Amélioration des performances de méthodes Galerkin discontinues d'ordre élevé pour la résolution numérique des équations de Maxwell instationnaires sur des maillages simplexes

Joseph Charles

INRIA Sophia Antipolis-Méditerranée
Nachos project-team
`joseph.charles@inria.fr`

Université de Nice - Sophia Antipolis - UFR Sciences
Ecole doctorale EDSFA
Sciences fondamentales et appliquées
April 26, 2012

Research grant between CEA DAM, CESTA center and Nachos project-team (2008-2011)

Outline

1. Time-domain electromagnetics
   - Overview of existing methods
   - A non-dissipative DGTD-$\mathbb{P}_{p_i}$ method
2. Numerical study of various polynomial interpolation methods
   - Numerical results in 1D
   - Design of hierarchical basis functions
   - Numerical results in 2D
3. Local time step high order DGTD method
   - Formulation
   - Algorithmic aspect
   - Numerical results in 1D
   - Numerical results in 2D
4. DGTD solver on GPU based systems
   - Performance results in 3D
5. Perspectives

Outline

1. Time-domain electromagnetics
   - Overview of existing methods
   - A non-dissipative DGTD-$\mathbb{P}_{p_i}$ method
2. Numerical study of various polynomial interpolation methods
   - Numerical results in 1D
   - Design of hierarchical basis functions
   - Numerical results in 2D
3. Local time step high order DGTD method
   - Formulation
   - Algorithmic aspect
   - Numerical results in 1D
   - Numerical results in 2D
4. DGTD solver on GPU based systems
   - Performance results in 3D
5. Perspectives

Time-domain electromagnetics

Boundary and initial value problem

Maxwell equations, $\mathbf{x} \in \Omega$, $t > 0$

$$\begin{cases} \varepsilon \partial_t \mathbf{E} - \nabla \times \mathbf{H} = 0 \\ \\ \mu \partial_t \mathbf{H} + \nabla \times \mathbf{E} = 0 \end{cases}$$

$$\mathbf{E} = \mathbf{E}(\mathbf{x}, t) \quad \text{and} \quad \mathbf{H} = \mathbf{H}(\mathbf{x}, t)$$

Boundary conditions: $\partial\Omega = \Gamma_a \cup \Gamma_m$

$$\begin{cases} \mathbf{n} \times \mathbf{E} = 0 \text{ on } \Gamma_m \\ \mathbf{n} \times \mathbf{E} - \sqrt{\frac{\mu}{\varepsilon}} \mathbf{n} \times (\mathbf{H} \times \mathbf{n}) = \mathbf{n} \times \mathbf{E}_{\text{inc}} - \sqrt{\frac{\mu}{\varepsilon}} \mathbf{n} \times (\mathbf{H}_{\text{inc}} \times \mathbf{n}) \text{ on } \Gamma_a \end{cases}$$

Initial conditions

$$\mathbf{E}_0 = \mathbf{E}(\mathbf{x}, 0) \quad \text{and} \quad \mathbf{H}_0 = \mathbf{H}(\mathbf{x}, 0)$$

FDTD: Finite Difference Time-Domain method

- Seminal work of K.S. Yee
  (IEEE Trans. Antennas Propag., Vol. AP-14, 1966)
- Structured (cartesian) meshes
- Second order accurate (space and time) on uniform meshes
- Advantages
  - Easy computer implementation
  - Computationally efficient (very low algorithmic complexity)
  - Mesh generation is straightforward
  - Modelization of complex sources (antennas, thin wires, etc.) is well established
- Drawbacks
  - Accuracy on non-uniform discretizations
  - Memory requirements for high resolution models
  - Approximate discretization of boundaries (stair case representation)

FETD: Finite Element Time-Domain method

- Often based on J.-C. Nédélec edge elements (Numer. Math, Vol. 35, 1980 and Vol. 50, 1986)
  - Unstructured meshes
  - Advantages
    - Accurate representation of complex shapes
    - Well suited to high order interpolation methods
  - Drawbacks
    - Computer implementation is less trivial
    - Unstructured mesh generation is hardly automated
    - Global mass matrix
  - Mass lumped FETD methods
    - S. Pernet, X. Ferrieres and G. Cohen
      IEEE Trans. Antennas Propag., Vol. 53, No. 9, 2005
    - Hexahedral meshes, high order Lagrange polynomials
    - Leap-frog time integration scheme

FVTD: Finite Volume Time-Domain method

- Imported from the computational fluid dynamics (CFD) community
  - V. Shankar, W. Hall and A. Mohammadian
    Electromag. Vol. 10, 1990
  - J.-P. Cioni, L. Fezoui and H. Steve
    IMPACT Comput. Sci. Eng., Vol. 5, No. 3, 1993
  - P. Bonnet, X. Ferrieres *et al.*
    J. Electromag. Waves and Appl., Vol. 11, 1997
  - S. Piperno and M. Remaki and L. Fezoui
    SIAM J. Num. Anal., Vol. 39, No. 6, 2002.
- Unstructured meshes
- Uknowns are cell averages of the field components
- Flux evaluation at cell interfaces
  - Upwind scheme → numerical dissipation
  - Centered scheme → numerical dispersion (on non-uniform meshes)
- Extension to higher order accuracy: MUSCL technique

DGTD: Discontinuous Galerkin Time-Domain method

Advantages of a DG-FEM method

- High order of accuracy
- Local definition of interpolation order (*p*-adaptivity)
- Handle complicated geometries and unstructured meshes (*h*-adaptivity)
- Yield block diagonal mass matrices when coupled to explicit time integration schemes
- No interelement continuity requirement (use of a numerical flux)
- Amenable to efficient parallelization

Reference text book

J.S. Hesthaven and T. Warburton
Nodal discontinuous Galerkin methods: algorithms, analysis, and applications
Springer, 2008

- F. Bourdel, P.A. Mazet and P. Helluy
  Proc. 10th Inter. Conf. on Comp. Meth. in Appl. Sc. and Eng., 1992.
    - Triangular meshes, first-order upwind DG method (i.e FV method)
    - Time-domain and time-harmonic Maxwell equations
- M. Remaki and L. Fezoui, INRIA RR-3501, 1998.
    - Time-domain Maxwell equations
    - Triangular meshes, P1 interpolation, Runke-Kutta time integration (RKDG)
- J.S. Hesthaven and T. Warburton (J. Comput. Phys., Vol. 181, 2002)
    - Tetrahedral meshes, high order Lagrange polynomials, upwind flux
    - Runge-Kutta time integration
- B. Cockburn, F. Li and C.-W. Shu (J. Comput. Phys., Vol. 194, 2004)
    - Locally divergence-free discontinuous Galerkin formulation
- G. Cohen, X. Ferrieres and S. Pernet (J. Comput. Phys., Vol. 217, 2006)
    - Hexahedral meshes, high order Lagrange polynomials, penalized formulation
    - Leap-frog time integration scheme
- And a steadily increasing number of other works since 2005

- Discretization of Ω: $\overline{\Omega}_h \equiv \mathcal{T}_h = \bigcup_{\tau_i \in \mathcal{T}_h} \overline{\tau}_i$
  - $\mathcal{F}_0$: set of purely internal faces
  - $\mathcal{F}_m$ and $\mathcal{F}_a$: sets of faces on the boundaries Γ_m and Γ_a
- Approximation space: $V_h = \{\mathbf{V}_h \in L^2(\Omega)^3 \mid \forall i, \mathbf{V}_{h|\tau_i} \equiv \mathbf{V}_i \in \mathbb{P}_{p_i}[\tau_i]^3\}$
- Variational formulation: $\forall \vec{\varphi} \in \mathcal{P}_i = \mathrm{Span}(\vec{\varphi}_{ij}\ ,\ 1 \leq j \leq d_i)$

$$\left\{ \begin{array}{rcl} \displaystyle\iiint_{\tau_i} \vec{\varphi} \cdot \varepsilon_i \partial_t \mathbf{E} d\omega & = & \displaystyle -\iint_{\partial\tau_i} \vec{\varphi} \cdot (\mathbf{H} \times \vec{n}) ds + \iiint_{\tau_i} \nabla \times \vec{\varphi} \cdot \mathbf{H} d\omega \\ \displaystyle\iiint_{\tau_i} \vec{\varphi} \cdot \mu_i \partial_t \mathbf{H} d\omega & = & \displaystyle \iint_{\partial\tau_i} \vec{\varphi} \cdot (\mathbf{E} \times \vec{n}) ds - \iiint_{\tau_i} \nabla \times \vec{\varphi} \cdot \mathbf{E} d\omega \end{array} \right.$$

- Approximate fields: $\forall i$, $\mathbf{E}_{h|\tau_i} \equiv \mathbf{E}_i$ and $\mathbf{H}_{h|\tau_i} \equiv \mathbf{H}_i$
- Integral over $\partial\tau_i$: $\mathbf{E}_{|a_{ik}} = \dfrac{\mathbf{E}_i + \mathbf{E}_k}{2}$ and $\mathbf{H}_{|a_{ik}} = \dfrac{\mathbf{H}_i + \mathbf{H}_k}{2}$
- Assume $\Gamma_a = \emptyset$ (to simplify the presentation)
  and on Γ_m: $\mathbf{E}_{k|a_{ik}} = -\mathbf{E}_{i|a_{ik}}$ and $\mathbf{H}_{k|a_{ik}} = \mathbf{H}_{i|a_{ik}}$

$$\left\{\begin{array}{rcl}
\displaystyle\iiint\limits_{\tau_i} \vec{\varphi} \cdot \varepsilon_i \partial_t \mathbf{E}_i d\omega & = & \displaystyle\frac{1}{2} \iiint\limits_{\tau_i} (\nabla \times \vec{\varphi} \cdot \mathbf{H}_i + \nabla \times \mathbf{H}_i \cdot \vec{\varphi}) d\omega \\
 & - & \displaystyle\frac{1}{2} \sum_{k \in \mathcal{V}_i} \iint\limits_{a_{ik}} \vec{\varphi} \cdot (\mathbf{H}_k \times \vec{n}_{ik}) ds \\
\displaystyle\iiint\limits_{\tau_i} \vec{\varphi} \cdot \mu_i \partial_t \mathbf{H}_i d\omega & = & \displaystyle -\frac{1}{2} \iiint\limits_{\tau_i} (\nabla \times \vec{\varphi} \cdot \mathbf{E}_i + \nabla \times \mathbf{E}_i \cdot \vec{\varphi}) d\omega \\
 & + & \displaystyle\frac{1}{2} \sum_{k \in \mathcal{V}_i} \iint\limits_{a_{ik}} \vec{\varphi} \cdot (\mathbf{E}_k \times \vec{n}_{ik}) ds
\end{array}\right.$$

Time-domain electromagnetics

A non-dissipative DGTD-$\mathbb{P}_{p_i}$ method

- Local projections

$$\mathbf{E}_i(\mathbf{x}) = \sum_{1\leq j\leq d_i} E_{ij}\vec{\varphi}_{ij}(\mathbf{x}) \quad \text{and} \quad \mathbf{H}_i(\mathbf{x}) = \sum_{1\leq j\leq d_i} H_{ij}\vec{\varphi}_{ij}(\mathbf{x})$$

- $\vec{\varphi}_{ij}$: Lagrange (nodal) polynomials
- Vector representation of local fields

$$\mathbb{E}_i = \{E_{ij}\}_{1\leq j\leq d_i} \text{ and } \mathbb{H}_i = \{H_{ij}\}_{1\leq j\leq d_i}$$

- For $1 \leq j, l \leq d_i$:
  - $(\mathbf{M}_i^{\varepsilon})_{jl} = \varepsilon_i \iiint_{\tau_i} {}^{\mathbf{T}}\vec{\varphi}_{ij}\vec{\varphi}_{jl}d\omega$ and $(\mathbf{M}_i^{\mu})_{jl} = \mu_i \iiint_{\tau_i} {}^{\mathbf{T}}\vec{\varphi}_{ij}\vec{\varphi}_{jl}d\omega$
  - $(\mathbf{K}_i)_{jl} = \dfrac{1}{2}\iiint_{\tau_i}({}^{\mathbf{T}}\vec{\varphi}_{ij}\nabla\times\vec{\varphi}_{il} + {}^{\mathbf{T}}\vec{\varphi}_{il}\nabla\times\vec{\varphi}_{ij})d\omega$
- For $1 \leq j \leq d_i$ and $1 \leq l \leq d_k$
  - $(\mathbf{S}_{ik})_{jl} = \dfrac{1}{2}\iint_{a_{ik}} {}^{\mathbf{T}}\vec{\varphi}_{ij}(\vec{\varphi}_{kl}\times\vec{n}_{ij})ds$

Local EDO systems

$$\forall \tau_i \quad : \quad \left\{ \begin{array}{rcrcl} \mathbf{M}_i^{\varepsilon} \dfrac{d\mathbb{E}_i}{dt} & = & \mathbf{K}_i \mathbb{H}_i & - & \displaystyle\sum_{k \in \mathcal{V}_i} \mathbf{S}_{ik} \mathbb{H}_k \\ \mathbf{M}_i^{\mu} \dfrac{d\mathbb{H}_i}{dt} & = & -\mathbf{K}_i \mathbb{E}_i & + & \displaystyle\sum_{k \in \mathcal{V}_i} \mathbf{S}_{ik} \mathbb{E}_k \end{array} \right.$$

Global EDO system (with $d = \sum_i d_i$)

$$\mathbf{M}^{\varepsilon} \frac{d\mathbb{E}}{dt} = \mathbf{S}\mathbb{H} \quad \text{and} \quad \mathbf{M}^{\mu} \frac{d\mathbb{H}}{dt} = -{}^{\mathbf{T}}\mathbf{S}\mathbb{E}$$

- $\mathbf{S} = \mathbf{K} - \mathbf{A} - \mathbf{B}$
- $\mathbf{M}^{\varepsilon}$ are $\mathbf{M}^{\mu}$ block diagonal symmetric definite positive matrices
- $\mathbf{K}$ is a $d \times d$ block diagonal symmetric matrix
- $\mathbf{A}$ is a $d \times d$ block sparse symmetric matrix (internal faces)
- $\mathbf{B}$ is a $d \times d$ block sparse skew symmetric matrix (metallic faces)

Leap-frog based explicit time integration

- L. Fezoui, S. Lanteri, S. Lohrengel and S. Piperno
  ESAIM: M2AN, Vol. 39, No. 6, 2005

$$\left\{\begin{array}{lcl} \mathbf{M}^{\varepsilon}\left(\dfrac{\mathbb{E}^{n+1}-\mathbb{E}^{n}}{\Delta t}\right) & = & \mathbf{S}\mathbb{H}^{n+\frac{1}{2}} \\ \mathbf{M}^{\mu}\left(\dfrac{\mathbb{H}^{n+\frac{1}{2}}-\mathbb{H}^{n-\frac{1}{2}}}{\Delta t}\right) & = & -{}^{\mathrm{T}}\mathbf{S}\mathbb{E}^{n+1} \end{array}\right.$$

Stability analysis

- Discrete electromagnetic energy
  $$\mathcal{E}^n = {}^{\mathrm{T}}\mathbb{E}^n\mathbf{M}^{\varepsilon}\mathbb{E}^n + {}^{\mathrm{T}}\mathbb{H}^{n+\frac{1}{2}}\mathbf{M}^{\mu}\mathbb{H}^{n-\frac{1}{2}}$$
- Condition for $\mathcal{E}^n$ being a positive definite form
  $$\Delta t \leq \frac{2}{d_2}, \quad \text{with} \quad d_2 = \| \, (\mathbf{M}^{-\mu})^{\frac{1}{2}} \; {}^{\mathrm{T}}\mathbf{S} \; (\mathbf{M}^{-\varepsilon})^{\frac{1}{2}} \, \|$$

Outline

1. Time-domain electromagnetics
   - Overview of existing methods
   - A non-dissipative DGTD-$\mathbb{P}_{p_i}$ method
2. Numerical study of various polynomial interpolation methods
   - Numerical results in 1D
   - Design of hierarchical basis functions
   - Numerical results in 2D
3. Local time step high order DGTD method
   - Formulation
   - Algorithmic aspect
   - Numerical results in 1D
   - Numerical results in 2D
4. DGTD solver on GPU based systems
   - Performance results in 3D
5. Perspectives

Objectives of the study

- Assessment of various polynomial interpolation methods in 1D and 2D
- Use of high order explicit time integration methods:
  - Second order leap-frog
  - Fourth order leap-frog
  - Fourth order Runge-Kutta

Context of the study

Investigate the hierarchical nature of polynomial basis functions towards the future development of a *hp*-adaptive DGTD-$\mathbb{P}_{p_i}$ methodology

Time-domain Maxwell equations on a one-dimensional domain

$$\begin{cases} \varepsilon \dfrac{\partial \mathbf{E}}{\partial t} - \dfrac{\partial \mathbf{H}}{\partial x} = & 0 \\ \\ \mu \dfrac{\partial \mathbf{H}}{\partial t} - \dfrac{\partial \mathbf{E}}{\partial x} = & 0 \end{cases}$$

$$\mathbf{E} = {}^t(0, 0, E_z) \text{ and } \mathbf{H} = {}^t(0, H_y, 0)$$

Key features

- Nodal or modal nature of the interpolation
- Hierarchical nature of the interpolation in view of *p*-adaptive method
- Efficient computation of surface integrals
- Condition number of the mass matrix

Candidate methods

- nodal basis: Lagrange
- modal basis: Bernstein
- modal hierarchical basis: Taylor, Canonique, Legendre

Numerical study of various polynomial interpolation methods

Numerical results in 1D: eigenmode in a unit interval with metallic boundaries

Eigenmode in a unit interval, LF4 scheme

Base	N = 11 points	N = 21 points	N = 41 points	Gain 11pts → 21pts	Gain 11pts → 41pts
L1	$2.93 \cdot 10^{-1}$	$7.23 \cdot 10^{-2}$	$1.94 \cdot 10^{-2}$	4.1	15.1
B1,G1,C1,T1	$2.95 \cdot 10^{-1}$	$7.24 \cdot 10^{-2}$	$1.94 \cdot 10^{-2}$	4.1	15.2
L2	$3.41 \cdot 10^{-4}$	$3.49 \cdot 10^{-5}$	$4.07 \cdot 10^{-6}$	9.8	83.8
B2,G2,C2,T2	$1.82 \cdot 10^{-4}$	$1.06 \cdot 10^{-5}$	$6.49 \cdot 10^{-7}$	17.2	280.4
L3	$3.93 \cdot 10^{-5}$	$2.65 \cdot 10^{-6}$	$2.89 \cdot 10^{-7}$	14.8	136.0
B3,G3,C3,T3	$2.94 \cdot 10^{-5}$	$2.65 \cdot 10^{-6}$	$2.90 \cdot 10^{-7}$	11.1	101.4
L4	$4.23 \cdot 10^{-6}$	$2.62 \cdot 10^{-7}$	$1.63 \cdot 10^{-8}$	16.2	259.5
B4,G4,C4,T4	$4.23 \cdot 10^{-6}$	$2.62 \cdot 10^{-7}$	$1.63 \cdot 10^{-8}$	16.2	259.5
L5	$1.09 \cdot 10^{-6}$	$6.79 \cdot 10^{-8}$	$4.26 \cdot 10^{-9}$	16.1	255.9
B5,G5,C5,T5	$1.09 \cdot 10^{-6}$	$6.79 \cdot 10^{-8}$	$4.26 \cdot 10^{-9}$	16.1	255.9

Eigenmode in a unit interval, $N = 41$pts

Base	Gain LF2 → LF4	Gain LF2 → RK4	Gain LF4 → RK4
L1	1.4	1.4	1.0
B1,G1,C1,T1	1.4	1.4	1.0
L2	486.5	0.2	$4 \cdot 10^{-4}$
B2,G2,C2,T2	3050.9	0.2	$7 \cdot 10^{-5}$
L3	2667.8	0.1	$5 \cdot 10^{-5}$
B3,G3,C3,T3	2658.6	0.1	$5 \cdot 10^{-5}$
L4	21042.9	0.1	$4 \cdot 10^{-6}$
B4,G4,C4,T4	21042.9	0.1	$4 \cdot 10^{-6}$
L5	39436.6	0.1	$2 \cdot 10^{-6}$
B5,G5,C5,T5	39436.6	0.1	$2 \cdot 10^{-6}$

Numerical study of various polynomial interpolation methods

Numerical results in 1D: triangular pulse in an interval [0, 4] with absorbing boundaries

Triangular pulse in a interval [0, 4], RK4 scheme

Base	N = 81 points	N = 121 points	N = 201 points	Gain 81pts → 121pts	Gain 81pts → 201pts
L1	$3.09 \cdot 10^{-2}$	$2.21 \cdot 10^{-2}$	$1.45 \cdot 10^{-2}$	1.4	2.1
B1,G1,C1,T1	$3.39 \cdot 10^{-2}$	$2.38 \cdot 10^{-2}$	$1.53 \cdot 10^{-2}$	1.4	2.2
L2	$1.73 \cdot 10^{-2}$	$9.64 \cdot 10^{-3}$	$4.29 \cdot 10^{-3}$	1.8	4.0
B2,G2,C2,T2	$1.65 \cdot 10^{-2}$	$8.77 \cdot 10^{-3}$	$4.07 \cdot 10^{-3}$	1.9	4.1
L3	$6.34 \cdot 10^{-3}$	$3.99 \cdot 10^{-3}$	$1.82 \cdot 10^{-3}$	1.6	3.5
B3,G3,C3,T3	$4.61 \cdot 10^{-3}$	$2.87 \cdot 10^{-3}$	$1.37 \cdot 10^{-3}$	1.6	3.4
L4	$3.71 \cdot 10^{-3}$	$2.51 \cdot 10^{-3}$	$1.10 \cdot 10^{-3}$	1.5	3.4
B4,G4,C4,T4	$2.11 \cdot 10^{-3}$	$1.46 \cdot 10^{-3}$	$7.42 \cdot 10^{-4}$	1.5	2.8
L5	$3.12 \cdot 10^{-3}$	$2.23 \cdot 10^{-3}$	$1.05 \cdot 10^{-3}$	1.4	3.0
B5,G5,C5,T5	$1.88 \cdot 10^{-3}$	$1.35 \cdot 10^{-3}$	$8.05 \cdot 10^{-4}$	1.4	2.3

Triangular pulse in an interval [0, 4], $N = 201$pts

Base	Gain LF2 → LF4	Gain LF2 → RK4	Gain LF4 → RK4
L1	1.4	2.2	1.5
B1,G1,C1,T1	1.5	2.2	1.5
L2	0.8	1.9	2.4
B2,G2,C2,T2	0.8	2.1	2.5
L3	0.9	2.9	3.1
B3,G3,C3,T3	0.9	3.6	4.1
L4	0.9	3.1	3.4
B4,G4,C4,T4	1.1	4.4	5.0
L5	1.0	2.3	2.4
B5,G5,C5,T5	0.9	2.8	3.0

Main related works

Several authors worked on hierarchical expansions for triangles and tetrahedra:

- P.Solin, K.Segeth and I.Dolezel
  Studies in Advanced Mathematics, Chapman & Hall/CRC Press, 2003
  Higher-order finite element methods
- G.E.Karniadakis and S.Sherwin
  Numerical Mathematics and Scientific Computation, Oxford Science Publications, 2005
  Spectral/*hp* element methods for computational fluid dynamics
- M.Ainsworth and J.Coyle
  International Journal for Numerical Methods in Engineering, Vol. 58, No. 14, 2003
  Hierarchic finite element bases on unstructured tetrahedral meshes
- S.Deng and W.Cai
  Applied Numerical Analysis & Computational Mathematics, Vol. 2, No. 3, 2005
  Analysis and application of an orthogonal nodal basis on triangles for discontinuous spectral element methods

Vertex functions

$$\left\{\begin{array}{lcl} \varphi_t^{\mathbf{v_1}}(\xi_1,\xi_2) & = & \lambda_1 \\ \varphi_t^{\mathbf{v_2}}(\xi_1,\xi_2) & = & \lambda_2 \\ \varphi_t^{\mathbf{v_3}}(\xi_1,\xi_2) & = & \lambda_3 \end{array}\right.$$

Edge functions

$$\left\{\begin{array}{lcll} \varphi_{k,t}^{\mathbf{e_1}} & = & \lambda_1\lambda_2\,\phi_{k-2}(\lambda_2-\lambda_1), & 2 \le k \le p_i^{\mathbf{e_1}} \\ \varphi_{k,t}^{\mathbf{e_2}} & = & \lambda_2\lambda_3\,\phi_{k-2}(\lambda_3-\lambda_2), & 2 \le k \le p_i^{\mathbf{e_2}} \\ \varphi_{k,t}^{\mathbf{e_3}} & = & \lambda_3\lambda_1\,\phi_{k-2}(\lambda_3-\lambda_1), & 2 \le k \le p_i^{\mathbf{e_3}} \end{array}\right.$$

Bubble functions

$$\varphi_{\mathbf{n_1},\mathbf{n_2},t}^{b} = \lambda_1(\lambda_2)^{\mathbf{n_1}}(\lambda_3)^{\mathbf{n_2}}, \quad 1 \le n_1, n_2; \quad n_1 + n_2 \le p_i^b - 1$$

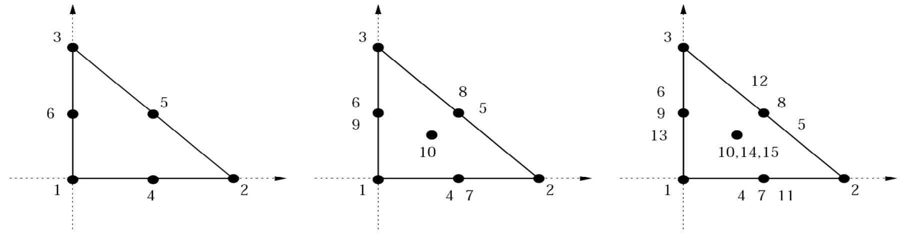
Numerical study of various polynomial interpolation methods

Design of hierarchical basis functions: Solin polynomials

Numbers of hierarchical shape functions in the basis of the space W_t

Node type	Polynomial order	Number of shape functions	Number of nodes
Vertex	**always**	**1**	**3**
Edge	$2 \leq p_i^{e_j}$	$p_i^{e_j} - 1$	**3**
Interior	$3 \leq p_i^b$	$(p_i^b - 1)(p_i^b - 2)/2$	**1**

Local numerotation of degrees of freedom for a $\mathbb{P}_{p_i}$ triangle ($p_i \leq 4$)



$\mathbb{P}_2$ basis

$\mathbb{P}_3$ basis

$\mathbb{P}_4$ basis

The noteworthy asset of hierarchical basis

P2 Implementation

$$\begin{aligned}
\varphi_t^1(\xi_1,\xi_2) &= \lambda_1\\
\varphi_t^2(\xi_1,\xi_2) &= \lambda_2\\
\varphi_t^3(\xi_1,\xi_2) &= \lambda_3\\
\varphi_t^4(\xi_1,\xi_2) &= -\lambda_1\lambda_2\,\sqrt{3}\\
\varphi_t^5(\xi_1,\xi_2) &= -\lambda_2\lambda_3\,\sqrt{3}\\
\varphi_t^6(\xi_1,\xi_2) &= -\lambda_3\lambda_1\,\sqrt{3}
\end{aligned}$$

P3 Implementation

$$\begin{aligned}
\varphi_t^7(\xi_1,\xi_2) &= -\lambda_1\lambda_2\,\sqrt{5}\,(\lambda_2-\lambda_1)\\
\varphi_t^8(\xi_1,\xi_2) &= -\lambda_2\lambda_3\,\sqrt{5}\,(\lambda_3-\lambda_2)\\
\varphi_t^9(\xi_1,\xi_2) &= -\lambda_3\lambda_1\,\sqrt{5}\,(\lambda_3-\lambda_1)\\
\varphi_t^{10}(\xi_1,\xi_2) &= 3\,\lambda_1\lambda_2\lambda_3
\end{aligned}$$

P4 Implementation

$$\begin{aligned}
\varphi_t^{11}(\xi_1,\xi_2) &= -\frac{1}{4}\,\lambda_1\lambda_2\,\sqrt{7}\,(5\,(\lambda_2-\lambda_1)^2-1)\\
\varphi_t^{12}(\xi_1,\xi_2) &= -\frac{1}{4}\,\lambda_2\lambda_3\,\sqrt{7}\,(5\,(\lambda_3-\lambda_2)^2-1)\\
\varphi_t^{13}(\xi_1,\xi_2) &= -\frac{1}{4}\,\lambda_3\lambda_1\,\sqrt{7}\,(5\,(\lambda_3-\lambda_1)^2-1)\\
\varphi_t^{14}(\xi_1,\xi_2) &= \lambda_1\lambda_2\lambda_3\,\sqrt{3}\,\sqrt{5}\,(\lambda_2-\lambda_1)\\
\varphi_t^{15}(\xi_1,\xi_2) &= \lambda_1\lambda_2\lambda_3\,\sqrt{3}\,\sqrt{5}\,(\lambda_3-\lambda_2)
\end{aligned}$$

Numerical study of various polynomial interpolation methods

Numerical results in 2D: eigenmode in a square PEC cavity

DGTD-$\mathbb{P}_{p_i}$ method based on Lagrange basis functions

Interpolation	Mesh	# iter	Final L^2 error	CPU time
$\mathbb{P}_1$	N=41	1698	8.07×10^{-3}	8.5 sec
-	N=81	3395	2.95×10^{-3}	68.2 sec
$\mathbb{P}_2$	N=21	1698	6.88×10^{-4}	3.4 sec
-	N=41	3395	1.69×10^{-4}	26.7 sec
$\mathbb{P}_3$	N=11	1415	9.09×10^{-4}	1.1 sec
-	N=21	2829	2.26×10^{-4}	9.1 sec
$\mathbb{P}_4$	N=11	2829	2.26×10^{-4}	3.8 sec
-	N=21	5657	5.61×10^{-5}	31.2 sec
$\mathbb{P}_5$	N=5	5657	5.30×10^{-5}	1.9 sec
-	N=11	14143	6.94×10^{-6}	29.3 sec
$\mathbb{P}_6$	N=3	5657	5.00×10^{-5}	0.8 sec
-	N=5	11314	5.55×10^{-6}	5.9 sec

DGTD-$\mathbb{P}_{p_i}$ method based on Bernstein basis functions

Interpolation	Mesh	# iter	Final L^2 error	CPU time
$\mathbb{P}_5$	N=5	5657	5.61×10^{-5}	1.9 sec
-	N=11	14143	8.17×10^{-6}	30.0 sec
$\mathbb{P}_6$	N=3	5657	5.69×10^{-5}	0.8 sec
-	N=5	11314	1.32×10^{-5}	5.9 sec

Numerical study of various polynomial interpolation methods

Numerical results in 2D: eigenmode in a square PEC cavity

DGTD-$\mathbb{P}_{p_i}$ method based on Ainsworth-Coyle basis functions

Interpolation	Mesh	# iter	Final L^2 error	CPU time
$\mathbb{P}_5$	N=5	5657	5.61×10^{-5}	2.7 sec
-	N=11	14143	8.17×10^{-6}	40.2 sec
$\mathbb{P}_6$	N=3	5657	5.69×10^{-5}	1.1 sec
-	N=5	11314	1.32×10^{-5}	8.3 sec

DGTD-$\mathbb{P}_{p_i}$ method based on Solin basis functions

Interpolation	Mesh	# iter	Final L^2 error	CPU time
$\mathbb{P}_5$	N=5	5657	5.61×10^{-5}	2.6 sec
-	N=11	14143	8.17×10^{-6}	40.0 sec
$\mathbb{P}_6$	N=3	5657	5.69×10^{-5}	1.1 sec
-	N=5	11314	1.32×10^{-5}	8.4 sec

DGTD-$\mathbb{P}_{p_i}$ method based on Sherwin-Karniadakis basis functions

Interpolation	Mesh	# iter	Final L^2 error	CPU time
$\mathbb{P}_5$	N=5	5657	5.61×10^{-5}	6.8 sec
-	N=11	14143	8.17×10^{-6}	113.5 sec
$\mathbb{P}_6$	N=3	5657	5.69×10^{-5}	2.4 sec
-	N=5	11314	1.32×10^{-5}	18.9 sec

Numerical study of various polynomial interpolation methods

Conclusion

- Preliminary study in 1D & 2D to improve the polynomial approximation flexibility in view of the development of p-adaptive DGTD methods
- Accuracy:
  - no distinction between candidates for basis expansions
  - High order time schemes clearly impact
- Performance:
  - Lagrange & Bernstein basis functions are the most efficient
  - Sherwin-Karniadakis basis functions are the worst

Outline

1. Time-domain electromagnetics
   - Overview of existing methods
   - A non-dissipative DGTD-$\mathbb{P}_{p_i}$ method
2. Numerical study of various polynomial interpolation methods
   - Numerical results in 1D
   - Design of hierarchical basis functions
   - Numerical results in 2D
3. Local time step high order DGTD method
   - Formulation
   - Algorithmic aspect
   - Numerical results in 1D
   - Numerical results in 2D
4. DGTD solver on GPU based systems
   - Performance results in 3D
5. Perspectives

Local time step high order DGTD method

Introduction

Objectives of the study

- Establish the formulation of an energy conserving explicit local time-stepping scheme for first-order Maxwell's equations from the standard second-order leap-frog scheme
- Discuss algorithmic aspects of this local time-stepping scheme
- Carry out a numerical comparative study based on various 1D & 2D problems to illustrate the usefullness of the local time-stepping scheme in terms of computational benefits

Context of the study

Study undertaken in collaboration with S. Descombes (University of Nice-Sophia Antipolis and INRIA Sophia Antipolis - Méditerrannée, NACHOS project team) and J. Diaz (INRIA Bordeaux - Sud-Ouest, MAGIQUE-3D project team)

Local time step high order DGTD method

Main related works

- F.Collino, T.Fouquet, and P.Joly
  J.Comput.Phys., Vol 211, No 1, 2006
  Conservative space-time mesh refinement methods for the FDTD solution of Maxwell's equations

- S.Piperno
  ESAIM: Math.Model.Numer.Anal., Vol 40, No 5, 2006
  Symplectic local time-stepping in non-dissipative DGTD methods applied to wave propagation problems

- J.Diaz and M.J.Grote
  SIAM J.Sci.Comput., Vol 31, 2009
  Energy conserving explicit local time-stepping for second-order wave equations

- M.J.Grote and T.Mitkova
  Technical Report, University of Basel, Department of Mathematics, 2009
  Explicit local time-stepping methods for Maxwell's equations

Local time step high order DGTD method

Formulation

Global EDO system

$$\left\{\begin{array}{lcl} \mathbf{M}^{\varepsilon}\dfrac{d\mathbb{E}}{dt} & = & \mathbb{S}\mathbb{H} \\[2ex] \mathbf{M}^{\mu}\dfrac{d\mathbb{H}}{dt} & = & -{}^{t}\mathbb{S}\mathbb{E} \end{array}\right. \Longrightarrow \left\{\begin{array}{lcl} \dfrac{d\mathbb{E}}{dt} & = & \mathbb{S}^{\varepsilon}\mathbb{H} \\[2ex] \dfrac{d\mathbb{H}}{dt} & = & \mathbb{S}^{\mu}\mathbb{E} \end{array}\right.$$

with $\mathbb{S}^{\varepsilon} = (\mathbf{M}^{\varepsilon})^{-1}\,\mathbb{S}$ and $\mathbb{S}^{\mu} = (-\mathbf{M}^{\mu})^{-1}\,{}^{t}\mathbb{S}$

Splitting of $\mathbb{E}$ and $\mathbb{H}$

$$\left\{\begin{array}{lclcl} \mathbb{E}(t) & = & (\mathbb{I}-\mathbb{P})\mathbb{E}(t)+\mathbb{P}\mathbb{E}(t) & = & \mathbb{E}^{[\text{coarse}]}(t)+\mathbb{E}^{[\text{fine}]}(t), \\ \mathbb{H}(t) & = & (\mathbb{I}-\mathbb{P})\mathbb{H}(t)+\mathbb{P}\mathbb{H}(t) & = & \mathbb{H}^{[\text{coarse}]}(t)+\mathbb{H}^{[\text{fine}]}(t), \end{array}\right.$$

Local time step high order DGTD method

Formulation

Splitting of $\mathbb{E}$ and $\mathbb{H}$

Introducing the auxiliary variable $\tau \in [0\ ,\ \Delta t]$, we consider the previous system from t^{n} to $t^{n} + \Delta t$:

$$\left\{\begin{array}{lcll} \dfrac{d\mathbb{E}}{d\tau}(t^{n}+\tau) & = & \mathbb{S}^{\varepsilon}\left[\mathbb{H}^{[\text{coarse}]}(t^{n}+\tau) + \mathbb{P}\mathbb{H}(t^{n}+\tau)\right], & \tau \in [0\ ,\ \Delta t], \\ \dfrac{d\mathbb{H}}{d\tau}(t^{n}+\tau) & = & \mathbb{S}^{\mu}\left[\mathbb{E}^{[\text{coarse}]}(t^{n}+\tau) + \mathbb{P}\mathbb{E}(t^{n}+\tau)\right], & \tau \in [0\ ,\ \Delta t], \end{array}\right.$$

Approximations of *slow components*

$$\left\{\begin{array}{lll} \mathbb{H}^{[\text{coarse}]}(t^{n}+\tau) & \text{is approximated by} & \left|\begin{array}{ll} \mathbb{H}^{[\text{coarse}]}(t^{n}) & \text{for } \tau \in [0\ ,\ \dfrac{\Delta t}{2}], \\ \mathbb{H}^{[\text{coarse}]}(t^{n}+\Delta t) & \text{for } \tau \in [\dfrac{\Delta t}{2}\ ,\ \Delta t]. \end{array}\right. \\ & & \\ \mathbb{E}^{[\text{coarse}]}(t^{n}+\tau) & \text{is approximated by} & \mathbb{E}^{[\text{coarse}]}(t^{n}+\dfrac{\Delta t}{2}) \quad \text{for } \tau \in [0\ ,\ \Delta t]. \end{array}\right.$$

Local time step high order DGTD method

Formulation

Approximations of $\mathbb{E}^{[\text{coarse}]}(t^n + \frac{\Delta t}{2})$ and $\mathbb{H}^{[\text{coarse}]}(t^n + \Delta t)$

$$
\left\{
\begin{array}{lcl}
\mathbb{E}^{[\text{coarse}]}(t^n + \frac{\Delta t}{2}) & = & (\mathbb{I} - \mathbb{P})\mathbb{E}(t^n + \frac{\Delta t}{2}) \\
 & \approx & (\mathbb{I} - \mathbb{P}) \left[\mathbb{E}(t^n) + \frac{\Delta t}{2} \frac{d\mathbb{E}}{d\tau}(t^n) \right] \\
 & = & (\mathbb{I} - \mathbb{P}) \left[\mathbb{E}(t^n) + \frac{\Delta t}{2} \mathbb{S}^{\varepsilon} \mathbb{H}(t^n) \right], \\
 & & \\
\mathbb{H}^{[\text{coarse}]}(t^n + \Delta t) & = & (\mathbb{I} - \mathbb{P})\mathbb{H}(t^n + \Delta t) \\
 & \approx & (\mathbb{I} - \mathbb{P}) \left[\mathbb{H}(t^n) + \Delta t \frac{d\mathbb{H}}{d\tau}(t^n + \frac{\Delta t}{2}) \right] \\
 & = & (\mathbb{I} - \mathbb{P}) \left[\mathbb{H}(t^n) + \Delta t \mathbb{S}^{\mu} \mathbb{E}(t^n + \frac{\Delta t}{2}) \right] \\
 & \approx & (\mathbb{I} - \mathbb{P}) \left[\mathbb{H}(t^n) + \Delta t \left(\mathbb{S}^{\mu} \mathbb{E}^{[\text{coarse}]}(t^n + \frac{\Delta t}{2}) + \mathbb{S}^{\mu} \mathbb{E}^{[\text{fine}]}(t^n + \frac{\Delta t}{2}) \right) \right] \\
 & \approx & (\mathbb{I} - \mathbb{P}) \left[\mathbb{H}(t^n) + \Delta t \left(\mathbb{S}^{\mu} \left((\mathbb{I} - \mathbb{P}) \left[\mathbb{E}(t^n) + \frac{\Delta t}{2} \mathbb{S}^{\varepsilon} \mathbb{H}(t^n) \right] \right) + \mathbb{S}^{\mu} \mathbb{E}^{[\text{fine}]}(t^n + \frac{\Delta t}{2}) \right) \right].
\end{array}
\right.
$$

Local time step high order DGTD method

Formulation

Three steps scheme

$$\left\{\begin{array}{rcl}
\dfrac{d\mathbb{E}}{d\tau}(\boldsymbol{t^n}+\tau) & = & \mathbb{S}^{\varepsilon}\,[(\mathbb{I}-\mathbb{P})\mathbb{H}(\boldsymbol{t^n})+\mathbb{P}\mathbb{H}(\boldsymbol{t^n}+\tau)]\,, \qquad \tau\in[0\;,\;\dfrac{\Delta t}{2}], \\
\dfrac{d\mathbb{H}}{d\tau}(\boldsymbol{t^n}+\tau) & = & \mathbb{S}^{\mu}\,\left[(\mathbb{I}-\mathbb{P})\left(\mathbb{E}(\boldsymbol{t^n})+\dfrac{\Delta t}{2}\mathbb{S}^{\varepsilon}\mathbb{H}(\boldsymbol{t^n})\right)+\mathbb{P}\mathbb{E}(\boldsymbol{t^n}+\tau)\right], \qquad \tau\in[0\;,\;\Delta t], \\
\dfrac{d\mathbb{E}}{d\tau}(\boldsymbol{t^n}+\tau) & = & \mathbb{S}^{\varepsilon}\,\Bigg[(\mathbb{I}-\mathbb{P})\Bigg(\mathbb{H}(\boldsymbol{t^n})+\Delta t\,\Bigg(\mathbb{S}^{\mu}\,\Bigg((\mathbb{I}-\mathbb{P})\left[\mathbb{E}(\boldsymbol{t^n})+\dfrac{\Delta t}{2}\mathbb{S}^{\varepsilon}\mathbb{H}(\boldsymbol{t^n})\right]\Bigg)+ \\
 & & \qquad \mathbb{S}^{\mu}\mathbb{E}^{[\mathrm{fine}]}(\boldsymbol{t^n}+\dfrac{\Delta t}{2})\Bigg)\Bigg)+\mathbb{P}\mathbb{H}(\boldsymbol{t^n}+\tau)\Bigg], \qquad \tau\in[\dfrac{\Delta t}{2}\;,\;\Delta t],
\end{array}\right.$$

Local time step high order DGTD method

Algorithmic aspects

The case $p = 2$

1. **Set** $\mathbb{U} = \mathbb{S}^{\varepsilon}(\mathbb{I} - \mathbb{P})\mathbb{H}^{n}$, $\tilde{\mathbb{U}} = \mathbb{S}^{\varepsilon}\mathbb{P}\mathbb{H}^{n}$ **and** $\mathbb{V} = \mathbb{S}^{\mu}(\mathbb{I} - \mathbb{P})\left(\mathbb{E}^{n} + \frac{\Delta t}{2}\left[\mathbb{U} + \tilde{\mathbb{U}}\right]\right)$
2. **Compute**

$$\begin{aligned}
\mathbb{E}^{n+\frac{1}{4}} &= \mathbb{E}^{n} + \frac{\Delta\tau}{2}\left[\mathbb{U} + \tilde{\mathbb{U}}\right], \\
\mathbb{H}^{n+\frac{1}{2}} &= \mathbb{H}^{n} + \Delta\tau\left[\mathbb{V} + \mathbb{S}^{\mu}\mathbb{P}\mathbb{E}^{n+\frac{1}{4}}\right], \\
\mathbb{E}^{n+\frac{1}{2}} &= \mathbb{E}^{n+\frac{1}{4}} + \frac{\Delta\tau}{2}\left[\mathbb{U} + \mathbb{S}^{\varepsilon}\mathbb{P}\mathbb{H}^{n+\frac{1}{2}}\right].
\end{aligned}$$

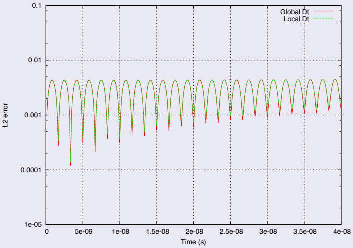
3. **Set** $\tilde{\mathbb{V}} = \mathbb{S}^{\mu}\mathbb{P}\mathbb{E}^{n+\frac{1}{2}}$ **and** $\mathbb{W} = \mathbb{U} + \Delta t\mathbb{S}^{\varepsilon}(\mathbb{I} - \mathbb{P})\left(\mathbb{V} + \tilde{\mathbb{V}}\right)$
4. **Compute**

$$\begin{aligned}
\mathbb{E}^{n+\frac{3}{4}} &= \mathbb{E}^{n+\frac{1}{2}} + \frac{\Delta\tau}{2}\left[\mathbb{W} + \mathbb{S}^{\varepsilon}\mathbb{P}\mathbb{H}^{n+\frac{1}{2}}\right], \\
\mathbb{H}^{n+1} &= \mathbb{H}^{n+\frac{1}{2}} + \Delta\tau\left[\mathbb{V} + \mathbb{S}^{\mu}\mathbb{P}\mathbb{E}^{n+\frac{3}{4}}\right], \\
\mathbb{E}^{n+1} &= \mathbb{E}^{n+\frac{3}{4}} + \frac{\Delta\tau}{2}\left[\mathbb{W} + \mathbb{S}^{\varepsilon}\mathbb{P}\mathbb{H}^{n+1}\right].
\end{aligned}$$

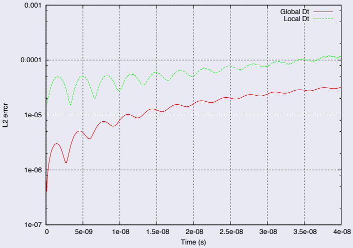
Local time step high order DGTD method

Numerical results in 1D: eigenmode in a unit interval with metallic boundaries

Time evolution of the L_2 error for $p = 2$



DGTD-$\mathbb{P}_1$ method

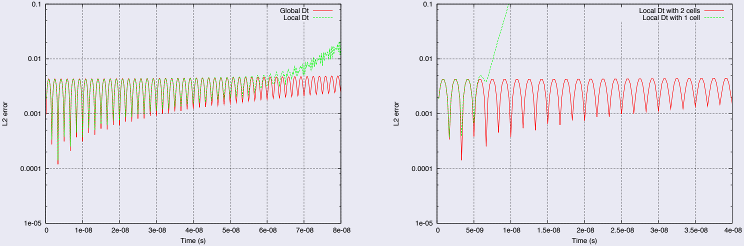


DGTD-$\mathbb{P}_2$ method

Local time step high order DGTD method

Numerical results in 1D: eigenmode in a unit interval with metallic boundaries

Time evolution of the L_2 error for the DGTD-$\mathbb{P}_1$ method and $p = 2$
Influence of the overlapping strategy



Local time step high order DGTD method

Numerical results in 1D: eigenmode in a unit interval with metallic boundaries

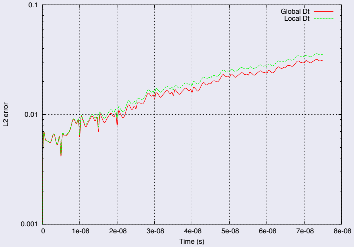
Maximum L_2 error in time (simulation time $T=4 \times 10^{-8}$ s)

p	Time stepping strategy	L_2 error DGTD-$\mathbb{P}_1$	CPU (s) DGTD-$\mathbb{P}_1$	L_2 error DGTD-$\mathbb{P}_2$	CPU (s) DGTD-$\mathbb{P}_2$
2	Global Δt	4.467×10^{-3}	0.09	3.222×10^{-5}	0.24
-	Local Δt	4.533×10^{-3}	0.06	1.224×10^{-4}	0.15
4	Global Δt	4.303×10^{-3}	0.19	8.065×10^{-6}	0.49
-	Local Δt	4.391×10^{-3}	0.07	1.612×10^{-3}	0.19
8	Global Δt	4.227×10^{-3}	0.42	2.078×10^{-6}	1.18
-	Local Δt	4.322×10^{-3}	0.13	7.137×10^{-4}	0.33

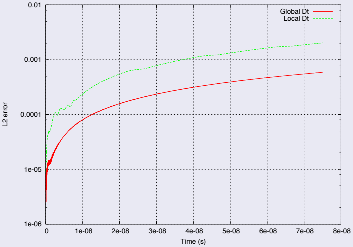
Local time step high order DGTD method

Numerical results in 1D: gaussian pulse with periodic boundaries

Time evolution of the L_2 error for $p = 2$



DGTD-$\mathbb{P}_1$ method

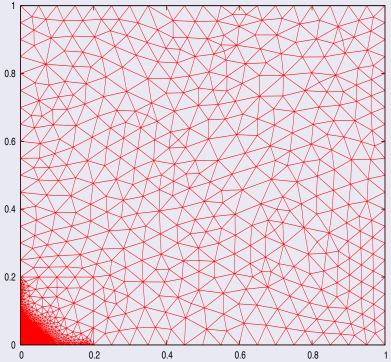


DGTD-$\mathbb{P}_2$ method

Local time step high order DGTD method

Numerical results in 2D: eigenmode in square PEC cavity

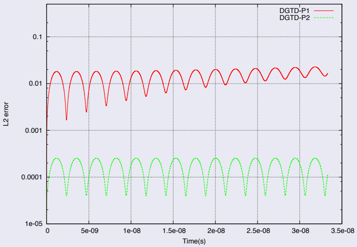
Unstructured triangular mesh ($p = 50$)



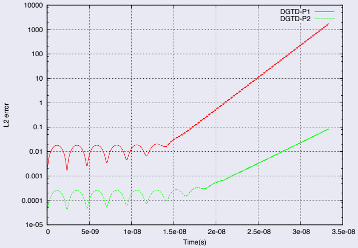
Local time step high order DGTD method

Numerical results in 2D: eigenmode in square PEC cavity

Time evolution of the L_2 error for $p = 50$



Global time stepping strategy

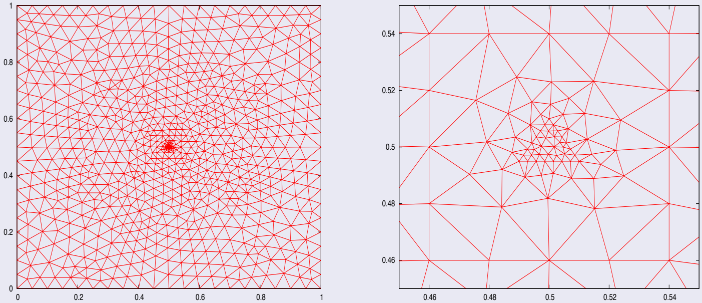


Local time stepping strategy

Local time step high order DGTD method

Numerical results in 2D: wave initiated by a localized source in vacuum

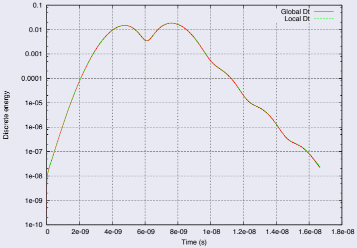
Unstructured triangular mesh ($p = 10$)



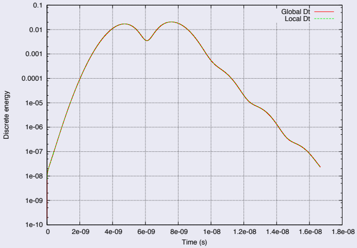
Local time step high order DGTD method

Numerical results in 2D: wave initiated by a localized source in vacuum

Time evolution of the discrete energy for $p = 10$

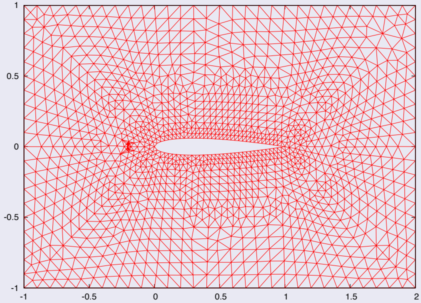


DGTD-$\mathbb{P}_1$ method



DGTD-$\mathbb{P}_2$ method

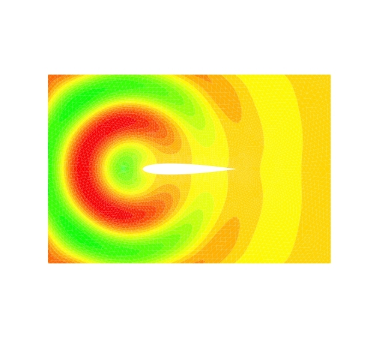
Unstructured triangular mesh ($p = 116$)



Local time step high order DGTD method

Numerical results in 2D: scattering by an airfoil profile

Contour lines of the E_z component, T = 9.16×10^{-9} sec



Performance results

Time stepping strategy	# iter DGTD-$\mathbb{P}_1$	CPU (s) DGTD-$\mathbb{P}_1$	# iter DGTD-$\mathbb{P}_2$	CPU (s) DGTD-$\mathbb{P}_2$
Global Δt	615,992	1220.0	5376	125.0
Local Δt	307,996	282.0	2688	27.0

Local time step high order DGTD method

Conclusion

- Requires numerical stability study
- Needs to be combined with a parallelization strategy
- Can be extended to higher order time integration schemes
- Motivates a 3D implementation of the method

Outline

1. Time-domain electromagnetics
   - Overview of existing methods
   - A non-dissipative DGTD-$\mathbb{P}_{p_i}$ method
2. Numerical study of various polynomial interpolation methods
   - Numerical results in 1D
   - Design of hierarchical basis functions
   - Numerical results in 2D
3. Local time step high order DGTD method
   - Formulation
   - Algorithmic aspect
   - Numerical results in 1D
   - Numerical results in 2D
4. DGTD solver on GPU based systems
   - Performance results in 3D
5. Perspectives

Objective of the study

Adaptation of a DGTD solver for the time-domain Maxwell equations on hybrid CPU/GPU parallel systems

Main related works

- D. Komatitsch, D. Michea, G. Erlebacher
  Journal of Parallel and Distributed Computing, Vol. 69, No. 5 2009
  Porting a high-order finite-element earthquake modeling application to NVIDIA graphics cards using CUDA.
- Tim Warburton, Jeff Bridge, Jan S Hesthaven
  Journal of Computational Physics, Vol. 228, No. 21, 2009
  Nodal Discontinuous Galerkin Methods on Graphics Processors.
- H. Riedmann
  Project Thesis at Scientific Computing Group, Brown University, 2009
  Efficient Numerical Treatment of the Compressible Navier-Stokes Equations with Nodal Discontinuous Galerkin Methods on Graphics Processors.

DGTD solver on GPU based systems

Performance results: computing plateforms

Simulations are performed in single precision arithmetic

Titane GT200

Hybrid CPU-GPU cluster of the CCRT (Centre de Calcul Recherche et Technologie).

- 1068 Intel CPU nodes with 2 quad-core Intel Xeon X5570 Nehalem processors (2.93 GHz)
- 48 Teslas S1070 GPU systems (4 C1060 GPUs of 4 GB of memory each and 2 PCI-Express 2.0 buses)
- Infiniband DDR: 20 Gb/s

Plafrim Fermi

Hybrid CPU-GPU cluster of the INRIA Bordeaux.

- 8 Intel CPU nodes with 2 hexa-core Wetsmare Intel Xeon X5650 processors (2.67 GHz)
- 24 Teslas M2070 GPU systems (6 GB of memory each)
- Infiniband QDR: 40 Gb/s

DGTD solver on GPU based systems

Performance results: weak scalability assessment

Model test problem and configurations

Propagation of a standing wave in a perfectly conducting unitary cubic cavity.

Regular uniform tetrahedral meshes respectively containing 3,072,000 elements for the DGTD-$\mathbb{P}_1$ and DGTD-$\mathbb{P}_2$ methods, 1,296,000 elements for the DGTD-$\mathbb{P}_3$ method and 750,000 elements for the DGTD-$\mathbb{P}_4$ method.

Boxwise domain decompositions with optimal computational load balance.

Timings for 1000 iterations and up to 128 GPUs.

Timings and sustained performance figures (C1060 & M2070)

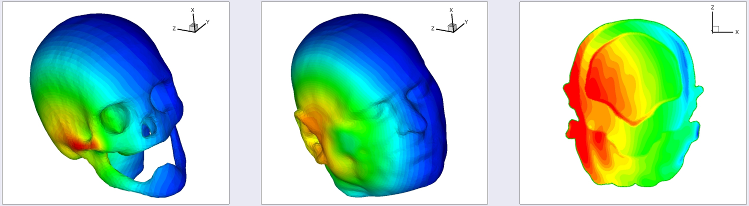
# GPU	DGTD-$\mathbb{P}_1$		DGTD-$\mathbb{P}_2$		DGTD-$\mathbb{P}_3$		DGTD-$\mathbb{P}_4$	
	Time	GFlops	Time	GFlops	Time	GFlops	Time	GFlops
1	104.7 sec	63	325.1 sec	92	410.3 sec	106	759.8 sec	94
128	104.9 sec	8072	323.1 sec	11844	408.4 sec	13676	763.6 sec	12009

# GPU	DGTD-$\mathbb{P}_1$		DGTD-$\mathbb{P}_2$		DGTD-$\mathbb{P}_3$		DGTD-$\mathbb{P}_4$	
	Time	GFlops	Time	GFlops	Time	GFlops	Time	GFlops
1	64.2 sec	106	166.7 sec	181	194.0 sec	226	370.5 sec	195
16	67.0 sec	1620	171.5 sec	2820	198.2 sec	3540	381.9 sec	3025

DGTD solver on GPU based systems

Performance results: strong scalability assessment

Head tissues exposure to mobile phone radiation



Characteristics of the fully unstructured tetrahedral mesh of head tissues

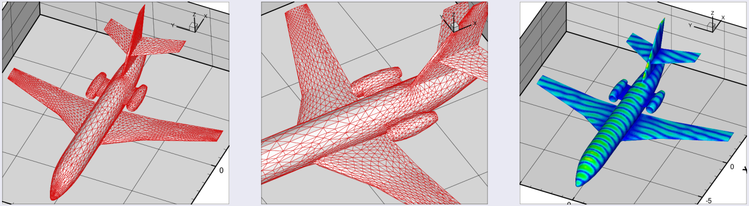
- $\#$ elements $= 7,894,172$
- Total $\#$ DOF is 189,458,688 (DGTD-$\mathbb{P}_1$) and 473,646,720 (DGTD-$\mathbb{P}_2$)
- Time on 128 CPU cores: 2786 sec (DGTD-$\mathbb{P}_1$) and 6057 sec (DGTD-$\mathbb{P}_2$)

# GPU	DGTD-$\mathbb{P}_1$			DGTD-$\mathbb{P}_2$		
	Time	GFlops	Speedup	Time	GFlops	Speedup
32	**162 sec**	**1460**	**-**	**816 sec**	**2370**	**-**
64	**97 sec**	**2470**	× **1.7**	**416 sec**	**4657**	× **2.0**
128	**69 sec**	**3469**	× **2.4**	**257 sec**	**7522**	× **3.2**

DGTD solver on GPU based systems

Performance results: strong scalability assessments, Fermi vs GT200

Scattering of a plane wave by an aircraft geometry



1000 iterations, Mesh = 2,024,924 elements

# GPU	DGTD-$\mathbb{P}_1$			DGTD-$\mathbb{P}_2$		
	Time	GFlops	Speedup	Time	GFlops	Speedup
4	**18.6 sec**	**242**	**-**	**42.7 sec**	**468**	**-**
8	**12.6 sec**	**355**	**×1.5**	**25.1 sec**	**795**	**×1.7**
16	**9.7 sec**	**464**	**×1.9**	**16.1 sec**	**1239**	**×2.7**

# GPU	DGTD-$\mathbb{P}_3$			DGTD-$\mathbb{P}_4$		
	Time	GFlops	Speedup	Time	GFlops	Speedup
4	**89.0 sec**	**771**	**-**	**265.6 sec**	**735**	**-**
8	**48.5 sec**	**1414**	**×1.8**	**139.7 sec**	**1396**	**×1.9**
16	**28.0 sec**	**2452**	**×3.2**	**75.2 sec**	**2593**	**×3.5**

DGTD solver on GPU based systems

Performance results: strong scalability assessments, Fermi vs GT200

Scattering of a plane wave by an aircraft geometry: 8 GPUs

GPU	DGTD-$\mathbb{P}_1$		DGTD-$\mathbb{P}_2$		DGTD-$\mathbb{P}_3$		DGTD-$\mathbb{P}_4$	
	Time	GFlops	Time	GFlops	Time	GFlops	Time	GFlops
CPU	**324.0 sec**	**-**	**902.0 sec**	**-**	**2517.4 sec**	**-**	**5736.3 sec**	**-**
C2050/C2070	**20.1 sec**	**223**	**32.6 sec**	**612**	**53.6 sec**	**1218**	**148.4 sec**	**1315**
M2070	**12.6 sec**	**355**	**25.1 sec**	**795**	**48.5 sec**	**1414**	**139.7 sec**	**1396**

GPU	**Speedups**			
	DGTD-$\mathbb{P}_1$	DGTD-$\mathbb{P}_2$	DGTD-$\mathbb{P}_3$	DGTD-$\mathbb{P}_4$
C2050/C2070	$\times$**16.1**	$\times$**27.7**	$\times$**47.0**	$\times$**38.6**
M2070	$\times$**25.7**	$\times$**35.9**	$\times$**51.9**	$\times$**41.1**

Outline

1. Time-domain electromagnetics
   - Overview of existing methods
   - A non-dissipative DGTD-$\mathbb{P}_{p_i}$ method
2. Numerical study of various polynomial interpolation methods
   - Numerical results in 1D
   - Design of hierarchical basis functions
   - Numerical results in 2D
3. Local time step high order DGTD method
   - Formulation
   - Algorithmic aspect
   - Numerical results in 1D
   - Numerical results in 2D
4. DGTD solver on GPU based systems
   - Performance results in 3D
5. Perspectives

Perspectives

Perspectives

- Polynomial interpolation methods:
  - Fully exploit the hierarchical characteristic of considered polynomial basis functions to enable the use of *p*-adaptive DGTD-$\mathbb{P}_{p_i}$ methods
  - Design of a comparative study of a-posteriori error estimators for the development of *hp*-adaptive DGTD-$\mathbb{P}_{p_i}$ methodology in 3D
- Local time step method:
  - Perform a theoretical stability study for the local time-stepping strategy
  - Pursue the related numerical study in-depth and complete it by an a-priori accuracy analysis and by the assessment of higher time integration schemes
- GPU implementation:
  - Minimize exchanges between GPUs and stress the locality of data by a multi-level mesh partitioning
  - Upgrade to CUDA 4.1 and assess performances when using double precision data