Two-Scale Wave Equation Modeling for Seismic Inversion

Susan E. Minkoff

Department of Mathematics and Statistics
University of Maryland Baltimore County
Baltimore, MD 21250, USA

RICAM Workshop 3: Wave Propagation and Scattering, Inverse Problems and Applications in Energy and the Environment
November 21, 2011

Thanks to: Tanya Vdovina, Oksana Korostyshevskaya, Sean Griffith, and Bill Symes
I. Motivation: A Seismic Inversion Example

II. Operator Upscaling for the Acoustic Wave Equation
   A. Parallel Cost and Timing Studies
   B. Numerical Examples (Accuracy)
   C. Numerical Experiments (Convergence)
   D. Matrix Analysis To Illustrate Physics of Solution
   E. Solution of the Inverse Problem

III. Conclusions
To illuminate a section of the subsurface, geophysicists introduce energy into the ground.

Seismic source is always a part of the resulting data.

Good estimate of the energy source essential to recovery of mechanical earth parameters.

Source’s shape (signature) and direction-dependence (radiation pattern) are of no use in themselves.

Seismic inverse problems generally do not have unique solutions. The model estimate may have unique average behavior.
Inputs to Numerical Experiments:

- Gulf of Mexico Data from Exxon Production Research Co.
- Eleven common-midpoint data gathers.
- Radon transform to yield 48 plane-wave traces (per gather) with slowness values from $p_{\text{min}} = 0.1158$ ms/m to $p_{\text{max}} = 0.36468$ ms/m.
- Estimate of anisotropic air gun source in a 31-component Legendre expansion in slowness, $p$.
- Viscoelastic model (with estimated attenuation coefficient) used as forward simulator for inversion.
Stacked Section of Gulf of Mexico Data
Seismic data in $\tau - p$ domain

**Figure:** The $\tau - p$ transformed seismic data from common midpoint gather 6. Data was filtered by convolving it with a 15 Hz Ricker filter.
Inversion Algorithm:

1. Estimate $P$-wave background velocity and elastic parameter reflectivities by *Differential Semblance Optimization*

   $$ J_{DSO}[v, r] = \frac{1}{2} \left\{ \| S[v, r] - S_{data} \|^2 + \lambda^2 \| W r \|^2 \right\} $$

   $$ + \sigma^2 \| \partial r / \partial p \|^2 \} $$

2. Estimate seismic source and elastic parameter reflectivities (again) by alternation and *Output Least Squares Inversion*

   $$ J_{OLS}[r, f] = \frac{1}{2} \left\{ \| S[r, f] - S_{data} \|^2 + \lambda^2 \| W[r, f] \|^2 \right\} $$
Alternation Algorithm:

Repeat until convergence:

1. Given the current source, $f_c$, and current reflectivity, $r_c$, invert for a new estimate of the reflectivity $r_+$ using Output Least Squares (i.e., \(J_{OLS}[r, f] = \frac{1}{2} \|d^{\text{pred}}[r, f] - d^{\text{obs}}\|^2\))

2. Replace $r_c$ by $r_+$.

3. Given the current source and reflectivity guesses, $f_c$, $r_c$, invert for a new estimate of the source $f_+$ using OLS.

4. Replace $f_c$ by $f_+$. 
Two Experiments:

1. Model $m = \text{three elastic parameter reflectivities. (Source not updated in inversion)}$

2. Model $m = \text{three elastic parameter reflectivities and seismic source.}$

In both cases the data is the same. The fixed background velocity is the same. The starting guesses for the reflectivities are the same (zero). The algorithmic stopping tolerances are the same.
Figure: Left: air gun model source; Right: anisotropic source from inversion
Results:

Inversion-estimated source appears to allow for a better corresponding reflectivity estimate:

- **Data Fit:**
  1. Experiment 1 (air gun source)
     - 55% rms error.
  2. Experiment 2 (inversion-estimated source)
     - 27% rms error.
  3. In fact in Experiment 2 (inversion-estimated source)
     - 10% rms error in region around gas sand target.

- **Model Fit:** Experiment 2 reflectivity estimates match well log better than Experiment 1 estimates.
Data Misfit Comparison for Two experiments

Figure: Left: 55% data misfit. Right: 29% data misfit
Well log comparisons for P-wave impedance

Figure: Left: air gun model source  Right: inversion estimated source

The solid line shows the inversion result. The dashed line shows the detrended well log.


Problem: Running inversion experiments is expensive...
One would like to simulate wave propagation on a coarser scale than the one on which the parameters are defined.

We adapted the subgrid upscaling technique developed for elliptic problems (flow in porous media) to the wave equation.

Goal is to be able to solve the problem on the coarser grid while still capturing some of the small scale features internal to coarse grid blocks.


Advantages: No Scale Separation or Periodic Medium Requirements.
Model problem

The Acoustic Wave Equation

\[ \frac{1}{\rho c^2} \frac{\partial^2 p}{\partial t^2} - \nabla \cdot \left( \frac{1}{\rho} \nabla p \right) = f \]

\( p \) is the pressure

\( c(x, y) \) is the sound velocity,

\( \rho(x, y) \) is the density,

\( f \) is the source of acoustic energy

The First Order System

\[ \begin{cases} \vec{v} = -\frac{1}{\rho} \nabla p & \text{in } \Omega, \\ \frac{1}{\rho c^2} \frac{\partial^2 p}{\partial t^2} = -\nabla \cdot \vec{v} + f & \text{in } \Omega, \end{cases} \]

Boundary conditions

\( \vec{v} \cdot \nu = 0, \text{ on } \partial\Omega \)
The finite element method

Find $\vec{v} \in V$ and $p \in W$ such that

\[
\begin{align*}
\langle \rho \vec{v}, \vec{u} \rangle &= \langle p, \nabla \cdot \vec{u} \rangle, \\
\left\langle \frac{1}{\rho c^2} \frac{\partial^2 p}{\partial t^2}, w \right\rangle &= -\langle \nabla \cdot \vec{v}, w \rangle + \langle f, w \rangle
\end{align*}
\]

for all $\vec{u} \in V$ and $w \in W$

- $W = \{\text{piecewise discontinuous constant functions}\}$
- $V = \{\text{piecewise continuous linear functions of the form } (a_1 x + b_1, a_2 y + b_2)\}$

Full fine grid

- Pressure
- Acceleration
**Goal:** Capture the fine scale behavior on the coarser grid

**Idea:** Decompose the solution

\[ \vec{v} = \vec{v}^c + \delta \vec{v} \]

- \( \vec{v}^c \in V^c \) is the coarse-scale solution
- \( \delta \vec{v} \in \delta V \) are the fine-grid unknowns internal to each coarse-grid cell – subgrid unknowns
Simplifying assumption: \( \delta \vec{V} \cdot \nu = 0 \) on the boundary of each coarse cell.

This assumption allows us to decouple the subgrid problems from coarse-grid block to coarse-grid block.

We use full fine-grid pressure.
Two steps of upscaling

- **Step 1:** On each coarse element solve the subgrid problem:
  - Find $\delta \vec{v} \in \delta V$ and $p \in W$ such that
    \[
    \begin{align*}
    \langle \rho (\vec{v}^c + \delta \vec{v}), \delta \vec{u} \rangle &= \langle p, \nabla \cdot \delta \vec{u} \rangle, \\
    \left\langle \frac{1}{\rho c^2} \frac{\partial^2 p}{\partial t^2}, w \right\rangle &= -\langle \nabla \cdot (\vec{v}^c + \delta \vec{v}), w \rangle + \langle f, w \rangle
    \end{align*}
    \]
    for all $\delta \vec{u} \in \delta V$ and $w \in W$
  - **Note:** the value of $\vec{v}^c$ is unknown at this stage.
  - The subgrid problem fully determines $p$.

- **Step 2:** Use the solution of subgrid problem to solve the coarse problem:
  - Find $\vec{v}^c \in V^c$ such that
    \[
    \langle \rho (\vec{v}^c + \delta \vec{v}), \vec{u}^c \rangle = \langle p, \nabla \cdot \vec{u}^c \rangle, \quad \text{for all} \quad \vec{u}^c \in V^c
    \]
Parallel implementation

**Serial cost**

\[(N_p + 2N_u)N_\delta N_c + O(N_c)\]

**Parallel cost**

\[\frac{(N_p + 2N_u)N_\delta N_c}{p} + O(N_c)\]

\(N_p\) and \(N_u\) are the number of flops required to solve for pressure and acceleration respectively, \(N_\delta\) and \(N_c\) are the sizes of the subgrid and coarse problems, \(p\) is the number of processors.

- **Subgrid problems:** Embarrassingly parallel.
  - No communication between processors.
  - No ghost-cell memory allocation required.

- **Coarse problem:** Solve in serial.

- **Post-processing:** Cheap and fast.
Timing Studies

- Numerical grid contains $3600 \times 3600$ fine grid blocks and $36 \times 36$ coarse grid blocks.
- Timings are for 20 time steps.

<table>
<thead>
<tr>
<th>number of processors</th>
<th>total time</th>
<th>subgrid problems</th>
<th>coarse problem</th>
<th>post-processing</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>29.70</td>
<td>29.69</td>
<td>0.00060</td>
<td>0.0026</td>
</tr>
<tr>
<td>2</td>
<td>15.46</td>
<td>15.38</td>
<td>0.00045</td>
<td>0.0711</td>
</tr>
<tr>
<td>4</td>
<td>7.63</td>
<td>7.56</td>
<td>0.00049</td>
<td>0.0707</td>
</tr>
<tr>
<td>6</td>
<td>5.23</td>
<td>5.14</td>
<td>0.00048</td>
<td>0.0749</td>
</tr>
<tr>
<td>8</td>
<td>4.37</td>
<td>4.26</td>
<td>0.00046</td>
<td>0.0896</td>
</tr>
<tr>
<td>12</td>
<td>3.07</td>
<td>2.94</td>
<td>0.00045</td>
<td>0.1150</td>
</tr>
</tbody>
</table>


Timing Studies:

<table>
<thead>
<tr>
<th>number of processors</th>
<th>Finite-difference code</th>
<th>Upscaling code number of fine blocks per coarse block</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>100 × 100</td>
<td>100 × 100 60 × 60 50 × 50 40 × 40</td>
</tr>
<tr>
<td>1</td>
<td>29.43</td>
<td>29.69 29.92 30.31 31.01</td>
</tr>
<tr>
<td>2</td>
<td>—</td>
<td>15.46 15.35 15.61 16.13</td>
</tr>
<tr>
<td>4</td>
<td>—</td>
<td>7.63 7.97 8.59 8.84</td>
</tr>
<tr>
<td>6</td>
<td>—</td>
<td>5.23 5.62 5.88 6.37</td>
</tr>
<tr>
<td>8</td>
<td>—</td>
<td>4.37 5.12 7.52 7.20</td>
</tr>
<tr>
<td>12</td>
<td>—</td>
<td>3.07 3.82 3.97 4.43</td>
</tr>
</tbody>
</table>

Note: Performance improves as coarse block size increases.
Numerical Experiment I (Acoustic Wave Equation)

- Domain is of size $10 \times 10$ km.
- Fine grid: $1000 \times 1000$. Coarse grid: $100 \times 100$.
- Gaussian source, 200 time steps.
- Sound velocity strips range from 3500 m/s to 7500 m/s.
Vertical Acceleration

Full finite-difference soln

Reconstructed Upscaled Soln
Numerical Experiment II (Elastic Wave Equation)

- Fine grid: $320 \times 320 \times 320$. Coarse grid: $80 \times 80 \times 80$.
- Source is Gaussian in space, Ricker in time.
- Wave propagates about 7 wavelengths.
- Mixture of two materials with $V_p = 2.5$ km/s or 3 km/s; $V_s = 1.5$ and 1.75 km/s respectively, $\rho = 2.2 \cdot 10^{12}$ and $2.3 \cdot 10^{12}$ kg/km$^3$. 

![Image of a grid with color scale representing wave propagation](image_url)
yz slices of first component of velocity solution

Full finite-element soln

Augmented Upscaled Soln

xy slices of first component of velocity solution

Full finite-element soln

Augmented Upscaled Soln
Comparison of time traces for the first component of velocity

Red curve = full finite element solution. Green curve = coarse solution. Blue curve = full finite element solution for the homogeneous medium with a single (average) value for each of the three input parameter.

Receiver at (16 km, 8 km, 12 km)  Receiver at (16 km, 8 km, 18 km)
Order of Accuracy Experiment 1

- Number of coarse grid blocks is fixed at 100 in \(x\) and \(y\). **Number of fine blocks per coarse block changes** in each experiment.

<table>
<thead>
<tr>
<th>Number of fine blocks</th>
<th>Number of coarse blocks</th>
<th>Time step (s)</th>
<th>Number of time steps</th>
<th>(|P - p|_0 / |p|_1)</th>
<th>(|U - u|_0 / |u|_1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>400 (\times) 400</td>
<td>100 (\times) 100</td>
<td>1.77 (\cdot) 10(^{-4})</td>
<td>200</td>
<td>5.48 (\cdot) 10(^{-3})</td>
<td>1.28 (\cdot) 10(^{-2})</td>
</tr>
<tr>
<td>800 (\times) 800</td>
<td>100 (\times) 100</td>
<td>8.83 (\cdot) 10(^{-5})</td>
<td>400</td>
<td>2.75 (\cdot) 10(^{-3})</td>
<td>1.11 (\cdot) 10(^{-2})</td>
</tr>
<tr>
<td>1600 (\times) 1600</td>
<td>100 (\times) 100</td>
<td>4.42 (\cdot) 10(^{-5})</td>
<td>800</td>
<td>1.40 (\cdot) 10(^{-3})</td>
<td>1.06 (\cdot) 10(^{-2})</td>
</tr>
<tr>
<td>3200 (\times) 3200</td>
<td>100 (\times) 100</td>
<td>2.21 (\cdot) 10(^{-5})</td>
<td>1600</td>
<td>7.27 (\cdot) 10(^{-4})</td>
<td>1.05 (\cdot) 10(^{-2})</td>
</tr>
<tr>
<td>6400 (\times) 6400</td>
<td>100 (\times) 100</td>
<td>1.10 (\cdot) 10(^{-5})</td>
<td>3200</td>
<td>3.97 (\cdot) 10(^{-4})</td>
<td>1.04 (\cdot) 10(^{-2})</td>
</tr>
</tbody>
</table>

- Linear convergence in pressure \((h)\).

Number of fine grid blocks is fixed at 6400 in $x$ and $y$. **Number of coarse blocks varies in each experiment.**

<table>
<thead>
<tr>
<th>Number of fine blocks</th>
<th>Number of coarse blocks</th>
<th>$| P - p |_0 / | p |_1$</th>
<th>$| U - u |_0 / | u |_1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>6400 × 6400</td>
<td>100 × 100</td>
<td>3.97 · 10$^{-4}$</td>
<td>1.04 · 10$^{-2}$</td>
</tr>
<tr>
<td>6400 × 6400</td>
<td>200 × 200</td>
<td>3.50 · 10$^{-4}$</td>
<td>5.24 · 10$^{-3}$</td>
</tr>
<tr>
<td>6400 × 6400</td>
<td>400 × 400</td>
<td>3.41 · 10$^{-4}$</td>
<td>2.65 · 10$^{-3}$</td>
</tr>
<tr>
<td>6400 × 6400</td>
<td>800 × 800</td>
<td>3.39 · 10$^{-4}$</td>
<td>1.38 · 10$^{-3}$</td>
</tr>
<tr>
<td>6400 × 6400</td>
<td>1600 × 1600</td>
<td>3.39 · 10$^{-4}$</td>
<td>7.97 · 10$^{-4}$</td>
</tr>
</tbody>
</table>

**Linear convergence in acceleration ($H$).**
An Alternate Analysis:


Recall two steps of upscaling algorithm:

- **Subgrid problems**: On each coarse element solve the subgrid problem:
  
  Find $\delta \vec{v} \in \delta V$ and $p \in W$ such that
  
  $$
  \begin{cases}
  \langle \rho(\vec{v}^c + \delta \vec{v}), \delta \vec{u} \rangle = \langle p, \nabla \cdot \delta \vec{u} \rangle, \\
  \langle \frac{1}{\rho c^2} \frac{\partial^2 p}{\partial t^2}, w \rangle = -\langle \nabla \cdot (\vec{v}^c + \delta \vec{v}), w \rangle + \langle f, w \rangle
  \end{cases}
  $$

  for all $\delta \vec{u} \in \delta V$ and $w \in W$

- **Coarse problem**: Use the solution of subgrid problem to solve the coarse problem:

  Find $\vec{v}^c \in V^c$ such that

  $$
  \langle \rho(\vec{v}^c + \delta \vec{v}), \vec{u}^c \rangle = \langle p, \nabla \cdot \vec{u}^c \rangle, \quad \text{for all } \vec{u}^c \in V^c
  $$

Use finite element expansions to obtain the matrix-vector form

- **Subgrid problems**:

  $$
  A^{cf} \vec{v}_x^c + A^{ff} \delta \vec{v}_x = B^f p
  $$

  $$
  W \frac{\partial^2 p}{\partial t^2} = -(B^c)^T \vec{v}_x^c - (B^f)^T \delta \vec{v}_x - (v_y \text{ terms}) + F
  $$

- **Coarse problem**:

  $$
  A^{cc} \vec{v}_x^c + (A^{cf})^T \delta \vec{v}_x = B^c p
  $$

- Matrix entries

  $$
  a_{i,n}^{ff} = \langle \rho(\delta u_x)_n, (\delta u_x)_l \rangle \quad a_{i,i}^{ef} = \langle \rho(u_x^c)_i, (\delta u_x)_l \rangle \quad a_{i,i}^{cc} = \langle \rho(u_x^c)_i, (u_x^c)_l \rangle
  $$

  $$
  w_{l,m} = \langle \frac{1}{\rho c^2} w_m, w_l \rangle \quad b_{l,m}^f = \langle w_m, \frac{\partial (\delta u_x)_l}{\partial x} \rangle \quad b_{l,m}^c = \langle w_m, \frac{\partial (u_x^c)_l}{\partial x} \rangle
  $$

  $$
  f_l = \langle f, w_l \rangle
Matrix equation

- **Idea:** Eliminate the subgrid acceleration from the coarse equation
  
  - Subgrid problem
    
    \[ \delta v_x = -(A^{ff})^{-1} A^{cf} v_c^c + (A^{ff})^{-1} B^f p \]
  
  - Coarse problem
    
    \[ (A^{cc} - (A^{cf})^T (A^{ff})^{-1} A^{cf}) v_c^c = (B^c - (A^{cf})^T (A^{ff})^{-1} B^f ) p \]

  - Obtain a matrix equation for the coarse acceleration and pressure only
    
    \[ U v_c^c = D p \]

  - Find explicit formulas for the entries of \( U \) and \( D \) to derive a difference equation
Differential equation (from Taylor series expansion)

\[ \rho^{ups} v^c_x = -\frac{\partial p}{\partial x} \]

\( \rho^{ups} \) is the upscaled density given by the average of the density values on the boundary of the coarse cell.

Recall one of two original continuous pde's in our system: 
\[ \vec{v} = -\frac{1}{\rho} \nabla p \]
Assuming appropriate smoothness on pressure $p$ and density $\rho$, the upscaling algorithm solves the following differential equation for pressure inside a coarse block:

$$\frac{1}{\rho c^2} \frac{\partial^2 p}{\partial t^2} = \frac{\partial}{\partial x} (\rho^{-1} \frac{\partial p}{\partial x}) + \frac{\partial}{\partial y} (\rho^{-1} \frac{\partial p}{\partial y}) + f,$$

[order of approximation $O(h_x^2 + h_y^2)$].

Along coarse block edges upscaling satisfies the following equation for pressure:

$$\frac{1}{\rho c^2} \frac{\partial^2 p}{\partial t^2} = \frac{\partial}{\partial x} \left( (\rho^{ups})^{-1} \frac{\partial p}{\partial x} \right) + \left( (\rho^{ups})^{-1} \frac{\partial^2 p}{\partial x \partial y} \right) K + \frac{\partial}{\partial y} \left( (\rho^{ups})^{-1} \frac{\partial p}{\partial y} \right) + f,$$

[order of approximation $O(h_x + h_y)$] and $K$ is a constant which depends on the location of the pressure node within a single coarse cell.
Solving the Inverse Problem

Want to find $c$ given some observed $p$.

- Easier to find $m = \frac{1}{c^2}$, squared slowness.
- The least squares functional is given by

$$J(m) = \frac{1}{2} \sum_{s,r} \left[ \int_0^T (S_{s,r}p - d_{s,r})^2 + (T_{s,r}(\nabla p) - b_{s,r})^2 \, dt \right].$$

- $s$ and $r$ are indices over source and receiver position.
- $b_{s,r}$ and $d_{s,r}$ are observed values for $p$ and $\nabla p$.
- $S_{s,r}$ and $T_{s,r}$ are sampling operators, restricting the values of $p$ and $\nabla p$ to locations with observed values.

(Reference: Plessix, A Review of the Adjoint State Method For Computing the Gradient of a Functional with Geophysical Applications)
The Formal Optimization Problem

Find \( \min_m J(\phi, m) \) s.t. \( F(\phi, m) = 0 \).

- \( \phi \) is the state variable (pressure).
- \( m \) is the control variable (squared slowness).
- \( F(\phi, m) = 0 \) is the forward problem (the wave equation).

Method:

\[
\delta_m J = \delta\phi J \circ \delta_m \phi + \delta^e_m J.
\]

- \( \delta^e_m J \) is the explicit variation of \( J \) w.r.t. \( m \).

Problem: How do we find \( \delta_m \phi \)?

(Reference: Gunzburger’s *Perspectives in Flow Control and Optimization.*)
The Adjoint Problem

- The adjoint problem is:

\[ \frac{1}{c^2} \frac{\partial^2 \lambda}{\partial t^2} - \Delta \lambda = \sum_r S^*_{s,r} (S_{s,r} p - d_{s,r}) - \sum_r \nabla \cdot T^*_{s,r} (T_{s,r} (\nabla p) - b_{s,r}) \]

\[ \lambda \bigg|_{t=T} = 0 \]

\[ \frac{\partial \lambda}{\partial t} \bigg|_{t=T} = 0 \]

\[ (\nabla \lambda + T^*_{s,r} (T_{s,r} (\nabla p) - b_{s,r})) \cdot \nu = 0 \text{ on } \Gamma. \]

- The auxiliary conditions are:

\[ \mu_0 = m \frac{\partial \lambda}{\partial t} \bigg|_{t=0} \]

\[ \mu_1 = -m \lambda \bigg|_{t=0} \]

\[ \psi = -\lambda \text{ on } \Gamma. \]
Obtaining an Expression for the Gradient

Question: How do we get $D_m \mathcal{J}^{(n)} (x, y)$?

1. To get $D_m \mathcal{J}^{(n)} (x, y)$ from $\delta_m \mathcal{J}$, we let $\tilde{m}$ be $\delta (x, y)$.
2. This gives:

   $$ (D_m \mathcal{J})(x, y) = - \sum_s \left[ \int_0^T \frac{\partial^2 p(x, y)}{\partial t^2} \lambda (x, y) \, dt \right]. $$

3. To calculate the gradient, we must have access to both the forward and adjoint solutions at all time steps.
4. Use a checkpointing scheme to optimally determine what to store and what to recalculate.

(Reference: Gunzburger’s *Perspectives in Flow Control and Optimization.*)
An Optimization Algorithm

Beginning with $m^{(0)}(x, y)$, we repeat the following:

1. Solve the forward problem to get the state variables $p^{(n)}(x, y, t)$ and then solve the adjoint problem to get the adjoint variable $\lambda^{(n)}(x, y, t)$.

2. Calculate the gradient $D_m\mathcal{J}^{(n)}(x, y) = -\sum_s \left[ \int_0^T \frac{\partial^2 p(x, y)}{\partial t^2} \lambda(x, y) \, dt \right]$. 

3. Update squared slowness for the next step:

$$m^{(n+1)}(x, y) = m^{(n)}(x, y) + \beta_n \sum_s \left[ \int_0^T \frac{\partial^2 p(x, y)}{\partial t^2} \lambda(x, y) \, dt \right],$$

using some step size $\beta_n$. 

Introduce $\chi = -\nabla \lambda$ and decompose $\chi$ into $\delta\chi + \chi^c$.

Then, for each subgrid problem,

$$
\begin{align*}
(\chi^c + \delta\chi, \delta u) &= (\lambda, \nabla \cdot \delta u) \\
\left(\frac{1}{c^2} \frac{\partial^2 \lambda}{\partial t^2}, w\right) &= -\left(\nabla \cdot (\chi^c + \delta\chi), w\right) \\
&\quad + \left(\sum_r S_{s,r}^* (S_{s,r} p - d_{s,r}) - \sum_r \nabla \cdot T_{s,r}^* (T_{s,r} (\nabla p) - b_{s,r}), w\right)
\end{align*}
$$

for $w \in W$ and $\delta u \in \delta V$.

And, for each coarse problem,

$$
(\chi^c + \delta\chi, u^c) = (\lambda, \nabla \cdot u^c)
$$

for $u^c \in V^c$. 

Susan E. Minkoff (UMBC) 
Wave Equation Upscaling 
11/21/11 42 / 44
Conclusions

- To speed up an expensive iterative process like seismic inversion, **operator upscaling can be applied to the wave equation.**
- The upscaled solution captures some of the sub-wavelength heterogeneities of the full solution.
- For the acoustic implementation, the method exhibits good parallel performance due to the independence of the subgrid problems (most expensive part of the algorithm).
- A matrix analysis of the algorithm indicates that, in fact, the upscaling technique is solving the continuous differential equation with density given by averaging along coarse block edges.
- The pressure equation corresponds to the standard acoustic wave equation at nodes internal to coarse blocks.
- Along coarse cell boundaries, the upscaled solution solves a modified wave equation which includes a mixed second-derivative term.
- Using the adjoint method we can reuse much of the upscaling code to solve the inverse problem.
This research was performed with funding from the Collaborative Math-Geoscience Program at NSF (grant #EAR–0222181), the Computational Mathematics Program at NSF (grant #DMS–0714159), a GAANN grant from the U.S. Department of Education (Award #P200A030097), a generous fellowship provided by NASA Goddard’s Earth Sciences and Technology Center (GEST), and UMBC’s ADVANCE grant (NSF #SBE-0244880).