RBF-FD for Forward Seismic Modeling

Presented by: Bengt Fornberg University of Colorado

In collaboration with: Brad Martin University of Colorado
                      Amik St-Cyr Shell Intl. Production and Research
                      Natasha Flyer NCAR
Forward vs. Inverse Modeling

2-D vertical slice near Madagascar:

Region inside dashed rectangle simplified to form standardized Marmousi test problem (shown on next slide)

Figure adapted from Martin, Wiley and Marfurt: Marmousi2: An elastic upgrade for Marmousi (2006)

Forward modeling

Assume subsurface structures known,
Simulate propagation of elastic waves

Inverse modeling

Adjust subsurface assumptions to reconcile forward modeling with seismic data
Elastic wave equation in 2-D

\[
\begin{align*}
\rho u_t &= f_x + g_y \\
\rho v_t &= g_x + h_y \\
f_t &= (\lambda + 2\mu)u_x + \lambda v_y \\
g_t &= \mu(u_x + v_y) \\
h_t &= (\lambda + 2\mu)v_y + \lambda u_x
\end{align*}
\]

Dependent variables:
- \(u, v\) horizontal and vertical velocities
- \(f, g, h\) components of the symmetric stress tensor

Material parameters:
- \(\rho\) density
- \(\lambda, \mu\) Lamé parameters (compression and shear)

Wave types:
- Pressure \(c_p = \sqrt{(\lambda + 2\mu)/\rho}\), Shear \(c_s = \sqrt{\lambda/\rho}\)
- Also: Rayleigh, Love, and Stonley waves
<table>
<thead>
<tr>
<th>Region Type</th>
<th>Dominant Errors</th>
<th>Computational Remedies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Smoothly variable medium</td>
<td>Dispersive errors</td>
<td>High order approximations</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1970’s  From 2\textsuperscript{nd} order to 4\textsuperscript{th} order FD (or FEM)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2010’s  20\textsuperscript{th} order (or higher still) FD</td>
</tr>
<tr>
<td>Interfaces</td>
<td>Reflection and transmission of pressure and shear waves</td>
<td>Analysis based enhancements:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Difficulties for lattice-based discretizations when complex geometries</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Industry standard:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Refine and ‘hope for the best’ (typically 1\textsuperscript{st} order)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Present novelties:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Distribute RBF-FD nodes to align with all interfaces</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(suffices for 2\textsuperscript{nd} order)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Modify basis functions to analytically correct for interface conditions (RBF-FD/AC)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(high order possible also for curved interfaces)</td>
</tr>
</tbody>
</table>
Present concept for spatial discretization

- Distribute nodes that ‘straddle’ the interface, but which transition to become lattice-based a short distance away from it.

- Use high order FD when all stencil nodes are lattice based.

- Use ‘regular’ RBF-FD when any nodes are irregularly placed, but still away from interface.

- Use RBF-FD/AC (Analytic Correction) when an interface intersects a stencil.

- Just straddling the interface suffices to go from 1\textsuperscript{st} to 2\textsuperscript{nd} order of accuracy at the interface.

- Combine with local mapping to make interfaces flat; can ‘correct’ to reach high orders.

Two main ‘ingredients’ needed for calculating weights in RBF-FD stencils

1. Calculation of regular FD weights in 1-D

Several fast and stable algorithms available (Fornberg, SIAM Review, 1998)

Conceptually simplest – find weights so that a differential operator \( L \) becomes treated exactly in case of the trial functions 1, \( x \), \( x^2 \), \( x^3 \),...,\( x^{n-1} \)

\[
\begin{bmatrix}
1 & 1 & \cdots & 1 \\
x_1 & x_2 & \cdots & x_n \\
\vdots & \vdots & \ddots & \vdots \\
x_1^{n-1} & x_2^{n-1} & \cdots & x_n^{n-1}
\end{bmatrix}
\begin{bmatrix}
w_1 \\
w_2 \\
\vdots \\
w_n
\end{bmatrix}
= \begin{bmatrix}
L 1 |_{x=x_c} \\
L x |_{x=x_c} \\
\vdots \\
L x^{n-1} |_{x=x_c}
\end{bmatrix}
\]

2. Calculation of an RBF interpolant

\[
s(x) = \sum_{k=1}^{n} \lambda_k \phi(|| x - x_k ||)
\]

Solve system \( A \ x = f \) , of the form:

\[
\begin{bmatrix}
\phi(|| x_1 - x_1 ||) & \phi(|| x_1 - x_2 ||) & \cdots & \phi(|| x_1 - x_n ||) \\
\phi(|| x_2 - x_1 ||) & \phi(|| x_2 - x_2 ||) & \cdots & \phi(|| x_2 - x_n ||) \\
\vdots & \vdots & \ddots & \vdots \\
\phi(|| x_n - x_1 ||) & \phi(|| x_n - x_2 ||) & \cdots & \phi(|| x_n - x_n ||)
\end{bmatrix}
\begin{bmatrix}
\lambda_1 \\
\lambda_2 \\
\vdots \\
\lambda_n
\end{bmatrix}
= \begin{bmatrix}
f_1 \\
f_2 \\
\vdots \\
f_n
\end{bmatrix}
\]
Calculation of weights in RBF-FD stencil for a differential operator $L$

Choose weights so the result becomes exact for all RBFs interpolants of the form

$$s(x) = \sum_{k=1}^{n} \lambda_k \phi(||x-x_k||) + \{p_m(x)\} \quad \text{with constraints} \quad \sum \lambda_k p_m(x_k) = 0$$

Illustration of system to solve for weights in case of 2-D, when also using up to linear polynomials with corresponding constraints:

$$\begin{bmatrix}
A & 1 & x_1 & y_1 \\
\vdots & \vdots & \vdots & \vdots \\
1 & x_n & y_n & w_n \\
x_1 & \cdots & x_n & 0 \\
y_1 & \cdots & y_n & w_{n+3}
\end{bmatrix}
= 
\begin{bmatrix}
L\phi(||x-x_1||) \big|_{x=x_c} \\
\vdots \\
L\phi(||x-x_n||) \big|_{x=x_c} \\
L^1 \big|_{x=x_c} \\
Lx \big|_{x=x_c} \\
Ly \big|_{x=x_c}
\end{bmatrix}$$

Same $A$-matrix as above; The entries $w_{n+1}, \ldots$ should be ignored.

Common RBF types:  Infinitely smooth, e.g. GA: $\phi(r) = e^{-(er)^2}$, MQ: $\phi(r) = \sqrt{1 + (er)^2}$
or finitely smooth, e.g. PHS: $\phi(r) = r^{2m} \log r, \quad \phi(r) = r^{2m+1}$.

When refining, the polynomial part gradually ‘takes over’ from RBF part.
RBF-FD/AC (Analytic Correction concept), described in 1-D

Equation on each side of interface:
\[
\frac{\partial}{\partial t} \begin{bmatrix} u \\ f \end{bmatrix} = \begin{bmatrix} 0 & \frac{1}{\rho} \frac{\partial}{\partial x} \\ \rho c^2 \frac{\partial}{\partial x} & 0 \end{bmatrix} \begin{bmatrix} u \\ f \end{bmatrix}
\]

(Left and Right denoted L&R, resp.):

Across the interface (at \(x = 0\)):
\[
\begin{bmatrix} u_L \\ f_L \\ u_R \\ f_R \end{bmatrix} = 0 \quad \Rightarrow \quad \frac{\partial^k}{\partial t^k} \left\{ \begin{bmatrix} u_L \\ f_L \\ u_R \\ f_R \end{bmatrix} \right\} = 0, \quad k = 0, 1, 2, \ldots
\]

Translate these \(t\)-derivatives into \(x\)-derivative relations at the interface. This gives equations for how \(u\) and \(f\) ‘kinks’ at the interface.

Key idea: Embed this information in the polynomial basis that supplements the RBFs when calculating the RBF-FD weights (but not in the RBFs).

Novelty: Achieves a procedure that generalizes to larger coupled PDE systems on scattered nodes also in higher dimensions.
In the case of $c_L = 1$, $c_R = 2$, $\rho_L = \rho_R = 1$

For the 2-D elastic wave equation, 4 out of the 5 variables ($u,v,g,h$; representing motion and traction) are continuous at (non-slip) interfaces.

Procedure above generalizes to curved interfaces in both 2-D and 3-D (and beyond)

Subtle point: The requirements for continuity of motion and traction then require more relations to be satisfied than what lattice-based FD can contain when approximating $x$ and $y$ derivatives by 1-D stencils in the respective directions.
‘Mini-Marmousi’ test case

Relative $p$-wave velocity in elastic medium

Initial condition for $v$ at $t = 0$

Solution for $v$ at $t = 0.3$

Errors with RBF-FD/AC discretization, at $t = 0.3$, using $n = 19$ node RBF-FD stencil

$N = 38,400$ nodes

$N = 153,600$ nodes

Typical node separation reduced by factor of two; error reduced by factor of 10, indicating better than 3$^{\text{rd}}$ order in all regions
3-D acoustic wave equation, solved by the RBF-FD/AC procedure

Ricker wavelet initial condition at location (0.5, 0.5, 0.75)
Material interface is here an inclined flat plane, RBF-FD/AC with $N = 10^6$, $n = 61$.

Views from two different angles of the RBF-FD/AC solution at a later time:
Timing comparison against very high order FD

Maximum relative error in reflected wave signal

- FD method (CPU)
- FD method (GPU)
- RBF method (CPU)
- RBF method (GPU)
Work in progress

- Treatment of intersections of interfaces
- Further timing and accuracy comparisons against a test suite used by Shell, using CPUs, GPUs, and very large-scale distributed memory parallel hardware
- Non-uniform RBF discretization also in the time direction
  (as an alternative to MOL, which separates the space and time discretizations)

Conclusions

- High accuracy feasible across curved interfaces by enhancing basic RBF-FD stencils
- Already the present RBF-FD/AC implementation handles interfaces far more effectively than ‘state-of-the-art’ extremely high order lattice-based FD approximations.

Publication

- Martin, B., Fornberg, B., St-Cyr, A., Flyer, N., Seismic modeling with radial basis function-generated finite differences (RBF-FD), to be submitted to *Geophysics*. 