Introduction of GPR Forward Modeling

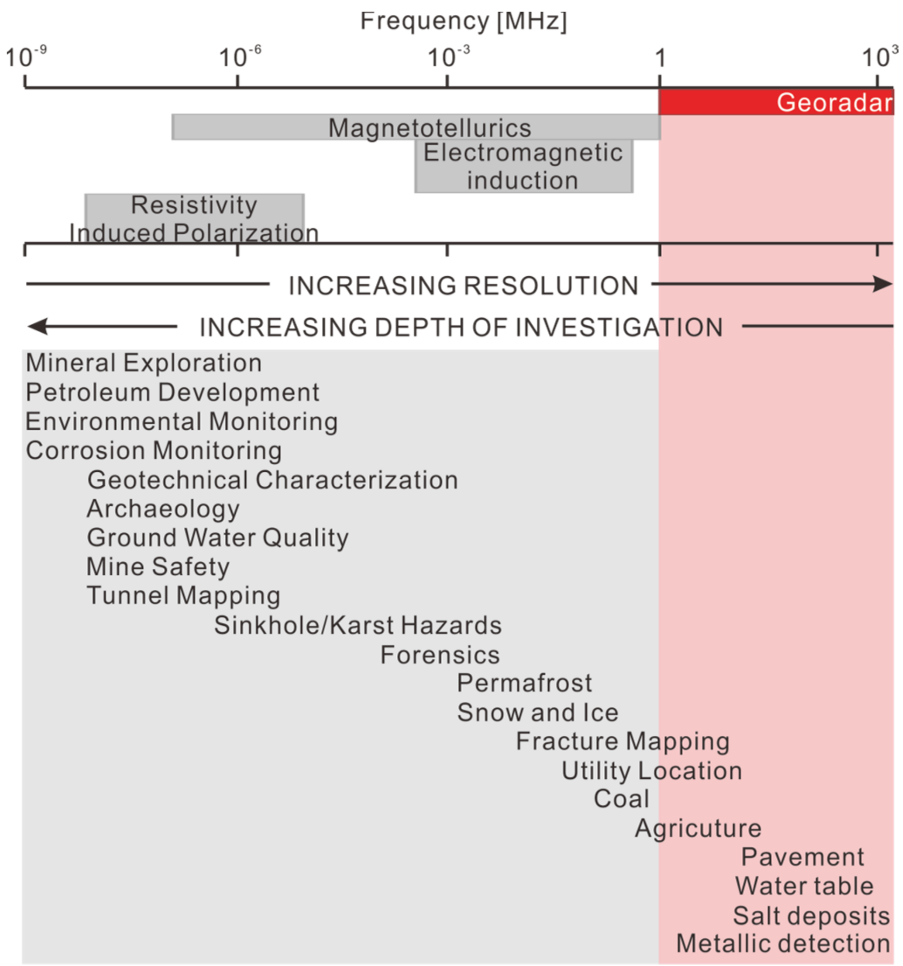
Xi Yang

26 July 2016

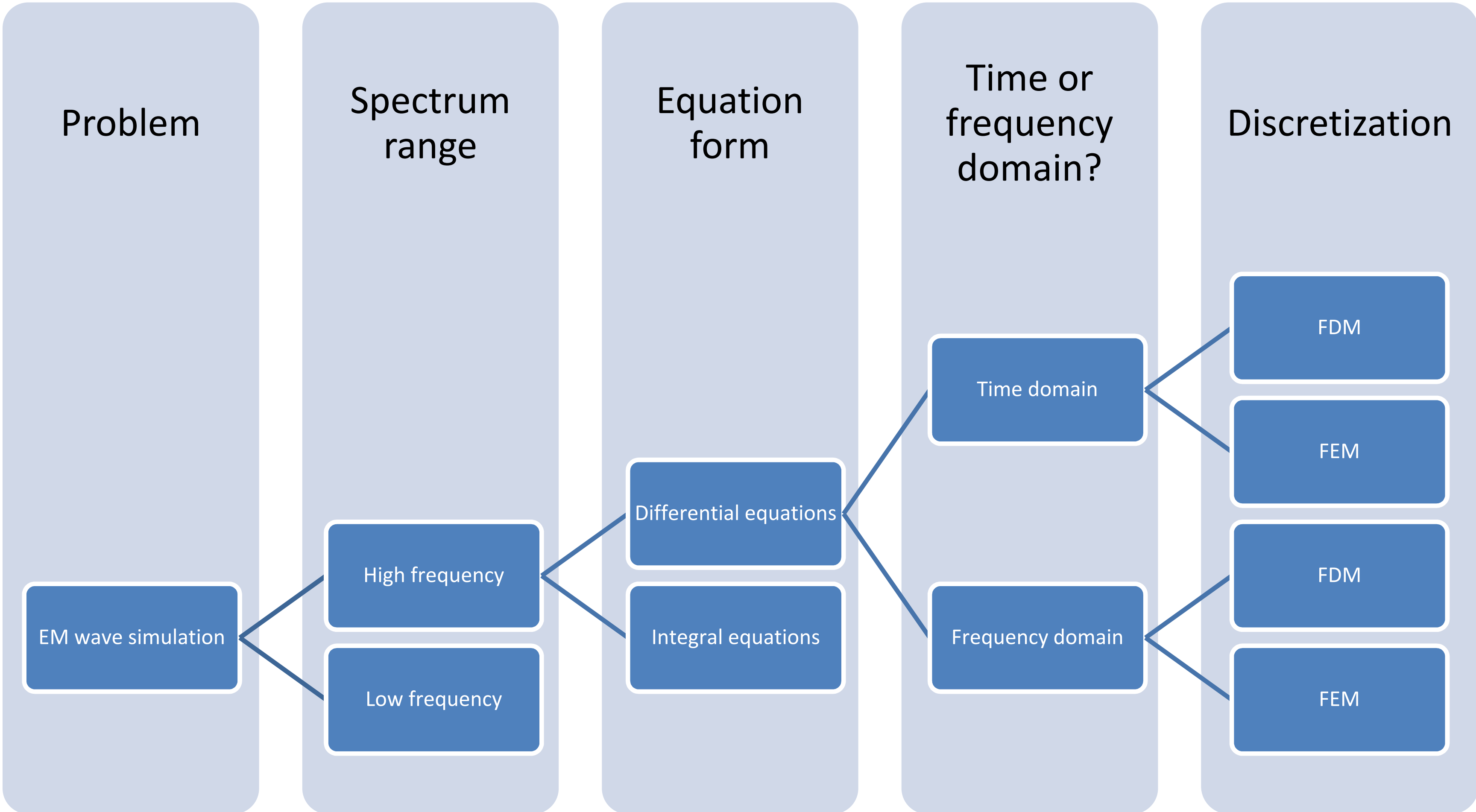
Outline

- Method choice
- Differential equations vs. Integral equations
- FDM vs. FEM
  - Finite difference vs. Discontinuous Galerkin (in 2D)
- Time domain vs. Frequency domain
- Direct solver vs. Iterative solver

Spectrum of geophysical methods



Method choice



Maxwell's equations

$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t}$ $\nabla \times \mathbf{H} = \frac{\partial \mathbf{D}}{\partial t} + \mathbf{J}$ $\nabla \cdot \mathbf{D} = \rho_v$ $\nabla \cdot \mathbf{B} = 0$	Faraday's law Ampere's law Coulomb's law Gauss's law
$\oint_{\partial\Sigma} \mathbf{E} \cdot \mathrm{d}\boldsymbol{\ell} = -\frac{\mathrm{d}}{\mathrm{d}t} \iint_{\Sigma} \mathbf{B} \cdot \mathrm{d}\mathbf{S}$ $\oint_{\partial\Sigma} \mathbf{B} \cdot \mathrm{d}\boldsymbol{\ell} = \mu_0 \varepsilon_0 \frac{\mathrm{d}}{\mathrm{d}t} \iint_{\Sigma} \mathbf{E} \cdot \mathrm{d}\mathbf{S} + \mu_0 \iint_{\Sigma} \mathbf{J} \cdot \mathrm{d}\mathbf{S}$ $\oiint_{\partial\Omega} \mathbf{E} \cdot \mathrm{d}\mathbf{S} = \frac{1}{\varepsilon_0} \iiint_{\Omega} \rho_v \mathrm{d}\boldsymbol{V}$ $\oiint_{\partial\Omega} \mathbf{B} \cdot \mathrm{d}\mathbf{S} = 0$	$\oiint_{\partial\Omega}$ —is a surface integral $\iiint_{\Omega}$ —is a volume integral $\iint_{\Sigma}$ —is a surface integral $\oint_{\partial\Sigma}$ —is a line integral

DE vs. IE

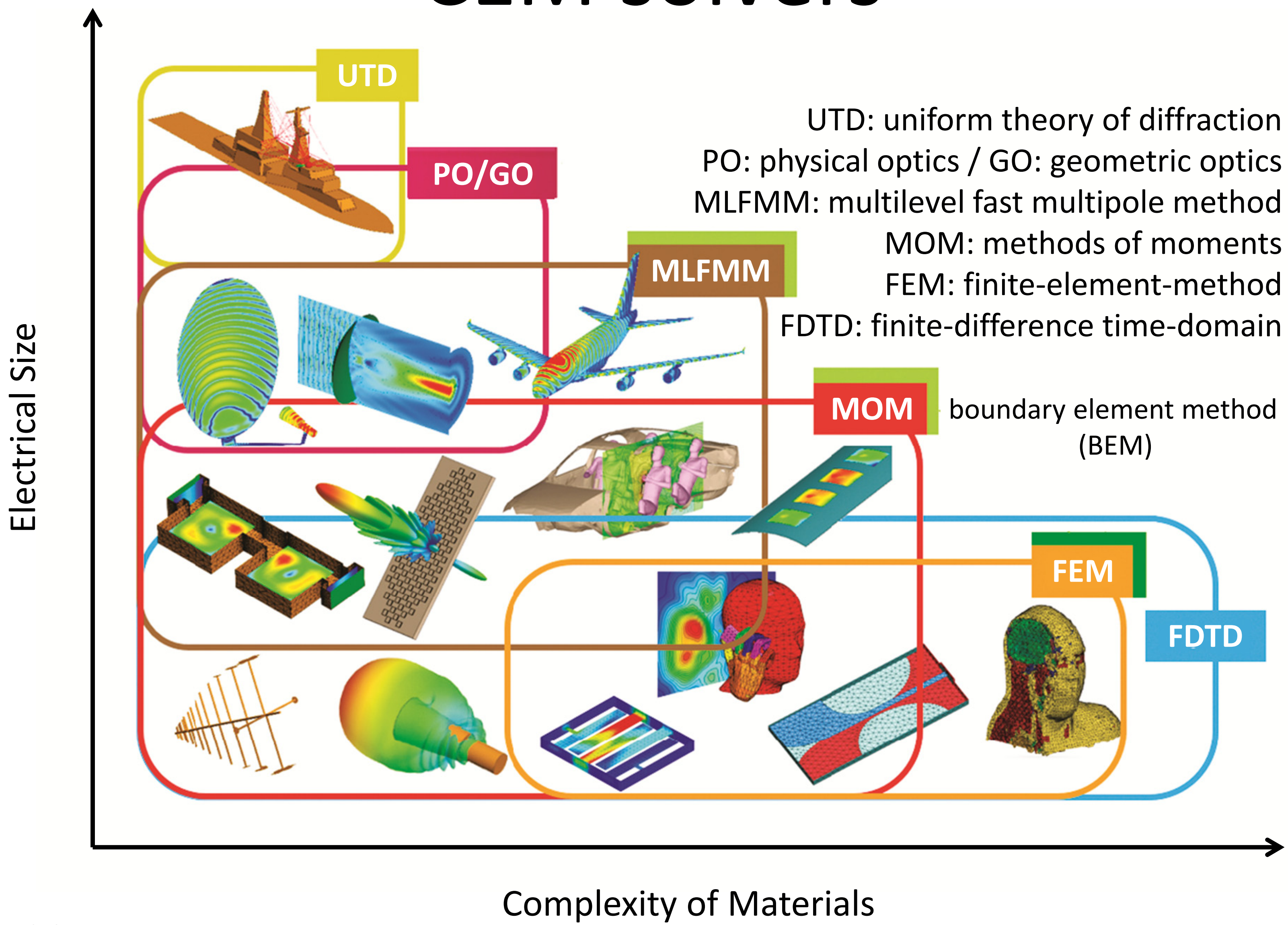
	Differential equations	Integral equations
Advantage	• Numerically stable • Sparse symmetric Matrix • Easy for coding • Effective for inhomogeneous dielectric systems • Easy switch between time domain and frequency domain (Nyquist) • Wide frequency range (FDTD)	• Efficient for problems with low surface-to-volume ratios • No ABC needed • No grid dispersion error (Green function) • Can use MLFMM technic • Several billion unknowns • High accuracy
Disadvantage	• ABC needed • No accelerating algorithm • Suite for electrically small dimensions • Low accuracy • Accumulated error	• Difficult to simulate non-linearly (inhomogeneous) media • Mainly used for metallic objects, dielectric structures, and radiation in free space • Difficult for coding • Dense matrix (Singular matrix) • Only frequency domain

DE vs. IE

	Differential equations	Integral equations
Advantage	• Numerically stable • Sparse symmetric Matrix • Easy for coding • Effective for inhomogeneous dielectric systems • Easy switch between time domain and frequency domain (Nyquist) • Wide frequency range (FDTD)	• Efficient for problems with low surface-to-volume ratios • No ABC needed • No grid dispersion error (Green function) • Can use MLFMM technic • Several billion unknowns • High accuracy
Disadvantage	• ABC needed • No accelerating algorithm • Suite for electrically small dimensions • Low accuracy • Accumulated error	• Difficult to simulate non-linearly (inhomogeneous) media • Mainly used for metallic objects, dielectric structures, and radiation in free space • Difficult for coding • Dense matrix (Singular matrix) • Only frequency domain

The FDTD method is best suited for the EM simulation of wide frequency band and applications using inhomogeneous materials.

CEM solvers


UTD
PO/GO
MLFMM
MOM
FEM
FDTD
UTD: uniform theory of diffraction
PO: physical optics / GO: geometric optics
MLFMM: multilevel fast multipole method
MOM: methods of moments
FEM: finite-element-method
FDTD: finite-difference time-domain
boundary element method (BEM)
Electrical Size
Complexity of Materials

FDM vs. FEM

	FDM	FEM
Advantage	• Easy to understand and implement • Flexibility for multi-physics problem • High-order is feasible • Explicit in time • Direction can be exploited - upwind	• Unstructured mesh (complex geometry) • High accuracy • higher order is straightforward in space (elements with variable size *h*, polynomial degree of the local approximations *p*)
Disadvantage	• Structured (staggered) mesh • Simple local approximation	• More memory requirement • Additional numerical stabilization • higher order in time is difficult (stability reasons)

FDM vs. FEM

	FDM	FEM
Advantage	• Easy to understand and implement • Flexibility for multi-physics problem • High-order is feasible • Explicit in time • Direction can be exploited - upwind	• Unstructured mesh (complex geometry) • High accuracy • higher order is straightforward in space (elements with variable size *h*, polynomial degree of the local approximations *p*)
Disadvantage	• Structured (staggered) mesh • Simple local approximation	• More memory requirement • Additional numerical stabilization • higher order in time is difficult (stability reasons)

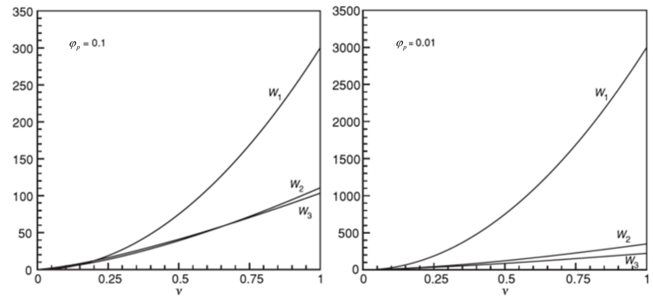
Generally speaking, the only advantage for FEM is the high accuracy results (high order accuracy and unstructured mesh).

Is high-order accuracy necessary?

For a given accuracy φ_p and a specific period of time v we have:

$$\text{Memory} \propto \left(\frac{v}{\varphi_p}\right)^{\frac{d}{2m}}, \quad \text{Work} \propto (2m)^d\, v \left(\frac{v}{\varphi_p}\right)^{\frac{d+1}{2m}}$$

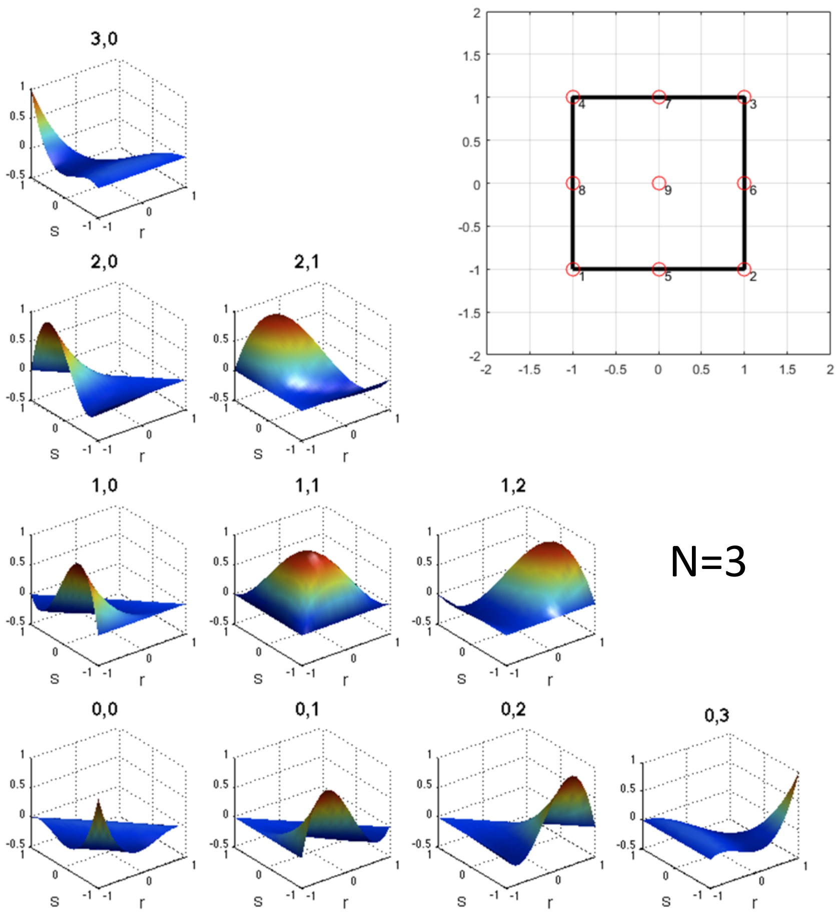
2m = order of the scheme
d = dimension of the problem
the number of points per wavelength



$$p\left(v, \varphi_p\right) \propto \sqrt[2m]{\frac{v}{\varphi_p}}$$

The simulation goes to long time integration, high-dimensional problems (3D) and memory restrictions becomes a bottleneck.

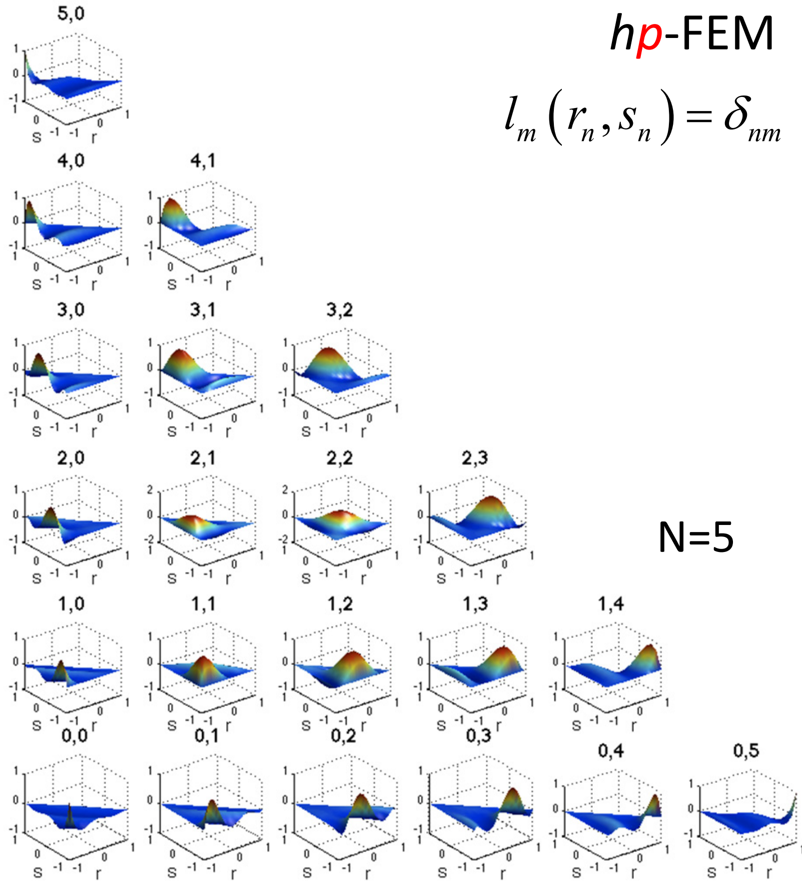
Lagrange Interpolant Basis Functions



N=3

*h**p***-FEM

$$l_m\left(r_n, s_n\right) = \delta_{nm}$$



N=5

Advantage of DG

	FVM	FEM
Advantage	• Robust and fast due to locality • Complex geometries • Well suited for conservation laws • Explicit in time	• High-order accuracy and complex geometries can be combined
Disadvantage	• Inability to achieve high-order on general grids due to extended stencils • Grid smoothness requirements	• Implicit in time • Not well suited for problems with direction • Large memory needed (matrix for whole domain)

Advantage of DG

	FVM	FEM
Advantage	• Robust and fast due to locality • Complex geometries • Well suited for conservation laws • Explicit in time	• High-order accuracy and complex geometries can be combined
Disadvantage	• Inability to achieve high-order on general grids due to extended stencils • Grid smoothness requirements	• Implicit in time • Not well suited for problems with direction • Large memory needed (matrix for whole domain)

What we need is a scheme that combines:

- The local high-order/flexible element of FEM
- The local statement on the equation for FVM

Discontinuous Galerkin Finite Element Method

Advantage of DG

	FVM	FEM
Advantage	• Robust and fast due to locality • Complex geometries • Well suited for conservation laws • Explicit in time	• High-order accuracy and complex geometries can be combined
Disadvantage	• Inability to achieve high-order on general grids due to extended stencils • Grid smoothness requirements	• Implicit in time • Not well suited for problems with direction • Large memory needed (matrix for whole domain)

	Complex geometries	High-order accuracy and hp-adaptivity	Explicit semi discrete form	Conservation laws	Elliptic problems
FDM	x	√	√	√	√
FVM	√	x	√	√	(√)
FEM	√	√	x	(√)	√
DG-FEM	√	√	√	√	(√)

Accuracy dependency

- Assume the earth medium is a heterogeneous isotropic lossy continuum.
- For the given space–time discretization the accuracy of the numerical modelling depends on:
  - accuracy in
    - a heterogeneous medium (How much heterogeneity)
    - a smoothly spatially varying medium (spatial variability of material parameters)
  - accuracy at
    - a material interface (geometry, continuity)
    - a free surface (geometry, radiation pattern)
  - accuracy of
    - a grid boundary (transparency or symmetry)
    - simulation of source (location, mechanism, time function)
    - incorporation of attenuation (frequency dependence, spatial variability)

Accuracy dependency

- Assume the earth medium is a heterogeneous isotropic lossy continuum.
- For the given space–time discretization the accuracy of the numerical modelling depends on:
  - accuracy in
    - a heterogeneous medium (How much heterogeneity)
    - a smoothly spatially varying medium (spatial variability of material parameters)
  - accuracy at
    - a material interface (geometry, continuity)
    - a free surface (geometry, radiation pattern)
  - accuracy of
    - a grid boundary (transparency or symmetry)
    - simulation of source (location, mechanism, time function)
    - incorporation of attenuation (frequency dependence, spatial variability)

High-order accuracy is necessary for geophysics!

FD

$$\begin{bmatrix} D_x^H \hat{\tilde{H}}_y - D_y^H \hat{\tilde{H}}_x = \varepsilon_{zz} \hat{E}_z \\ D_y^E \hat{E}_z = \mu_{xx} \hat{\tilde{H}}_x \\ -D_x^E \hat{E}_z = \mu_{yy} \hat{\tilde{H}}_y \end{bmatrix}$$

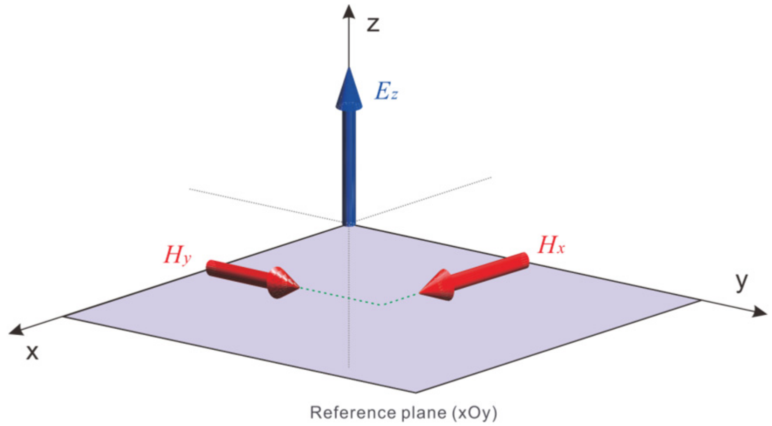
$$\mathbf{A}_E(\mathbf{x},\omega)\hat{\mathbf{E}}_z^s(\mathbf{x},\omega) = \hat{\mathbf{J}}^s(\mathbf{x},\omega)$$

$$\mathbf{A}_E = \mathbf{D}_x^H \mu_{yy}^{-1} \mathbf{D}_x^E + \mathbf{D}_y^H \mu_{xx}^{-1} \mathbf{D}_y^E + \varepsilon_{zz}$$

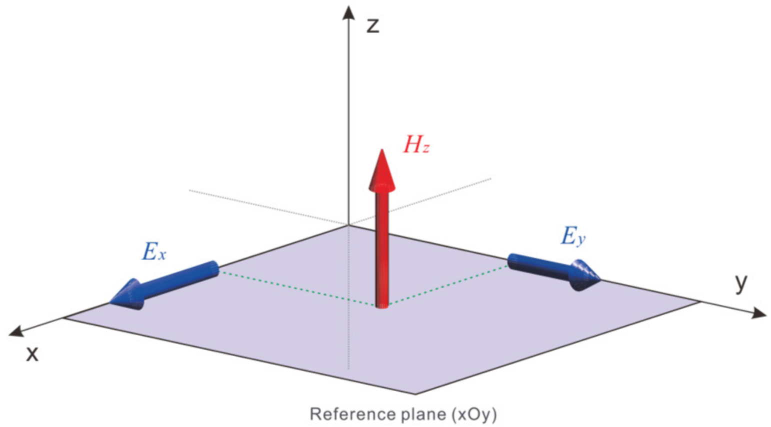
$$\begin{bmatrix} D_x^E \hat{E}_y - D_y^E \hat{E}_x = \mu_{zz} \hat{\tilde{H}}_z \\ D_y^H \hat{\tilde{H}}_z = \varepsilon_{xx} \hat{E}_x \\ -D_x^H \hat{\tilde{H}}_z = \varepsilon_{yy} \hat{E}_y \end{bmatrix}$$

$$\mathbf{A}_E(\mathbf{x},\omega)\hat{\mathbf{E}}_z^s(\mathbf{x},\omega) = \hat{\mathbf{J}}^s(\mathbf{x},\omega)$$

$$\mathbf{A}_E = \mathbf{D}_x^H \mu_{yy}^{-1} \mathbf{D}_x^E + \mathbf{D}_y^H \mu_{xx}^{-1} \mathbf{D}_y^E + \varepsilon_{zz}$$

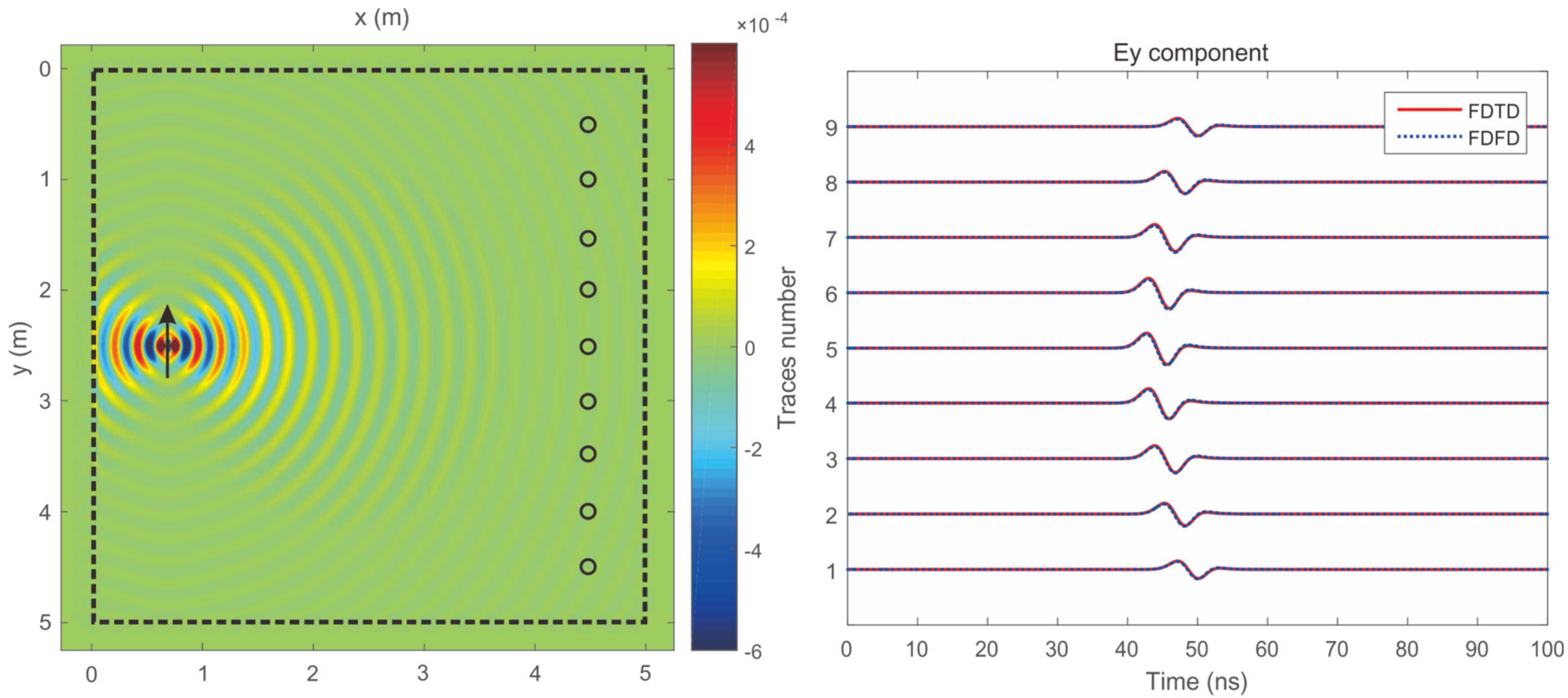


(a) TM mode



(b) TE mode

FD



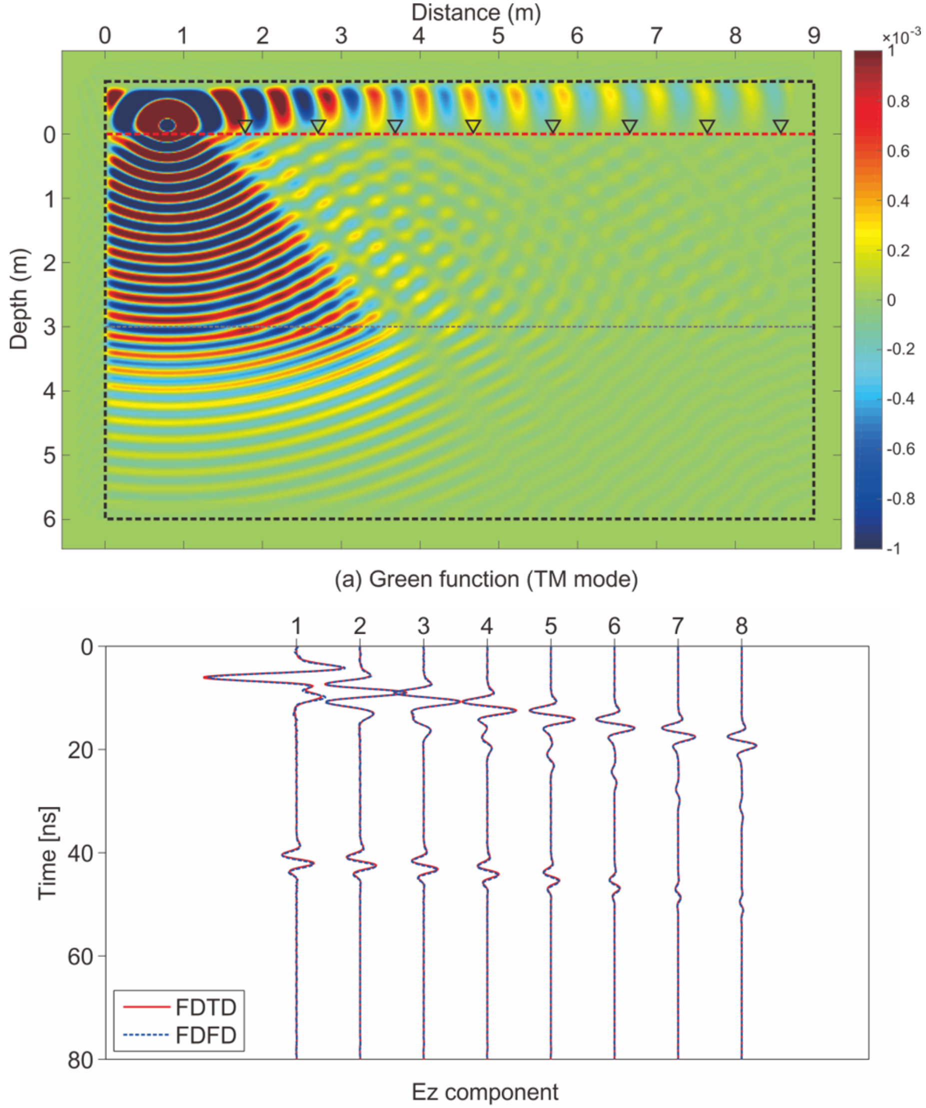
(a) Green function (TE mode)

(b) Traces cross-validation of FDTD and FDFD

A cross-validation of FDFD and FDTD simulation for a crosshole GPR setup (TE mode). (a) The frequency-domain numerical wavefield. The dashed black line indicates the interface between simulation domain and PML. (b) Comparisons of the radargrams between FDTD results and inverse Fourier transformed FDFD results.

FD

A cross-validation of FDFD and FDTD simulation for a on-ground GPR setup (TM mode). (a) The frequency-domain numerical wavefield. The dashed red line indicates the interface between free air and the Earth surface. The dashed gray line indicates the depth of the layer. The dashed black line indicates the interface of PML. (b) Comparison of the radargrams between FDTD results and inverse Fourier transformed FDFD results for the one-layer synthetic model.



(a) Green function (TM mode)

(b) Traces cross-validation of FDTD and FDFD

DG

Maxwell

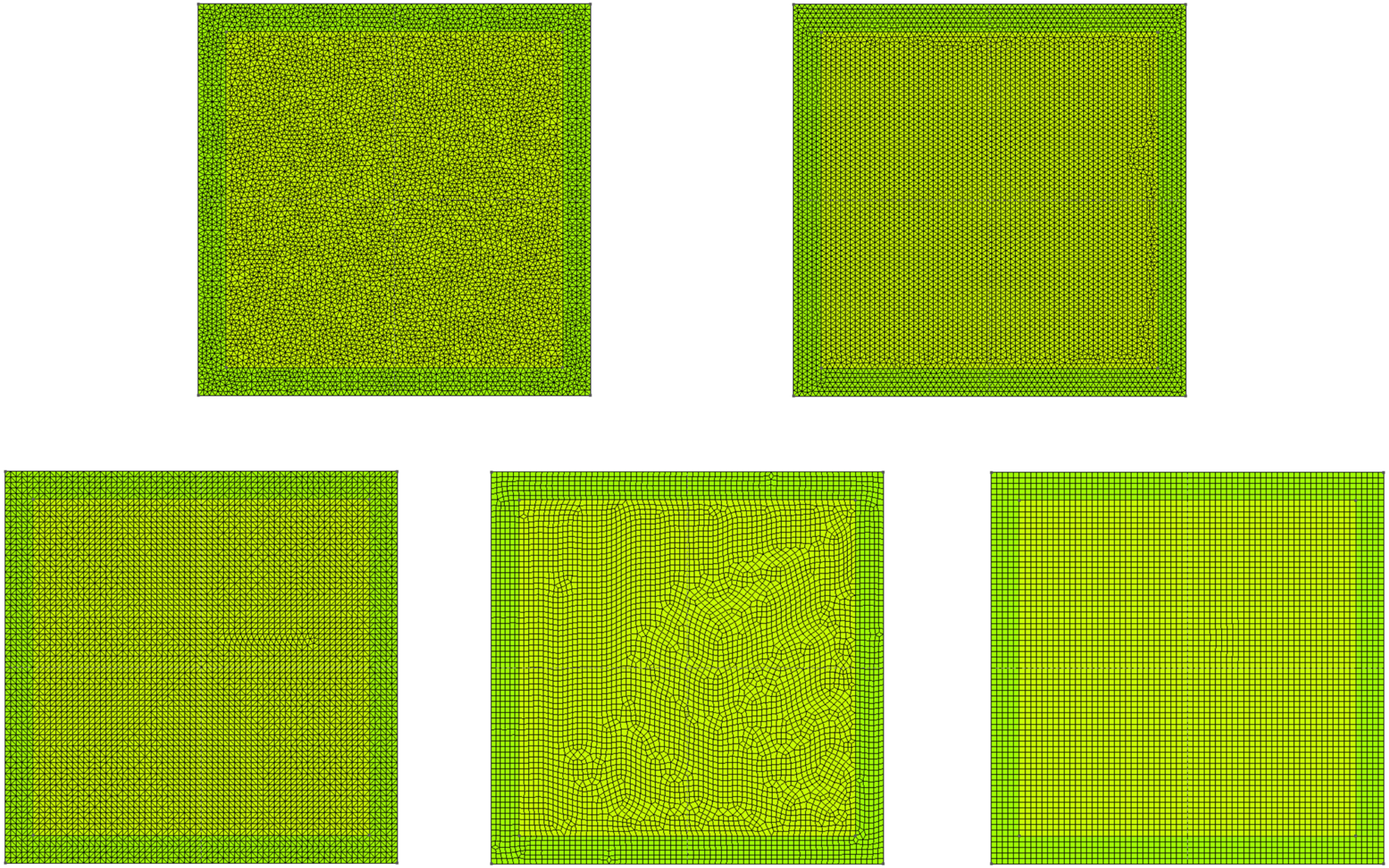
$$\begin{cases} \varepsilon \dfrac{\partial \mathrm{E}}{\partial t} - \nabla \times \mathrm{H} + \sigma \mathrm{E} = -\mathrm{J}^s \\ \mu \dfrac{\partial \mathrm{H}}{\partial t} + \nabla \times \mathrm{E} = 0 \\ \\ \dfrac{\partial \mathrm{D}}{\partial t} - \nabla \times \mu^{-1} \mathrm{B} + \dfrac{\sigma}{\varepsilon} \mathrm{D} = -\mathrm{J}^s \\ \dfrac{\partial \mathrm{B}}{\partial t} + \nabla \times \varepsilon^{-1} \mathrm{D} = 0 \end{cases}$$

$$\frac{\partial u}{\partial t} + \frac{\partial\left(\mathbf{A}^{(x)} u\right)}{\partial x} + \frac{\partial\left(\mathbf{A}^{(y)} u\right)}{\partial y} + \frac{\partial\left(\mathbf{A}^{(z)} u\right)}{\partial z} = j$$

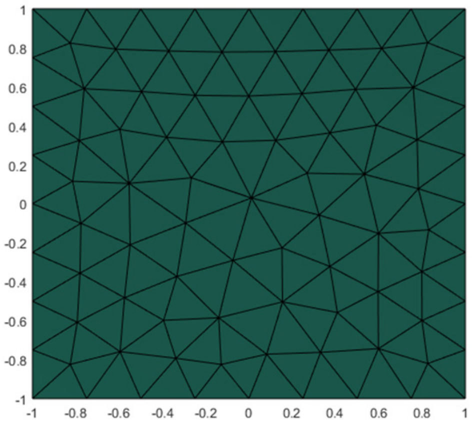
where

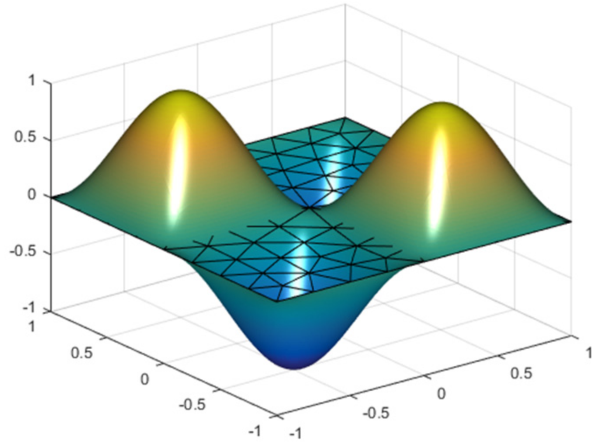
$$\begin{cases} \mathbf{A}^{(x)} = \begin{pmatrix} 0 & -\mu^{-1}\mathbf{C}\left(\hat{\mathrm{e}}^{(x)}\right) \\ \varepsilon^{-1}\mathbf{C}\left(\hat{\mathrm{e}}^{(x)}\right) & 0 \end{pmatrix}, \\ \mathbf{A}^{(y)} = \begin{pmatrix} 0 & -\mu^{-1}\mathbf{C}\left(\hat{\mathrm{e}}^{(y)}\right) \\ \varepsilon^{-1}\mathbf{C}\left(\hat{\mathrm{e}}^{(y)}\right) & 0 \end{pmatrix}, \\ \mathbf{A}^{(z)} = \begin{pmatrix} 0 & -\mu^{-1}\mathbf{C}\left(\hat{\mathrm{e}}^{(z)}\right) \\ \varepsilon^{-1}\mathbf{C}\left(\hat{\mathrm{e}}^{(z)}\right) & 0 \end{pmatrix}, \\ j = \begin{pmatrix} -\mathrm{J}^s \\ 0 \end{pmatrix} \qquad u = \begin{pmatrix} \mathrm{D} \\ \mathrm{B} \end{pmatrix} \\ \mathrm{C}(r) = \begin{pmatrix} 0 & -r_z & r_y \\ r_z & 0 & -r_x \\ -r_y & r_x & 0 \end{pmatrix} \end{cases}$$

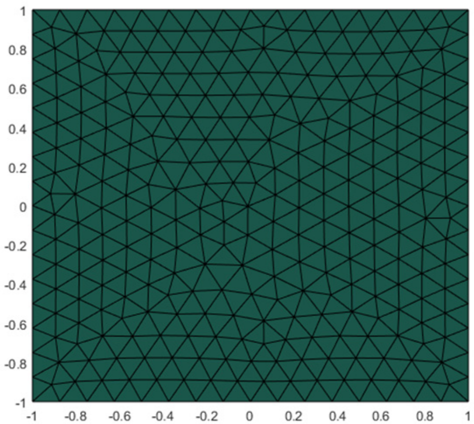
DG

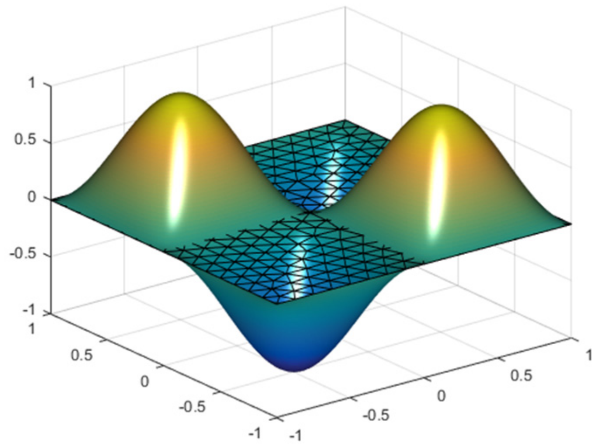


DG


1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-1 -0.8 -0.6 -0.4 -0.2 0 0.2 0.4 0.6 0.8 1


1
0.5
0
-0.5
-1
1 0.5 0 -0.5 -1
-1 -0.5 0 0.5 1


1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-1 -0.8 -0.6 -0.4 -0.2 0 0.2 0.4 0.6 0.8 1


1
0.5
0
-0.5
-1
1 0.5 0 -0.5 -1
-1 -0.5 0 0.5 1

Time domain vs. Frequency domain

	Time-domain	Frequency-domain
Advantage	• Intuitional (follow the GPR measurement signal) • Flexibility in the data processing • Parallel fashion by source points	• Benefit from the capability of direct solvers (multiple sources) • take benefit of the data redundancy (not the case of GPR) • Better for dealing with dispersive materials • Removal of time dependence • Parallel fashion by frequency
Disadvantage	• More and more frequency-domain measurements • Hard vs. soft sources • The order of the time-derivative • Hard to achieve the near-to-far field transformation • Accumulated error (time step)	• Not efficient (3D problem) • Very little flexibility in the data processing

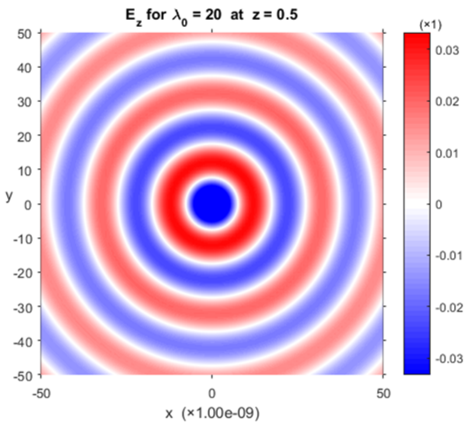
Prerequisite: the electromagnetic equations are linear.

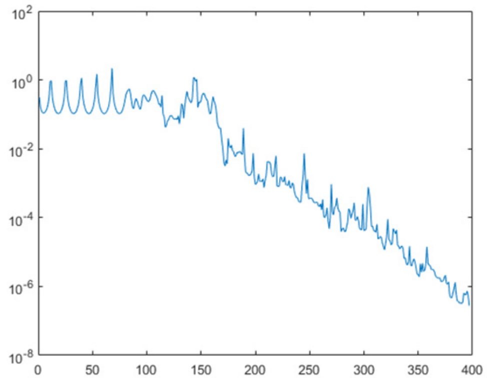
Direct solver vs. Iterative solver

Direct solvers	Iterative solvers

• For sparse matrix
• Intuitional
• Easy coding
• Very robust
• Work grows as $O\left(N^{1+\frac{2(d-1)}{d}}\right)$, i.e.,
  • $O(N^2)$ in 2D
  • $O\left(N^{7/3}\right)$ in 3D
• Memory grows as $O\left(N^{1+\frac{(d-1)}{d}}\right)$, i.e.,
  • $O\left(N^{3/2}\right)$ in 2D
  • $O\left(N^{5/3}\right)$ in 3D
	• For dense matrix
	• Efficiency
	• Can solve very large problems
	• Depending on solver and preconditioner type
	• Work can be $O(N)$
	• Memory is typically linear, $O(N)$
	• The final guess does not solve *Ax=b* exactly

Iterative solver



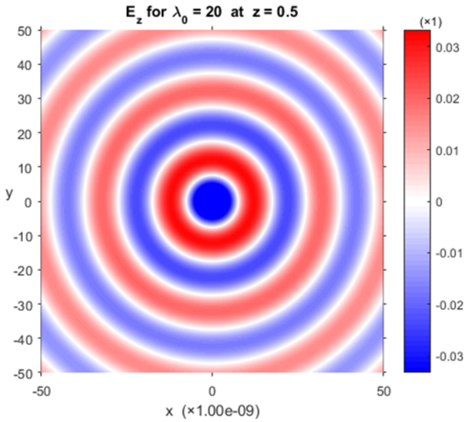


Iteration number
= 396

Relative residual error
= 8.811031e-07

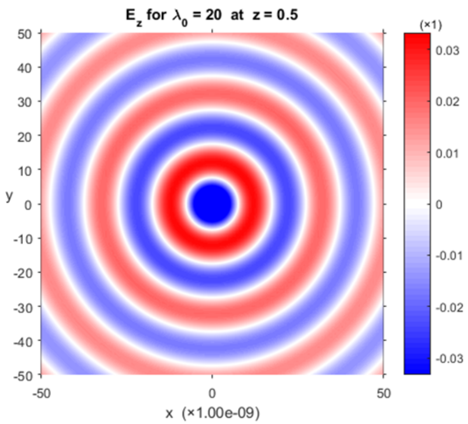
Function Name	Calls	Total Time	Self Time*	Allocated Memory	Freed Memory	Self Memory	Peak Memory	Total Time Plot (dark band = self time)
pointsrc_2d	1	5.793 s	0.174 s	55932.00 Kb	18780.00 Kb	-180.00 Kb	9028.00 Kb	
maxwell_run	1	3.572 s	0.082 s	26832.00 Kb	18596.00 Kb	180.00 Kb	9028.00 Kb	
vis2d	1	1.952 s	0.038 s	29096.00 Kb	0.00 Kb	0.00 Kb	5512.00 Kb	
solve_eq_iterative	1	1.924 s	0.037 s	26128.00 Kb	16864.00 Kb	-8016.00 Kb	9028.00 Kb	
bicg	1	1.593 s	1.140 s	8512.00 Kb	7832.00 Kb	0.00 Kb	7832.00 Kb	
Painter2d>Painter2d.init_display	1	1.084 s	0.016 s	6608.00 Kb	0.00 Kb	0.00 Kb	832.00 Kb	
build_system	1	1.051 s	0.115 s	0.00 Kb	1028.00 Kb	0.00 Kb	0.00 Kb	
Enumerated>Enumerated.subsindex	5566	0.860 s	0.039 s	304.00 Kb	356.00 Kb	0.00 Kb	144.00 Kb	
Enumerated>Enumerated.int	5747	0.857 s	0.585 s	304.00 Kb	356.00 Kb	-340.00 Kb	144.00 Kb	
Painter2d>Painter2d.draw_objsrc	1	0.668 s	0.041 s	21640.00 Kb	0.00 Kb	1380.00 Kb	5512.00 Kb	

Direct solver



Function Name	Calls	Total Time	Self Time*	Allocated Memory	Freed Memory	Self Memory	Peak Memory	Total Time Plot (dark band = self time)
pointsrc_2d	1	5.653 s	0.252 s	123512.00 Kb	9892.00 Kb	620.00 Kb	8348.00 Kb	
vis2d	1	2.848 s	0.053 s	70432.00 Kb	0.00 Kb	0.00 Kb	8220.00 Kb	
maxwell_run	1	2.430 s	0.093 s	51820.00 Kb	9712.00 Kb	840.00 Kb	8348.00 Kb	
Painter2d>Painter2d.init_display	1	1.811 s	0.398 s	35564.00 Kb	0.00 Kb	5764.00 Kb	7820.00 Kb	
build_system	1	1.167 s	0.147 s	17872.00 Kb	0.00 Kb	1900.00 Kb	1024.00 Kb	
Enumerated>Enumerated.subsindex	5566	0.849 s	0.038 s	604.00 Kb	0.00 Kb	0.00 Kb	128.00 Kb	
Enumerated>Enumerated.int	5747	0.848 s	0.581 s	604.00 Kb	0.00 Kb	324.00 Kb	128.00 Kb	
Painter2d>Painter2d.draw_objsrc	1	0.797 s	0.049 s	33972.00 Kb	0.00 Kb	1572.00 Kb	8220.00 Kb	
solve_eq_direct	1	0.718 s	0.375 s	32880.00 Kb	8568.00 Kb	-2592.00 Kb	8348.00 Kb	
ezplot	1	0.646 s	0.014 s	30040.00 Kb	0.00 Kb	64.00 Kb	8220.00 Kb	

Direct solver



1s faster !

Function Name	Calls	**Total Time**	Self Time*	Allocated Memory	Freed Memory	Self Memory	Peak Memory	Total Time Plot (dark band = self time)
pointsrc_2d	1	5.653 s	0.252 s	123512.00 Kb	9892.00 Kb	620.00 Kb	8348.00 Kb	
vis2d	1	2.848 s	0.053 s	70432.00 Kb	0.00 Kb	0.00 Kb	8220.00 Kb	
maxwell_run	1	2.430 s	0.093 s	51820.00 Kb	9712.00 Kb	840.00 Kb	8348.00 Kb	
Painter2d>Painter2d.init_display	1	1.811 s	0.398 s	35564.00 Kb	0.00 Kb	5764.00 Kb	7820.00 Kb	
build_system	1	1.167 s	0.147 s	17872.00 Kb	0.00 Kb	1900.00 Kb	1024.00 Kb	
Enumerated>Enumerated.subsindex	5566	0.849 s	0.038 s	604.00 Kb	0.00 Kb	0.00 Kb	128.00 Kb	
Enumerated>Enumerated.int	5747	0.848 s	0.581 s	604.00 Kb	0.00 Kb	324.00 Kb	128.00 Kb	
Painter2d>Painter2d.draw_objsrc	1	0.797 s	0.049 s	33972.00 Kb	0.00 Kb	1572.00 Kb	8220.00 Kb	
solve_eq_direct	1	0.718 s	0.375 s	32880.00 Kb	8568.00 Kb	-2592.00 Kb	8348.00 Kb	
ezplot	1	0.646 s	0.014 s	30040.00 Kb	0.00 Kb	64.00 Kb	8220.00 Kb	

Direct solvers

SPARSE DIRECT SOLVERS	License	Support	Type		Language			Mode			Dense	Sparse Direct			Sparse Iterative		Sparse Eigenvalue		Last release date
			Real	Complex	F77/ F95	C	C++	Shared	Accel.	Dist		SPD	SI	Gen	SPD	Gen	Sym	Gen	
DSCPACK	PD	yes	X			X		X		M		X							?
KKTDirect	PD	yes	X			X	X	X				LDL^T							2010-04-21
MUMPS	CeCILL-C	yes	X	X	X	X		X		M		X	X	X					2015-02-20
PaStiX	CeCILL-C	yes	X	X	X	X	X	X	C	M		X	X	X	X	X			2014-12-18
PSPASES	Own	yes	X		X	X				M		X							1999-05-09
QR-MUMPS	LGPL	yes	X	X	X	X		X						X					2012-08-01
Quern	PD	yes	X			X	X	X						X		X			2009-02-04
SPARSE	Own	?	X	X		X		X				X		X					1988-04-01
SPOOLES	PD	?	X	X		X		X		M				X		X			1999-04-08
SPRAL	New BSD	yes	X	X	X	X		X	C			X	X				X		2015-04-20
SuiteSparse	LGPL	yes	X	X		X		X	C			X		X					2015-03-24
SuperLU	Own	yes	X	X	X	X		X	C	M				X					2015-05-01
TAUCS	Own	yes	X	X		X		X				X		X	X	X			2003-09-04
Trilinos/Amesos	LGPL	yes	X			X	X	X		M		X		X					2015-05-07
Trilinos/Amesos2	BSD	yes	X	X			X	X		M		X		X					2015-05-07
Y12M	?	yes	X		X			X				X		X					?

Iterative solvers

SPARSE ITERATIVE SOLVERS	License	Support	Type		Language			Mode			Dense	Sparse Direct			Sparse Iterative		Sparse Eigenvalue		Last release date
			Real	Complex	F77/ F95	C	C++	Shared	Accel.	Dist		SPD	SI	Gen	SPD	Gen	Sym	Gen	
BDDCML	LGPL	yes	X		X			X		M					X	X			2014-03-12
BILUM	Own	yes	X		X			X							X	X			1998-03-18
BlockSolve95	Own	?	X		X	X	X			M					X	X			1997-07-08
CERFACS	?	yes	X	X	X			X							X	X			2007-07-01
DUNE / ISTL	GPL	yes	X	X			X	X		M					X	X			2014-12-18
GMM++	LGPL	yes	X	X			X	X				X		X	X	X			2014-08-21
HIPS	CeCILL-C	yes	X	X	X	X	X	X		M					X	X			2010-10-13
HYPRE	LGPL	yes	X		X	X		X		M					P	P	P		2015-01-22
IML++	PD	?	X				X	X							X	X			1995-01-05
ITL	Own	yes	X				X	X							X	X			2001-10-26
ITPACK	PD	?	X		X			X							X	X			1989-05-01
ITSOL	GPL	yes	X		X			X								X			2012-10-25
Lis	BSD	yes	X		X	X		X		M					P	P	P	P	2015-05-14
PARALUTION	GPL	yes	X				X	X	C/O						X	X			2015-02-27
pARMS	LGPL	yes	X		X	X		X		M					X	X			2011-01-14
PETSc	Own	yes	X	X	X	X		X	C/O	M					P	P			2015-01-31
PIM	Own	yes	X	X	X			X		M					X	X			2003-06-20
QMRPACK (tar.gz)	Own	?	X	X	X			X							X	X	X	X	1996-04-15
SLAP	PD	?	X		X										X	X			1998-07-21
SOL	CPL or BSD	yes	X		X	X		X							X	X			2015-05-14
SPARSKIT	LGPL	yes	X		X			X								X			2009-11-18
SPLIB (tar.gz)	Own	?	X		X			X							X	X			1999-04-01
Templates	BSD	yes	X		X	X		X							X	X			1998-07-21
Trilinos/AztecOO	BSD	yes	X		X	X	X	X		M					X	X			2015-05-07
Trilinos/Belos	BSD	yes	X	X			X	X		M					X	X			2015-05-07
Trilinos/Komplex	BSD	yes		X		X	X	X		M					X	X			2015-05-07

http://www.netlib.org/utk/people/JackDongarra/la-sw.html

Ignored details

- explicit/implicit form
- upwind/central (difference/flux)
- weak/strong formulation
- nodal/modal basic function
- noise
- pre-/post-process of the data

Thanks for your attention!

Notes

- First to do: analyze the stability and grid dispersion.
- Best validation: compare with analytic results.