Finite Difference vs. Discontinuous Galerkin: Efficiency in Smooth Models

Mario Bencomo

TRIP Review Meeting 2014

RICE

PROBLEM STATEMENT

Acoustic Equations (pressure-velocity form):

$$\rho(\boldsymbol{x})\frac{\partial \boldsymbol{v}}{\partial t}(\boldsymbol{x},t) + \nabla p(\boldsymbol{x},t) = 0 \tag{1a}$$

$$\beta(\boldsymbol{x})\frac{\partial p}{\partial t}(\boldsymbol{x},t) + \nabla \cdot \boldsymbol{v}(\boldsymbol{x},t) = f(\boldsymbol{x},t) \tag{1b}$$

for $\boldsymbol{x} = [x, z]^T \in \Omega$ and $t \in [0, T]$,

- $p =$ pressure
- $\boldsymbol{v} = [v_x, v_z]^T =$ velocity fields
- $\rho =$ density
- $\beta =$ compressibility $= 1/\kappa$

Boundary and initial conditions:

$$p = 0, \quad \text{on } \partial\Omega \times [0, T]$$

$$p(\boldsymbol{x}, 0) = 0 \quad \text{and} \quad \boldsymbol{v}(\boldsymbol{x}, 0) = 0$$

MOTIVATION

Why DG?

- viable numerical method for forward modeling (discontinuous media)
- outperforms FD methods when using mesh aligning techniques for complex discontinuous media

Why smooth media?

- smooth trends in bulk modulus and density are observed in real data
- forward map applied to smooth background model in inversion

Comparison between FD and DG in smooth media has not been done before!

FOCUS OF RESEARCH

Disclaimer: limited comparison

- will not incorporate HPC architectures (stuck with Matlab for DG code)
- FD code in IWAVE (implemented in C)

What kind of comparison?

- counting FLOPs for a prescribed accuracy

OUTLINE

1. Background
   - literature review
   - motivate study
2. Methods
   - staggered finite difference method
   - discontinuous Galerkin method
3. Numerical Experiments and Results
4. Conclusions
5. Future Work

BACKGROUND:
Finite Difference (FD) Method

Will be considering 2-2 and 2-4 staggered grid finite difference schemes *(Virieux 1986, Levander 1988)*. Numerical properties well known:

- Stability criterion:

$$\Delta t < \frac{1}{\sqrt{2}V_p}h \quad \text{(2-2 FD)}$$

$$\Delta t < \frac{0.606}{V_p}h \quad \text{(2-4 FD)}$$

  where $h =$ grid size, and $V_p =$ compressional velocity.

- Common rule of thumb for small grid dispersion:
  10 or 5 grid points per wavelength, for the shortest wavelength (2-2 and 2-4 resp.)

BACKGROUND: Discontinuous Galerkin (DG) Method

First introduced for the neutron transport problem *(Lesaint and Raviart 1974)*:

- gained popularity due to geometric flexibility and mesh and polynomial order adaptivity (*hp* adaptivity)
- can yield explicit schemes after inverting block diagonal matrix

Stability criterion:

- will be considering Runge-Kutta DG (RK-DG) scheme with upwind flux *(Hesthaven and Warburton, 2002, 2007)*:

$$\Delta t \leq \frac{CFL}{v_{max}} \min_{\Omega} h$$

where $CFL = \mathcal{O}(N^{-2})$

BACKGROUND:
Discontinuous Galerkin (DG) Method

Grid dispersion and dissipation errors:

- dissipation due to upwind flux

Ainsworth (2004):

- study of dissipation and dispersion error under *hp* refinement, applied to linear advection equation
- polynomial order N can be chosen such that dispersion error decays super-exponentially if $2p+1 \approx chN$, for given mesh size h, wavenumber k, and some constant $c > 1$

Hu et al. (1999):

- anisotropic dispersion and dissipation errors in quadrilateral and triangular uniform meshes for DG applied to 2D wave problems

BACKGROUND: RK-DG and Seismic Modeling

Wang (2009), and Wang et al.(2010)

- comparison of FD and DG in 2D acoustic wave propagation with discontinuous media
- interface error over discontinuities reduce convergence rates of FD methods to 1st order while DG scheme with aligned mesh yields sub-optimal second order rates

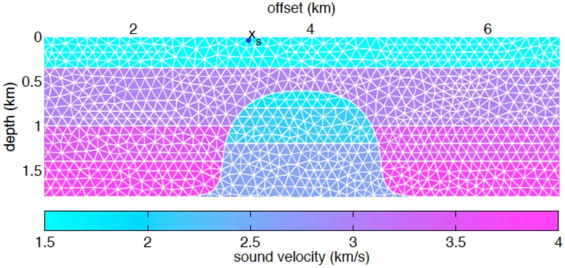


Figure: Dome model from Wang (2009).

BACKGROUND: RK-DG ans Seismic Modeling

Simonaho et al. (2012), Applied Acoustics journal:

- DG simulations of 3D acoustic wave propagation compared to real data; pulse propagation and scattering from a cylinder

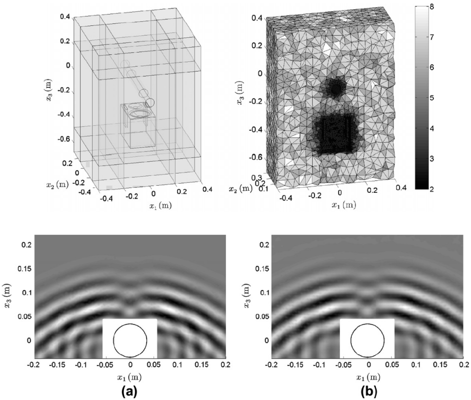
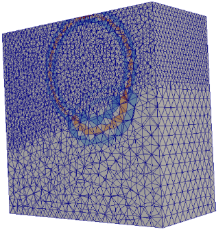


Figure: Snapshots of (a) simulated and (b) measured reassure field.

BACKGROUND: RK-DG and Seismic Modeling

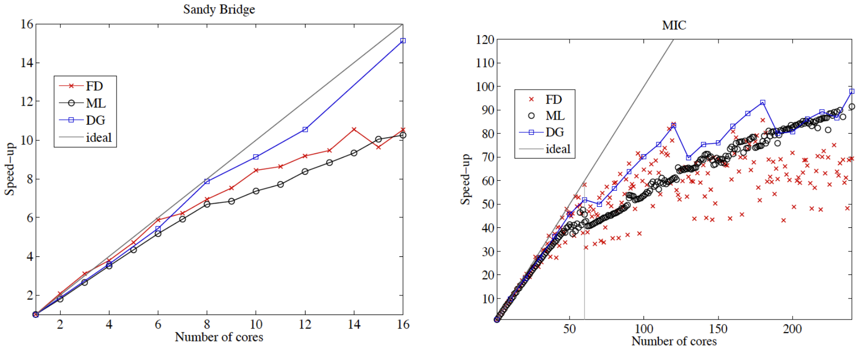
Zhebel et al. (2013):

- perform study on parallel scalability of FD and finite element methods (mass lumped finite elements and DG) for 3D acoustic wave propagation with piecewise constant media with dipping interface
- hardware: Intel Sandy Bridge dual 8-core machine and Intel's 61-core Xeon Phi



BACKGROUND:
RK-DG and Seismic Modeling

Zhebel et al. (2013)



BACKGROUND:
DG and Smooth Coefficients

Quadrature-free implementations *(Shu 1998, Hesthaven and Warburton 2007)*:

- assumes that media is piecewise constant on mesh
- lower memory cost associated with storing DG operators

Quadrature based implementations *(Ober et al. 2010, Collis et al. 2010)*:

- weighted inner products between basis functions computed via quadrature

This study will compare both implementations, along with FD methods for smoothly varying coefficients.

METHODS:
Finite Difference (FD)

2-2k staggered FD method applied to 2D acoustic wave equation in first order form:

$$(v_x)^{n+1}_{i+\frac{1}{2},j} = (v_x)^{n}_{i+\frac{1}{2},j} + \Delta t \frac{1}{(\rho)_{i+\frac{1}{2},j}} \Big\{ -D^{h,(k)}_x (p)^{n+\frac{1}{2}}_{i+\frac{1}{2},j} \Big\}$$

$$(v_y)^{n+1}_{i,j+\frac{1}{2}} = (v_y)^{n}_{i,j+\frac{1}{2}} + \Delta t \frac{1}{(\rho)_{i,j+\frac{1}{2}}} \Big\{ -D^{h,(k)}_y (p)^{n+\frac{1}{2}}_{i,j+\frac{1}{2}} \Big\}$$

$$(p)^{n+\frac{1}{2}}_{ij} = (p)^{n-\frac{1}{2}}_{ij} + \Delta t \frac{1}{(\beta)_{ij}} \Big\{ -D^{h,(k)}_x (v_x)^n_{i,j} - D^{h,(k)}_y (v_y)^n_{i,j} + (f)^n_{i,j} \Big\},$$

where $p^{n+\frac{1}{2}}_{i,j} = p(ih, jh, (n+\frac{1}{2})\Delta t)$, and

$$D^{h,(k)}_x f(x_0) := \frac{1}{h} \sum_{n=1}^{k} a^{(k)}_n \Big\{ f\Big(x_0 + \Big(n-\frac{1}{2}\Big)h\Big) - f\Big(x_0 - \Big(n-\frac{1}{2}\Big)h\Big) \Big\}.$$

METHODS:
DG Method Introduction

Definition: For a given triangulation $\mathscr{T}_h$, define approximation space $\mathscr{W}_h$,

$$\mathscr{W}_h = \{w : w|_\tau \in \mathbb{P}^N(\tau), \forall \tau \in \mathscr{T}\}$$

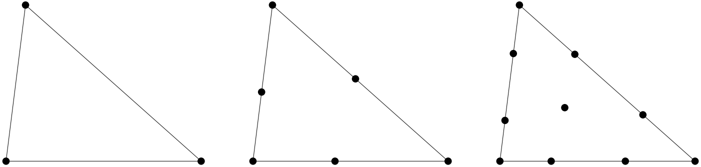
Nodal DG: Use nodal basis, i.e.,

$$\mathbb{P}^N(\tau) = \text{span}\{\ell_j(\boldsymbol{x})\}_{j=1}^{N^*} \quad \forall \tau \in \mathscr{T},$$

where

- Lagrange polynomials $\ell_j(\boldsymbol{x}_i) = \delta_{ij}$ for given nodal set $\{\boldsymbol{x}_i\}_{i=1}^{N^*} \subset \tau$
- $N^* = \frac{1}{2}(N+1)(N+2)$, a.k.a., degrees of freedom per triangular element

Example of nodal sets $\{\boldsymbol{x}_j\}_{j=1}^{N^*}$:



(a) $N = 1, N^* = 3$ (b) $N = 2, N^* = 6$ (c) $N = 3, N^* = 10$

METHODS:
DG Method Introduction

Strong-formulation: find $p, v_x, v_z \in \mathscr{W}_h$ such that

$$\int_\tau \rho \frac{\partial v_x}{\partial t} w \, d\boldsymbol{x} + \int_\tau \frac{\partial p}{\partial x} w \, d\boldsymbol{x} + \int_{\partial \tau} \hat{n}_x (p^* - p) w \, d\sigma = 0$$

$$\vdots$$

for all $w \in \mathscr{W}_h$ and all $\tau \in \mathscr{T}_h$.

Flux term $(p^* - p)$:

- how “information” propagates from element tot element
- plays role in stability of method
- derivation of numerical fluxes $p^*, \boldsymbol{v}^*$ for this case stem from Riemmann solvers

METHODS: Nodal Coefficient Vectors

Note $\mathscr{W}_h$ is finite dimensional space $\Longrightarrow$ need only to solve for nodal coefficients $p(\boldsymbol{x}_j, t)$:

$$p|_\tau = \sum_{j=1}^{N^*} p(\boldsymbol{x}_j, t)\ell_j(\boldsymbol{x})$$

Nodal coefficient vectors:

$$\mathbb{R}^{N^*} \quad \begin{cases} \mathbf{p}(t) & := \quad [p(\boldsymbol{x}_1, t), p(\boldsymbol{x}_2, t), \ldots, p(\boldsymbol{x}_{N^*}, t)]^T \\ \mathbf{v}_x(t) & := \quad \cdots \\ \mathbf{v}_z(t) & := \quad \cdots \end{cases}$$

$$\mathbb{R}^{N+1} \quad \begin{cases} \mathbf{p}^{(e)}(t) & := \quad [p(\boldsymbol{x}_{m_1}, t), p(\boldsymbol{x}_2, t), \ldots, p(\boldsymbol{x}_{m_{N+1}}, t)]^T \\ \quad \vdots & \end{cases}$$

METHODS: DG Semi-Discrete Scheme

From strong formulation: find $p, v_x, v_z \in \mathscr{W}_h$ such that

$$\int_\tau \rho \frac{\partial v_x}{\partial t} w \, d\boldsymbol{x} + \int_\tau \frac{\partial p}{\partial x} w \, d\boldsymbol{x} + \int_{\partial\tau} \hat{n}_x (p^* - p) w \, d\sigma = 0$$

$$\vdots$$

for all $w \in \mathscr{W}_h$ and all $\tau \in \mathscr{T}_h$.

To DG semi-discrete scheme:

$$M[\rho] \frac{d}{dt} \mathbf{v}_x(t) + S^x \mathbf{p}(t) + \sum_{e \in \partial\tau} \hat{n}_x M^{(e)} \left((\mathbf{p}^{(e)})^* - \mathbf{p}^{(e)} \right)(t) = 0,$$

$$\vdots$$

for each $\tau \in \mathscr{T}_h$.

METHODS:
DG Semi-Discrete Scheme

DG operators:

$$
\begin{array}{rcll}
\text{weighted mass matrix} \quad M[\omega]_{ij} & := & \displaystyle\int_{\tau} \omega \ell_i \ell_j \, d\boldsymbol{x}, & \text{in } \mathbb{R}^{N^* \times N^*} \\
\text{edge mass matrix} \quad M^{(e)}{}_{ij} & := & \displaystyle\int_{e} \ell_i \ell_{m_j} \, d\sigma, & \text{in } \mathbb{R}^{N^* \times (N+1)} \\
\alpha\text{-stiffness matrix} \quad S^{\alpha}{}_{ij} & := & \displaystyle\int_{\tau} \ell_i \frac{\partial \ell_j}{\partial \alpha} \, d\boldsymbol{x}, & \text{in } \mathbb{R}^{N^* \times N^*}
\end{array}
$$

for $\omega \in \{\rho, \beta\}$ and $\alpha \in \{x, z\}$.

METHODS:
DG Semi-Discrete Scheme

Explicit Scheme:

$$\frac{d}{dt}\mathbf{v}_x(t) = -D^x[\tfrac{1}{\rho}]\mathbf{p}(t) + \sum_{e\in\partial\tau} \hat{n}_x L^{(e)}[\tfrac{1}{\rho}]\left((\mathbf{p}^{(e)})^* - \mathbf{p}^{(e)}\right)(t)$$

$$\vdots$$

for each $\tau \in \mathscr{T}_h$, where

$$D^\alpha[\tfrac{1}{\omega}] = M[\omega]^{-1}S^\alpha, \quad L^{(e)}[\tfrac{1}{\omega}] = M[\omega]^{-1}M^{(e)}$$

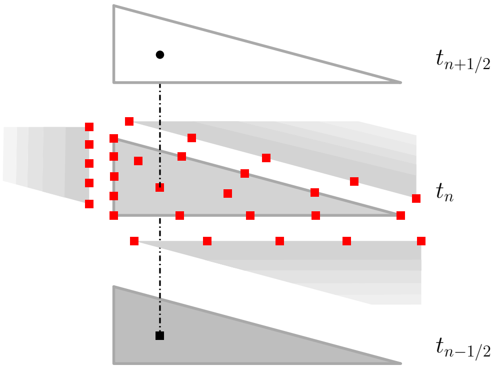
$$\vdots$$

for $\omega \in \{\alpha, \beta\}$ and $\alpha \in \{x, z\}$.

METHODS:
Visualizing DG

After time discretization (leapfrog):

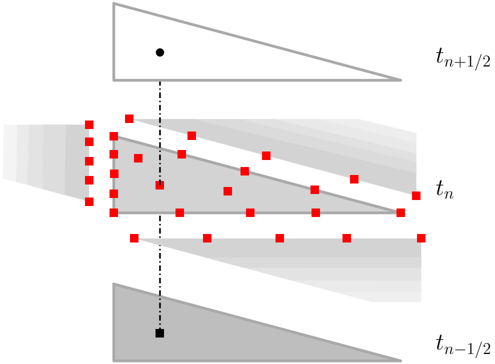
$$\mathbf{v}_x^{n+1/2} = \mathbf{v}_x^{n-1/2} - \Delta t \left[D^x[\tfrac{1}{\rho}]\mathbf{p}^n + \sum_{e \in \partial\tau} \hat{n}_x L^{(e)} \left((\mathbf{p}^{(e)})^* - \mathbf{p}^{(e)} \right)^n \right]$$



METHODS: Visualizing DG

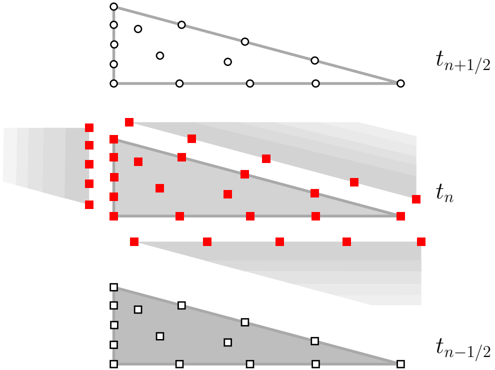
After time discretization (leapfrog):

$$\mathbf{v}_x^{n+1/2} = \mathbf{v}_x^{n-1/2} - \Delta t[D^x[\tfrac{1}{\rho}]\mathbf{p}^n + \sum_{e \in \partial\tau} \hat{n}_x L^{(e)} \left((\mathbf{p}^{(e)})^* - \mathbf{p}^{(e)} \right)^n]$$



After time discretization (leapfrog):

$$\mathbf{v}_x^{n+1/2} = \mathbf{v}_x^{n-1/2} - \Delta t [D^x [\tfrac{1}{\rho}] \mathbf{p}^n + \sum_{e \in \partial\tau} \hat{n}_x L^{(e)} \left((\mathbf{p}^{(e)})^* - \mathbf{p}^{(e)} \right)^n]$$



METHODS:
Handling Varying Coefficients: Quadrature-Free Approach

Idea: assume $\omega \in \{\rho, \beta\}$ are constant within $\tau \Longrightarrow$ media is piecewise constant

- mass matrix computations

$$\int_\tau \omega \ell_j \ell_i \, d\boldsymbol{x} = \omega(\tau) J(\tau) \int_{\hat{\tau}} \hat{\ell}_j \hat{\ell}_i \, d\hat{\boldsymbol{x}} \Longrightarrow M[\omega] = \omega(\tau) J(\tau) \hat{M}$$

- for variable media *(LeVeque 2002)*:

$$\omega(\tau) = \frac{1}{|\tau|} \int_\tau \omega \, d\boldsymbol{x}$$

METHODS:
Handling Varying Coefficients: Quadrature-Free Approach

- Compute

$$D^{\alpha}[\tfrac{1}{\omega}] = \tfrac{1}{\omega}\left(c_1 D^{\hat{x}} + c_2 D^{\hat{z}}\right), \quad L^{(e)}[\tfrac{1}{\omega}] = \tfrac{1}{\omega} c_3 L^{(\hat{e})},$$

  at run time, and only need to store geometric factors and one copy of operators defined on some reference element *(Hesthaven and Warburton 2007)*.

- memory storage: K triangular elements, using polynomial order N,

$$\begin{aligned} \text{memory} \quad &\approx \quad c_1 K + c_2(N^* \times N^*) + c_3(N^* \times (N+1)) \\ &\approx \quad \mathscr{O}(K + N^4) \end{aligned}$$

METHODS:
Handling Varying Coefficients: With Quadrature

Idea: Compute integrals up to accuracy $2N+Q$

$$\int_\tau \omega \ell_i \ell_j \, d\boldsymbol{x}$$

- higher accuracy
- operators $D^x[\frac{1}{\omega}], D^z[\frac{1}{\omega}], L^{(e)}[\frac{1}{\omega}]$ are computed offline and stored
- memory storage: K triangular elements, using polynomial order N,

$$\begin{aligned} \text{memory} &\approx c_1 K(N^* \times N^*) + c_2 K(N^* \times (N+1)) \\ &\approx \mathcal{O}(KN^4) \end{aligned}$$

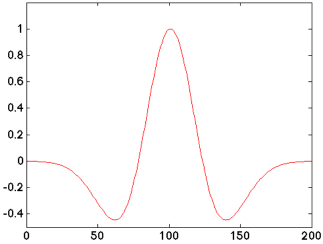
For all simulations:

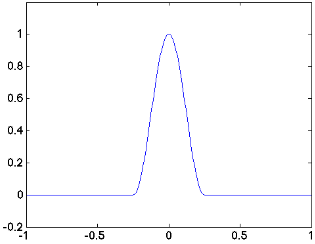
- source term $f(\boldsymbol{x}, t) = \chi(\boldsymbol{x})\Psi(t)$, where

$$\Psi(t) = \Psi(t; f_{peak}) = \text{ Ricker wavelet}$$

$$\chi(\boldsymbol{x}) = \chi(\boldsymbol{x}; \boldsymbol{x}_c, \delta_x) = \text{ cosine bump function}$$

with $f_{peak} = 10\ Hz$ and $\delta_x = [50\ m, 50\ m]$





NUMERICAL EXPERIMENTS: Estimating Convergence Rates

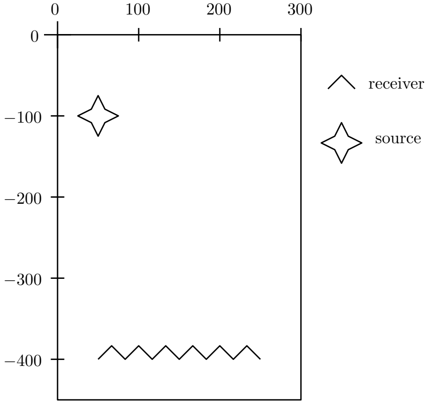
Estimating convergence rates using Richardson extrapolation:

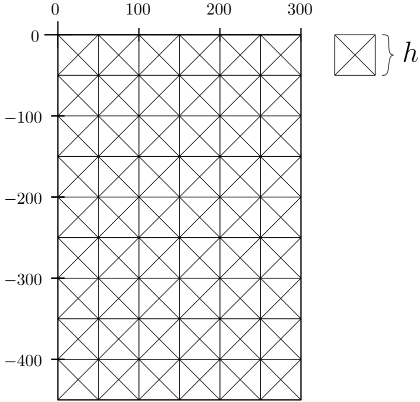
$$R \approx log_2 \frac{|\boldsymbol{p}_h - \boldsymbol{p}_{h/2}|}{|\boldsymbol{p}_{h/2} - \boldsymbol{p}_{h/4}|}$$

Setup

- homogenous model: $\rho = 2.3\ g/cm^3, c = 3\ km/s$
- uniform triangulation for DG method
- final time $T = 350\ ms$

NUMERICAL EXPERIMENTS: Estimating Convergence Rates





RESULTS:
Estimating Convergence Rates

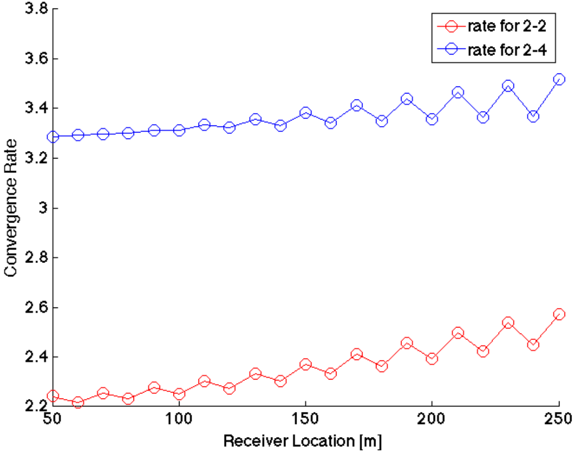


Figure: Estimated convergence rates for 2-2 and 2-4 FD methods.

RESULTS:
Estimating Convergence Rates

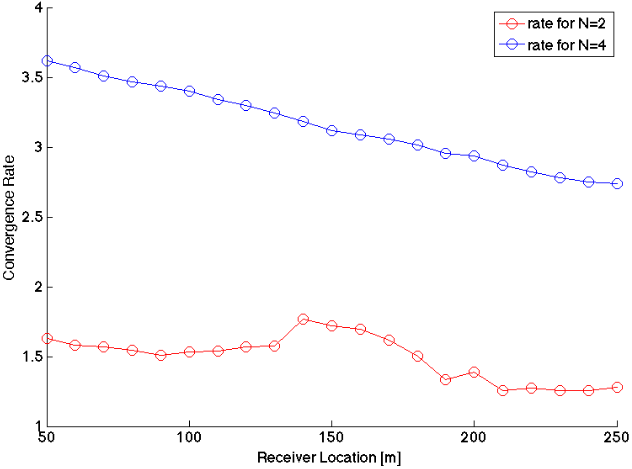


Figure: Estimated convergence rates for DG methods.

NUMERICAL EXPERIMENTS: Calibration

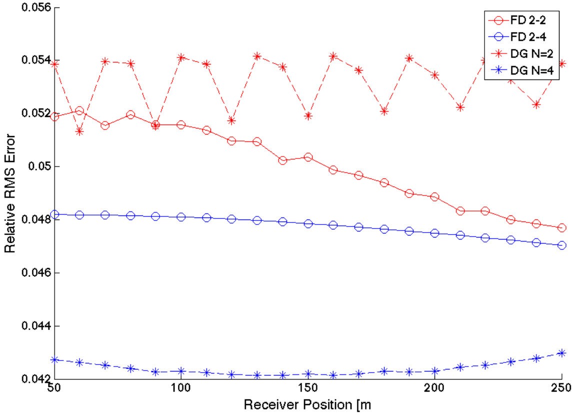
Idea: compare numerical $\boldsymbol{p}$ from DG and FD to highly discretized FD solution

Setup:

- similar to convergence test, i.e., homogeneous model
- final time $T = 350$ *ms*

	fine FD	FD 2-2	FD 2-4	DG $N = 2$	DG $N = 4$
h [m]	0.5	7	10	30	80

RESULTS: Calibration

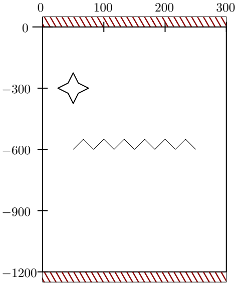


NUMERICAL EXPERIMENTS: Accuracy and Efficiency

Idea: compare accuracy of methods after traveling multiple wavelengths

Setup:

- similar to convergence test, i.e., homogeneous model
- final time $T = 750$ ms
- free-surface boundary conditions at top and bottom of domain



RESULTS:
Accuracy and Efficiency

RESULTS: Accuracy and Efficiency

	fine FD	FD 2-2	FD 2-4	DG $N=2$	DG $N=4$
h [m]	0.5	7	10	30	80
GFLOPS					
runtime [min.]					

Pending Work and Future Directions

Pending work:

- numerical experiments with other smooth models, e.g., Gaussian lens, sinusoidal in depth velocity models
- finish implementing staggered DG method *(Chung & Engquist, 2009)*
  - energy conservative and optimally convergent

Future directions:

- incorporate parallel programming

References

Brezzi, F., Marini, L., and Sïli, E. (2004). “Discontinuous galerkin methods for first-order hyperbolic problems.” Mathematical models and methods in applied sciences, 14(12):1893-1903.

Chung, E. and Engquist, B. (2009). “Optimal discontinuous galerkin methods for the acoustic wave equation in higher dimensions.” SIAM Journal on Numerical Analysis, 47(5):3820-3848.

Chung, E. T. and Engquist, B. (2006). “Optimal discontinuous galerkin methods for wave propagation.” SIAM Journal on Numerical Analysis, 44(5):2131-2158.

Cockburn, B. (2003). “Discontinuous Galerkin methods.” ZAMM-Journal of Applied Mathematics and Mechanics/ Zeitschrift für Angewandte Mathematik und Mechanik, 83(11):731-754.

Wilcox, Lucas C., et al. “A high-order discontinuous Galerkin method for wave propagation through coupled elasticÐacoustic media.” Journal of Computational Physics 229.24 (2010): 9373-9396.

Hesthaven, J. S. and Warburton, T. (2007). “Nodal discontinuous Galerkin methods: algorithms, analysis, and applications,” volume 54. Springer.

Wang, X. (2009). “Discontinuous Galerkin Time Domain Methods for Acoustics and Comparison with Finite Difference Time Domain Methods.” Master’s thesis, Rice University, Houston, Texas.

Warburton, T. and Embree, M. (2006). “The role of the penalty in the local discontinuous Galerkin method for Maxwell’s eigenvalue problem.” Computer methods in applied mechanics and engineering, 195(25):3205-3223.