td>43</td>
<td>25</td>
</tr>
<tr>
<td>10</td>
<td>140</td>
<td>52</td>
</tr>
<tr>
<td>15</td>
<td>281</td>
<td>80</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$k$</th>
<th>Jacobi</th>
<th>GMRES</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\pi$</td>
<td>96</td>
<td>41</td>
</tr>
<tr>
<td>2 $\pi$</td>
<td>64</td>
<td>36</td>
</tr>
<tr>
<td>3 $\pi$</td>
<td>52</td>
<td>39</td>
</tr>
<tr>
<td>4 $\pi$</td>
<td>60</td>
<td>38</td>
</tr>
</tbody>
</table>

**Table:** Number of iterations vs. number of subdomains for $k = \pi$ (left) and Number of iterations vs. wavenumber for $N_{\text{dom}} = 8$ (right) when using the Jacobi or GMRES algorithm, for the “circle-concentric” decomposition.
**Numerical examples**

**Figure:** Convergence for the “circle-concentric” decomposition. Number of iterations vs. wavenumber.
Numerical examples

Figure: Convergence for the “circle-concentric” decomposition. Number of iterations vs. mesh density.
Numerical examples

![Graph showing convergence for the “circle-concentric” decomposition. Number of iterations vs. number of subdomains.](image)

**Figure:** Convergence for the “circle-concentric” decomposition. Number of iterations vs. number of subdomains.
Numerical examples

Figure: Convergence for the “circle-pie” decomposition. Number of iterations vs. wavenumber.
Numerical examples

Figure: Convergence for the “circle-pie” decomposition. Number of iterations vs. mesh density.
Numerical examples

**Figure:** Convergence for the “circle-pie” decomposition. Number of iterations vs. number of subdomains.
Numerical examples

**Figure:** Submarine problem with 5 subdomains: iso-surfaces of the real part of the scattered field for $k = 14\pi$. 
Numerical examples

submarine, $N_{\text{dom}} = 5, n_\lambda = 10$

**Figure:** Convergence of the GMRES DDM solvers for the submarine problem.
Introduction to DDM

Transmission boundary operators

Numerical examples

Improvement in GetDP

Conclusion and perspectives
Why an improvement . . . ?

The code is implemented using Matlab together with GMSH/GetDP:

\[\text{While}(!\text{gmres\_convergence}) \quad // \text{ launched by matlab}\]
\begin{itemize}
  \item For \(idom = 1, \ldots, N_{\text{dom}}\)
    \begin{itemize}
    \item Launch GetDP to...
      \begin{itemize}
      \item Generates the matrix of the linear system
      \item Computes the LU factorization of the matrix
      \item Solves the system
      \end{itemize}
    \end{itemize}
  \item Close GetDP // (and erase everything from the memory)
\end{itemize}
\item EndFor
\item EndWhile

**Problem:** Redundant operations from one iteration to the other (system generation and LU factorization).

**Solution:** Let GetDP launch the Krylov solver.
New GetDP function: IterativeLinearSolver

Define the unknown

- GetDP creates a .pos file (Post Processing)
- GMSH (linked with GetDP) reads the .pos file and creates a “View”
- This view is used as the unknown of the Krylov solver

New GetDP operation*

IterativeLinearSolver[KSP, tol, maxit, restart, List_of_View] { 
    Matrix-vector product (ex.: Generate[A]; SolveAgain[A]; . . . ) 
}

- List_of_View: indices of the views (=the unknown) (here $g_{ij}$ on the transmission boundaries).
- Matrix-vector product: matrix-free vector product (here: solving Helmholtz on each subdomain + compute $g_{ji}$)
- KSP: one of the Krylov solver available in PETSc (gmres, cg, . . . )

*but available only on my computer for the moment... sorry :-)

Operation {
  ...
  IterativeLinearSolver["gmres", 1e-6, 500, 500, {0:N_DOM-1}]
  {
    // solve Helmholtz locally
    For idom In {0:N_DOM-1}
      //compute u on Omega_i
      Generate[Helmholtz~{idom}]; SolveAgain[Helmholtz~{idom}];
      SaveSolution[Helmholtz~{idom}];
      //update g_out
      Generate[ComputeG~{idom}]; SolveAgain[ComputeG~{idom}];
      SaveSolution[ComputeG~{idom}];
      //write g_out
      PostOperation[g_out~{idom}];
    EndFor
    //clear the View in memory
    GmshClearAll;
    //read the new g_in
    For idom In {0:N_DOM-1}
      GmshRead[Sprintf("g_out_%g.pos", idom)];
    EndFor
  } //end IterativeLinearSolver
  ...
} //end Operation

Submarine (Helmholtz equation)

Parameters

- characteristic length: \( L = 1 \)
- \( n_\lambda = \lambda/10 \)
- \( N_{\text{dom}} = 5 \)

Result

<table>
<thead>
<tr>
<th>( k )</th>
<th>Iter.</th>
<th>Size</th>
<th>Old</th>
<th>Fast Solve</th>
</tr>
</thead>
<tbody>
<tr>
<td>( k = 12\pi )</td>
<td>37</td>
<td>22908</td>
<td>6m24s</td>
<td>3m33s</td>
</tr>
<tr>
<td>( k = 14\pi )</td>
<td>31</td>
<td>29700</td>
<td>10m42s</td>
<td>5m00s</td>
</tr>
<tr>
<td>( k = 16\pi )</td>
<td>32</td>
<td>39396</td>
<td>19m04s</td>
<td>7m33s</td>
</tr>
</tbody>
</table>

Table: Comparison of the CpuTime between the old and the new code (re-use the LU factorization)
Maxwell 3D - “sphere concentric”

Perfectly conductor (interior sphere) with boundary \( \Gamma \).

2D Mesh of the geometry

Electric field:

\[
\begin{align*}
\text{curl} \, \text{curl} E - ik^2 E &= 0 & \text{in } \Omega_i \\
E \times n &= \mathbf{E}^{inc} \times n & \text{on } \Gamma_i \\
\text{curl} E - ikE &= 0 & \text{on } \Gamma^\infty_i \\
SE &= g_{in} & \text{on } \Sigma_i
\end{align*}
\]

\[g_{out} = -g_{in} + 2SE\]

on \( \Sigma_i \).

Magnetic field:
\[H = \frac{1}{ik} \text{curl} E.\]

Surface current:
\[j = n \times (H + H^{inc}).\]

Silver-Müller transmission condition (order 0)

\[SE = S^0 E = \text{curl} E - ikE.\]
Maxwell 3D - “sphere concentric”

\[ N_{\text{dom}} = 3, \quad n_\lambda = \lambda/10, \quad R_{\text{INT}} = 1, \quad R_{\text{EXT}} = 4, \quad \text{tol} = 10^{-6}. \]

Mesh (displayed = 2D)  
Surface current \( j \) on \( \Gamma \)

<table>
<thead>
<tr>
<th></th>
<th>Iter.</th>
<th>Size</th>
<th>“Old”</th>
<th>New</th>
</tr>
</thead>
<tbody>
<tr>
<td>( k = 1 )</td>
<td>89</td>
<td>11664</td>
<td>1m04s</td>
<td>0m34s</td>
</tr>
<tr>
<td>( k = 2 )</td>
<td>92</td>
<td>44676</td>
<td>7m36s</td>
<td>2m26s</td>
</tr>
<tr>
<td>( k = 3 )</td>
<td>92</td>
<td>97452</td>
<td>47m52s</td>
<td>6m40s</td>
</tr>
</tbody>
</table>

**Table:** Comparison of the CpuTime between old and the new code (fast solving + fast generation of the system).
Introduction to DDM

Transmission boundary operators

Numerical examples

Improvement in GetDP

Conclusion and perspectives
Conclusion and perspectives

Conclusion

- New robust transmission boundary condition for DDM
- Easy-to-implement in a basic FEM solver
- Quasi-optimal convergence of the resulting DDM

Perspectives (currently under progress 2011-2012)

- What happens for mesh partitions with non-smooth interfaces?
- Coupling with Phase Reduction techniques (Antoine & Geuzaine JCP09) for high frequency waves with much less degrees of freedom
- Extension to Maxwell’s equations
Conclusion and perspectives

Code optimization

**In a near future**
- Allow the user to define a preconditioner
  \[ M^{-1}AX = M^{-1}b \]
- Implement a Jacobi method (does not exist in PETSc)

**In a not so near but no so far future**
- Take care of the “cross-point” (points that belong to more than 2 subdomains) (A. Bendali & Y. Boubendir, 2006)
- Improve parallelization
Mixed formulation
Mixed formulation

Let $v_i = -(ik)^{-1} \nabla u_i$.

\[ \Delta u_i + k^2 u_i = 0 \iff \Delta u_i - (ik)^2 u_i = 0 \iff \begin{cases} v_i = -(ik)^{-1} \nabla u_i, \\ \text{div}(v_i) + (ik) u_i = 0. \end{cases} \]

\[ \Rightarrow \begin{cases} u_i \in L^2(\Omega_i) \\ v_i \in H_{\text{div}}(\Omega_i) = \left\{ w \in (L^2(\Omega_i))^3 \text{ such that } \text{div}(w) \in L^2(\Omega_i) \right\}. \end{cases} \]

Mixed problem

\[ \begin{cases} \text{div}(v_i) + (ik) u_i = 0 & \text{in } \Omega_i, \\ v_i = -(ik)^{-1} \nabla u_i & \text{in } \Omega_i, \\ u_i = -u^{inc} & \text{on } \Gamma_i, \\ -ik(v_i \cdot n_i) + B u_i = 0 & \text{on } \Gamma^\infty_i, \\ -ik(v_i \cdot n_i) + S^{\text{sq},\varepsilon} u_i = g_{\text{in}} & \text{on } \Sigma_i, \end{cases} \]
Weak formulation (mixed case)

“Standard” square-root transmission condition

\[
\begin{cases}
\partial_n u - ikC_0 u_i - ik \sum_{\ell=1}^{N_p} A_\ell \text{div}_{\Sigma_i} \left( \frac{1}{k^2} \nabla \Sigma_i \varphi_\ell \right) - g_{in} = 0, \\
-u_i - B_\ell \text{div}_{\Sigma_i} \left( \frac{1}{k^2} \nabla \Sigma_i \varphi_\ell \right) + \varphi_\ell = 0, \quad \forall \ell = 1, \ldots, N.
\end{cases}
\]

Mixed square-root transmission condition

\[
\begin{cases}
-ik(v_i \cdot n) - ikC_0 u_i - ik \sum_{\ell=1}^{N_p} A_\ell \text{div}_{\Sigma_i} \left( \frac{1}{k^2} \Phi_\ell \right) - g_{in} = 0, \\
-u_i - B_\ell \text{div}_{\Sigma_i} \left( \frac{1}{k^2} \Phi_\ell \right) + \varphi_\ell = 0, \quad \forall \ell = 1, \ldots, N, \\
\Phi_\ell = \nabla \Sigma_i \varphi_\ell, \quad \forall \ell = 1, \ldots, N,
\end{cases}
\]

with \( \Phi_\ell \in H_{\text{div}}(\Sigma_i) \) and \( \varphi_\ell \in L^2(\Sigma_i) \).
Mixed weak formulation

Find \( u_i \in L^2(\Omega_i) \) and \( v_i \in H_{\text{div}}(\Omega_i) \) such that

\[
\forall u_i' \in L^2(\Omega_i), \quad \int_{\Omega_i} \text{div}(v_i) u_i' d\Omega_i + \int_{\Omega_i} i k u_i u_i' d\Omega_i = 0,
\]

\[
\forall v_i' \in H_{\text{div}}(\Omega_i),
\]

\[
\int_{\Omega_i} i k (v_i \cdot v_i') d\Omega_i - \int_{\Omega_i} u_i \cdot \text{div}(v_i') d\Omega_i - \int_{\Gamma_i} u_{\text{inc}} (v_i' \cdot n_i) d\Gamma_i + \int_{\Gamma_i^\infty} u_i (v_i' \cdot n_i) d\Gamma_i^\infty 
\]

\[
- \int_{\Sigma_i} \frac{1}{i k C_0} g_{\text{inc}} (v_i' \cdot n_i) d\Sigma_i - \int_{\Sigma_i} \frac{1}{C_0} (v_i \cdot n_i) (v_i' \cdot n_i) d\Sigma_i 
\]

\[
- \sum_{\ell=1}^{N_p} \int_{\Sigma_i} \frac{A_\ell}{C_0} \text{div}_{\Sigma_i} \left( \frac{1}{k_E^2} \Phi_\ell \right) (v_i' \cdot n_i) d\Sigma_i = 0,
\]

\[
\forall \ell = 1, \ldots, N_p, \text{ find } (\varphi_\ell) \in L^2(\Sigma_i) \text{ and } (\Phi_\ell) \in H_{\text{div}}(\Sigma_i), \text{ such that}
\]

\[
\forall \varphi_\ell' \in L^2(\Sigma_i), \quad \int_{\Sigma_i} -u_i \varphi_\ell' d\Sigma_i - \int_{\Sigma_i} B_\ell \cdot \text{div} \left( \frac{1}{k_E^2} \Phi_\ell \right) \varphi_\ell' d\Sigma_i + \int_{\Sigma_i} \varphi_\ell \varphi_\ell' d\Sigma_i = 0,
\]

\[
\forall \Phi_\ell' \in H_{\text{div}}(\Sigma_i), \quad \int_{\Sigma_i} \Phi_\ell \Phi_\ell' d\Sigma_i + \int_{\Sigma_i} \varphi_\ell \cdot \text{div}(\Phi_\ell') d\Sigma_i = 0.
\]