Computational electromagnetics

Stéphane Lanteri

NACHOS project-team, INRIA Sophia Antipolis - Méditerranée
06902 Sophia Antipolis Cedex, France
Stephane.Lanteri@inria.fr

MathMods - Erasmus Mundus MSc Course
Applied mathematics: between industry and research
March 31st, 2011

Computational electromagnetics

- Computational electromagnetics, computational electrodynamics or electromagnetic modeling is the process of modeling the interaction of electromagnetic fields with physical objects and the environment
- Several real-world electromagnetic problems like scattering, radiation, waveguiding etc., are not analytically calculable, for the multitude of irregular geometries found in actual devices
- Computational (numerical) techniques can overcome the inability to derive closed form solutions of Maxwell's equations under various constitutive relations of media, and boundary conditions
- This makes computational electromagnetics (CEM), important to the design, and modeling of:
  - antenna, radar, satellite and other communication systems,
  - nanophotonic devices and high speed silicon electronics,
  - medical imaging,
  - cell-phone antenna design,
  - etc.

Computational electromagnetics

- Computational electromagnetics, computational electrodynamics or electromagnetic modeling is the process of modeling the interaction of electromagnetic fields with physical objects and the environment
- Several real-world electromagnetic problems like scattering, radiation, waveguiding etc., are not analytically calculable, for the multitude of irregular geometries found in actual devices
- Computational (numerical) techniques can overcome the inability to derive closed form solutions of Maxwell's equations under various constitutive relations of media, and boundary conditions
- This makes computational electromagnetics (CEM), important to the design, and modeling of:
    - antenna, radar, satellite and other communication systems,
    - nanophotonic devices and high speed silicon electronics,
    - medical imaging,
    - cell-phone antenna design,
    - etc.

Radar cross section (RCS)

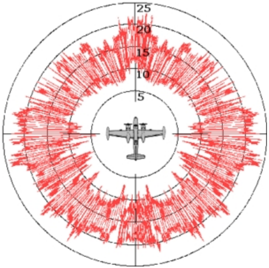
- RCS is a measure of how detectable an object is with a radar
- A larger RCS indicates that an object is more easily detected
- An object reflects a limited amount of radar energy
- Factors that determine how much electromagnetic energy returns to the source:
  - material of which the target is made,
  - absolute size of the target,
  - relative size of the target (in relation to the wavelength of the illuminating radar),
  - the incident angle (angle at which the radar beam hits a particular portion of target),
  - reflected angle (angle at which the reflected beam leaves the part of the target hit),
  - strength of the radar emitter,
  - distance between emitter-target-receiver.

Radar cross section (RCS)

- RCS is a measure of how detectable an object is with a radar
- A larger RCS indicates that an object is more easily detected
- An object reflects a limited amount of radar energy
- Factors that determine how much electromagnetic energy returns to the source:
  - material of which the target is made,
  - absolute size of the target,
  - relative size of the target (in relation to the wavelength of the illuminating radar),
  - the incident angle (angle at which the radar beam hits a particular portion of target),
  - reflected angle (angle at which the reflected beam leaves the part of the target hit),
  - strength of the radar emitter,
  - distance between emitter-target-receiver.

Radar cross section (RCS)

- RCS is used to detect planes in a wide variation of ranges
- For example, a stealth aircraft (which is designed to have low detectability) will have design features that give it a low RCS (such as absorbent paint, smooth surfaces, surfaces specifically angled to reflect signal somewhere other than towards the source)
- RCS is integral to the development of radar stealth technology, particularly in applications involving aircraft and ballistic missiles
- RCS data for current military aircraft is most highly classified

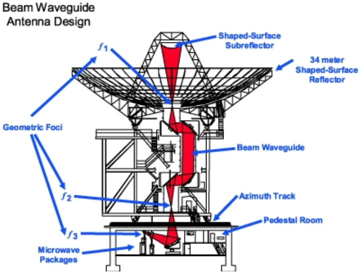


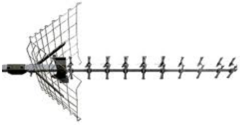
Typical RCS diagram (A-26 Invader)

The B-2 Spirit was one of the first aircraft to successfully become invisible to radar.

Antenna design

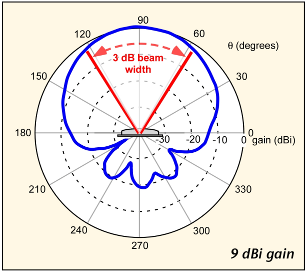
- An antenna (or aerial) is an electrical device which couples radio waves in free space to an electrical current used by a radio receiver or transmitter
- In reception, the antenna intercepts some of the power of an electromagnetic wave in order to produce a tiny voltage that the radio receiver can amplify
- Alternatively, a radio transmitter will produce a large radio frequency current that may be applied to the terminals of the same antenna in order to convert it into an electromagnetic wave (radio wave) radiated into free space





Antenna design

- In the field of antenna design the term radiation pattern most commonly refers to the directional (angular) dependence of the strength of the radio waves from the antenna or other source
- The far-field pattern of an antenna may be determined experimentally at an antenna range, or alternatively, the near-field pattern may be found using a near-field scanner, and the radiation pattern deduced from it by computation
- The radiation pattern can also be calculated from the antenna shape by computer programs



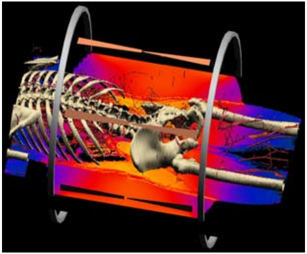
Microwave based hyperthermia

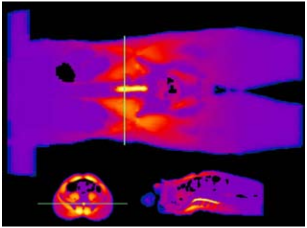
- Hyperthermia means a body temperature that is higher than normal
- High body temperatures are often caused by illness such as fever or heat stroke
- But hyperthermia can also refer to heat treatment, the carefully controlled use of heat for medical purposes
- When cells in the body are exposed to higher than normal temperatures, changes take place inside the cells
- These changes can make the cells more likely to be affected by radiation therapy or chemotherapy
- Very high temperatures can kill cancer cells outright, but they also can injure or kill normal cells and tissues

Example applications

Microwave based hyperthermia

- Computational modeling is used for:
  - the design of the antenna array,
  - the planning of the treatment.
- Systems used for hyperthermia treatment of cancerous tumors require optimization for individual patients (treatment planning)
- This involves the generation of patient-specific models, electromagnetic simulations of the antenna array, simulation of the resulting temperature distribution and its optimization





Opening: scientific and technological context

- Challenges with the simulation of ElectroMagnetic (EM) wave propagation
  - Geometrical characteristics of the propagation domain:
    - dimensions relatively to the wavelength,
    - irregularly shaped objects and singularities.
  - Physical characteristics of the propagation medium:
    - heterogeneity and anisotropy,
    - physical dispersion and dissipation.
  - Characteristics of the radiating sources and incident fields
- PDE model: the system of Maxwell equations

James Clerk Maxwell (1831-1879)

Maxwell's equations

- Maxwell-Faraday equation (Faraday's law of induction)

$$\partial_t \mathbf{B} + \nabla \times \mathbf{E} = 0$$

- Ampère's circuital law (with Maxwell's correction)

$$\partial_t \mathbf{D} - \nabla \times \mathbf{H} = -\mathbf{J}$$

- Gauss's law

$$\nabla . \mathbf{D} = \rho$$

- Gauss's law for magnetism

$$\nabla . \mathbf{B} = 0$$

- Constitutive laws (assumption: no polarization)

$$\mathbf{D} = \varepsilon \mathbf{E} \quad \text{and} \quad \mathbf{B} = \mu \mathbf{H}$$

Content

1. Time-domain electromagnetics
   - Overview of existing methods
   - Discontinuous Galerkin Time Domain (DGTD) method
   - Non-conforming DGTD-$\mathbb{P}_{p_i}$ method
   - Towards higher order time accuracy
   - Numerical treatment of grid-induced stiffness
   - 3D application: electromagnetic waves and humans
2. Frequency-domain electromagnetics
   - Numerical results in the 2D TMz case
   - Domain decomposition solver
3. High performance computing

1. Time-domain electromagnetics
   - Overview of existing methods
   - Discontinuous Galerkin Time Domain (DGTD) method
   - Non-conforming DGTD-$\mathbb{P}_{p_i}$ method
   - Towards higher order time accuracy
   - Numerical treatment of grid-induced stiffness
   - 3D application: electromagnetic waves and humans

2. Frequency-domain electromagnetics
   - Numerical results in the 2D TMz case
   - Domain decomposition solver

3. High performance computing

Time-domain Maxwell equations

$$\left\{\begin{array}{l} \varepsilon\partial_t\mathbf{E} - \nabla\times\mathbf{H} = 0 \\ \\ \mu\partial_t\mathbf{H} + \nabla\times\mathbf{E} = 0 \end{array}\right.$$

$$\mathbf{E} = {}^{\mathrm{T}}(E_x, E_y, E_z) \;\text{ and }\; \mathbf{H} = {}^{\mathrm{T}}(H_x, H_y, H_z)$$

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

$$\left\{\begin{array}{l} \mathbf{n}\times\mathbf{E} = 0 \text{ on } \Gamma_m \\ \mathbf{n}\times\mathbf{E} - \sqrt{\dfrac{\mu}{\varepsilon}}\mathbf{n}\times(\mathbf{H}\times\mathbf{n}) = \mathbf{n}\times\mathbf{E}_{\mathrm{inc}} - \sqrt{\dfrac{\mu}{\varepsilon}}\mathbf{n}\times(\mathbf{H}_{\mathrm{inc}}\times\mathbf{n}) \text{ on } \Gamma_a \end{array}\right.$$

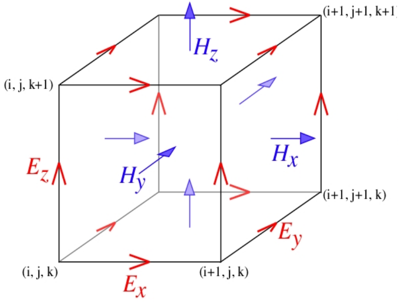
Outline

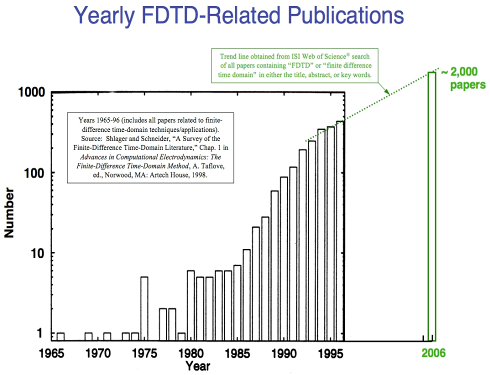
1. Time-domain electromagnetics
   - Overview of existing methods
   - Discontinuous Galerkin Time Domain (DGTD) method
   - Non-conforming DGTD-$\mathbb{P}_{p_i}$ method
   - Towards higher order time accuracy
   - Numerical treatment of grid-induced stiffness
   - 3D application: electromagnetic waves and humans
2. Frequency-domain electromagnetics
   - Numerical results in the 2D TMz case
   - Domain decomposition solver
3. High performance computing

- 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)

Yee's scheme

- Staggered grid
- Non-dissipative scheme (centered in space and time)
- Second-order accurate in space and time (for a uniform grid)





- 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 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 $\rightarrow$ numerical dissipation
    - Centered scheme $\rightarrow$ numerical dispersion (on non-uniform meshes)
  - Extension to higher order accuracy: MUSCL technique

- DGTD: Discontinuous Galerkin Time-Domain method
  - 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 RKDG 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 several other recent works

- Problem to be solved

  $\mathbf{x} \in \Omega \subset \boldsymbol{R}^d$, $t \in \boldsymbol{R}^+$, $u = u(\mathbf{x}, t)$, $a_i = a_i(\mathbf{x})$ scalar real functions

$$\frac{\partial u}{\partial t} + \sum_{i=1}^{d} a_i \frac{\partial u}{\partial x_i} = 0$$

- Weak formulation

$$< \frac{\partial u}{\partial t}, v >_\Omega + \sum_{i=1}^{d} < a_i \frac{\partial u}{\partial x_i}, v >_\Omega = 0$$

$< u \, , \, v >_\Omega = \int_\Omega uv d\mathbf{x}$, v being a *test function*

- Galerkin method
  - $\tau_h = \{K\}$ triangulation of Ω
  - $P^m(K)$: polynomials of degree at most m on K

For each $K \in \tau_h$ find $u^h : u^h|_K \in P^m(K)$ such that:

$$< \frac{\partial u^h}{\partial t}, v >_K + \sum_{i=1}^{d} < a_i \frac{\partial u^h}{\partial x_i}, v >_K = 0 \ , \forall v \in P^m(K)$$

Integrating by parts (setting $a|_K \in P^0(K)$)

$$\begin{aligned} < \frac{\partial u^h}{\partial x_i}, v >_K &= - < u^h, \frac{\partial v}{\partial x_i} >_K + < u^h n_i, v >_{\partial K} \\ < u, v >_{\partial K} &= \sum_{j=1}^{N_f(K)} < u, v >_{\partial K \cap \partial K_j} \end{aligned}$$

- $\mathbf{n} = \{n_i\}$ outward unit normal of ∂K
- $N_f(K)$ = number of faces of K

- Discontinous approximation: $u^h|_{K\cap K_j}$ not well defined!
  $\Rightarrow$ Centered or upwind approximations
- Linear algebra
  - $u^h|_K(\mathbf{x},t) = \sum_{j=1}^{m_K} u^h_{j,K}(t)\psi_{j,K}(\mathbf{x})$, $m_K = \dim(P^m(K))$
  - $\{\psi_{j,K}\}, j = 1,\ldots,m_K$: basis of $P^m(K)$

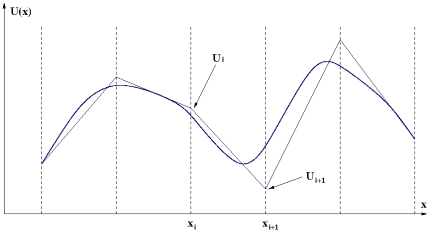
$$\mathbf{M}_K \frac{\partial \mathbf{U}^h_K}{\partial t} = \sum_{i=1}^{d} a_i \left(\mathbf{R}_{i,K}\mathbf{U}^h_K - n_i \sum_{j=1}^{N_f(K)} \mathbf{S}_{K,K_j}\mathbf{U}^h_K \right)$$

$$\left\{\begin{array}{lcl} \mathbf{U}^h_K & = & \mathbf{U}^h_K(t) = \{u^h_{j,K}(t)\},\ j = 1,\ldots,m_K \\ \mathbf{M}_K[l,m] & = & <\psi_{l,K},\ \psi_{m,K}>_K \\ \mathbf{R}_{i,K}[l,m] & = & <\dfrac{\partial\psi_{l,K}}{\partial x_i},\ \psi_{m,K}>_K \\ \mathbf{S}_{K,K_j}[l,m] & = & <\psi_{l,K},\ \psi_{m,K_j}>_{\partial K\cap\partial K_j} \end{array}\right.$$

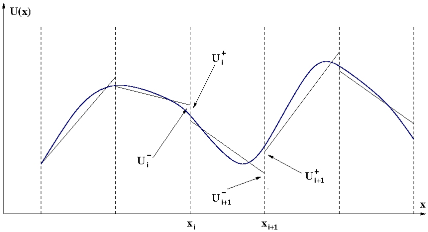
Dimension of local systems: $m_K \times m_K$

Outline

1. Time-domain electromagnetics
   - Overview of existing methods
   - Discontinuous Galerkin Time Domain (DGTD) method
   - Non-conforming DGTD-$\mathbb{P}_{p_i}$ method
   - Towards higher order time accuracy
   - Numerical treatment of grid-induced stiffness
   - 3D application: electromagnetic waves and humans
2. Frequency-domain electromagnetics
   - Numerical results in the 2D TMz case
   - Domain decomposition solver
3. High performance computing



Continuous P1 interpolation



Discontinuous P1 interpolation

- Naturally adapted to heterogeneous media and discontinuous solutions
- Can easily deal with unstructured, possibly non-conforming meshes (h-adaptivity)
- High order with compact stencils and non-conforming approximations (p-adaptivity)
- Usually rely on polynomial interpolation but can also accomodate alternative functions (e.g plane waves)
- Yield block diagonal mass matrices when coupled to explicit time integration schemes
- Amenable to efficient parallelization
- But leads to larger problems compared to continuous finite element methods

DG for electromagnetic wave propagation in heterogeneous media

- Heterogeneity is ideally treated at the element level
  - Discontinuities occur at material (i.e element) interfaces
  - Mesh generation process is simplified
- Wavelength varies with ε and μ
  - For a given mesh density, approximation order can be adapted at the element level in order to fit to the local wavelength

Discretization of irregularly shaped domains

- Unstructured simplicial meshes
- The basic support of the DG method is the element (triangle in 2D and tetrahedron in 3D)
- Local refinement is facilitated by allowing non-conformity
- Non-conformity opens the route to the coupling of different discretization methods (e.g structured/unstructured)

DG for electromagnetic wave propagation in heterogeneous media

- Heterogeneity is ideally treated at the element level
  - Discontinuities occur at material (i.e element) interfaces
  - Mesh generation process is simplified
- Wavelength varies with ε and μ
  - For a given mesh density, approximation order can be adapted at the element level in order to fit to the local wavelength

Discretization of irregularly shaped domains

- Unstructured simplicial meshes
- The basic support of the DG method is the element (triangle in 2D and tetrahedron in 3D)
- Local refinement is facilitated by allowing non-conformity
- Non-conformity opens the route to the coupling of different discretization methods (e.g structured/unstructured)

- Triangulation of Ω: $\overline{\Omega}_h \equiv \mathscr{T}_h = \bigcup_{\tau_i \in \mathscr{T}_h} \overline{\tau}_i$
  - $\mathscr{F}_0$: set of purely internal faces
  - $\mathscr{F}_m$ and $\mathscr{F}_a$: 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: $\forall \vec{\varphi} \in \mathscr{P}_i = \text{Span}(\vec{\varphi}_{ij}\ ,\ 1 \leq j \leq d_i)$

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

- 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 $\mathbf{H}_{k|_{a_{ik}}} = \mathbf{H}_{i|_{a_{ik}}}$

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

- Local projections

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

- 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 $1 \leq j, l \leq d_i$:
  - $(\mathbf{M}_i^{\varepsilon})_{jl} = \varepsilon_i \iiint\limits_{\tau_i} {}^{\mathsf{T}}\vec{\varphi}_{ij}\vec{\varphi}_{jl} d\omega$ and $(\mathbf{M}_i^{\mu})_{jl} = \mu_i \iiint\limits_{\tau_i} {}^{\mathsf{T}}\vec{\varphi}_{ij}\vec{\varphi}_{jl} d\omega$
  - $(\mathbf{K}_i)_{jl} = \dfrac{1}{2} \iiint\limits_{\tau_i} ({}^{\mathsf{T}}\vec{\varphi}_{ij}\nabla \times \vec{\varphi}_{il} + {}^{\mathsf{T}}\vec{\varphi}_{il}\nabla \times \vec{\varphi}_{ij}) d\omega$

- For $1 \leq j \leq d_i$ and $1 \leq l \leq d_k$
  - $(\mathbf{S}_{ik})_{jl} = \dfrac{1}{2} \iint\limits_{a_{ik}} {}^{\mathsf{T}}\vec{\varphi}_{ij}(\vec{\varphi}_{kl} \times \vec{n}_{ij}) ds$

Local EDO systems

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

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

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

- $\mathbf{G} = \mathbf{K} - \mathbf{A} - \mathbf{B}$
- $\mathbf{M}^{\varepsilon}$ 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)

Local EDO systems

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

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

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

- $\mathbf{G} = \mathbf{K} - \mathbf{A} - \mathbf{B}$
- $\mathbf{M}^{\varepsilon}$ 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)

- L. Fezoui, S. Lanteri, S. Lohrengel and S. Piperno
  ESAIM: M2AN, Vol. 39, No. 6, 2005
  Second order leap-frog time integration scheme, centered fluxes

Formulation: 2nd order Leap-Frog

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

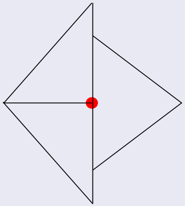
Stability analysis

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

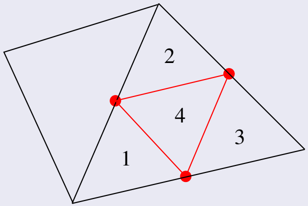
Outline

1. Time-domain electromagnetics
   - Overview of existing methods
   - Discontinuous Galerkin Time Domain (DGTD) method
   - Non-conforming DGTD-$\mathbb{P}_{p_i}$ method
   - Towards higher order time accuracy
   - Numerical treatment of grid-induced stiffness
   - 3D application: electromagnetic waves and humans
2. Frequency-domain electromagnetics
   - Numerical results in the 2D TMz case
   - Domain decomposition solver
3. High performance computing

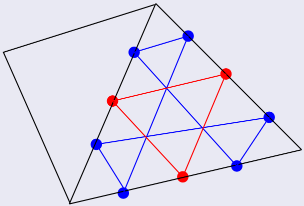
Non-conforming simplicial meshes



Non-conforming mesh



1-irregular mesh



3-irregular mesh

- Non-conforming refinement
  - Each triangle is split into 4 similar triangles
- Can be used for more flexibility in the discretization of:
  - complex domains,
  - heterogeneous media.
- Expected to reduce memory consumption and computing time

Non-conforming DGTD-$\mathbb{P}_{p_i}$ method

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

- Interface matrix (of size $d_i \times d_k$)

$$(\mathbf{S}_{ik})_{jl} = \frac{1}{2} \iint_{a_{ik}} {}^{\mathsf{T}}\vec{\varphi}_{ij} (\vec{\varphi}_{kl} \times \vec{n}_{ij}) ds$$

  - If a_{ik} is a conforming interface $\Rightarrow$ no problem
  - If a_{ik} is a non-conforming interface $\Rightarrow$ calculate S_{ik} using cubature formulas
    - ✘ 2D : Gauss-Legendre quadrature
    - ✘ 3D : Dunavant cubature formula

Stability analysis

- H. Fahs, L. Fezoui, S. Lanteri and F. Rapetti
  IEEE. Trans. Magn., Vol. 44, No. 6, 2008
  - Local stability condition

$$\forall i, \forall k \in \mathscr{V}_i, \quad c_i \Delta t [2\alpha_i + \beta_{ik}] < \frac{4V_i}{P_i}$$

where the dimensionless constants α_i and β_{ik} $(k \in \mathscr{V}_i)$ verify:

$$\forall \mathbf{X} \in \mathscr{P}_i = \mathrm{Span}(\vec{\varphi}_{ij}\ ,\ 1 \leq j \leq d_i)$$

$$\|\nabla \times \mathbf{X}\|_{\tau_i} \leq \frac{\alpha_i P_i}{V_i} \|\mathbf{X}\|_{\tau_i} \quad \text{and} \quad \|\mathbf{X}\|^2_{a_{ik}} \leq \frac{\beta_{ik} S_{ik}}{V_i} \|\mathbf{X}\|^2_{\tau_i}$$

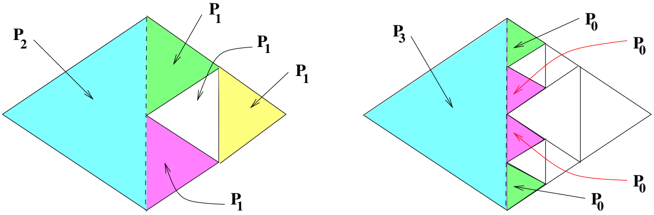
  - Numerical CFL values (second order Leap-Frog time scheme)

p	0	1	2	3	4	5	6	7	8	9
CFL	1.0	0.3	0.2	0.1	0.08	0.06	0.045	0.035	0.03	0.025

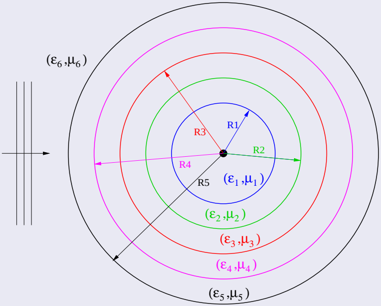
- 2D Maxwell equations (TMz)

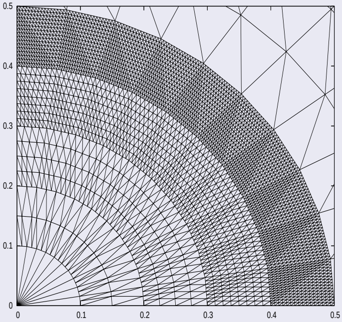
$$\left\{\begin{aligned} &\mu\frac{\partial H_x}{\partial t}+\frac{\partial E_z}{\partial y}=0\\ &\mu\frac{\partial H_y}{\partial t}-\frac{\partial E_z}{\partial x}=0\\ &\varepsilon\frac{\partial E_z}{\partial t}-\frac{\partial H_y}{\partial x}+\frac{\partial H_x}{\partial y}=0\end{aligned}\right.$$

- DGTD-$\mathbb{P}_{(p_1,p_2)}$ method
  - High polynomial degree p_1 in coarse elements
  - Low polynomial degrees p_2 in refined elements



Comparison between conforming and non-conforming methods





Region	L1	L2	L3	L4	L5	L6
Interpolation order	p_1	p_2	p_3	p_4	p_5	p_6
Level of refinement	0	1	2	3	4	0
# Triangles	40	320	1280	5120	20480	400

Comparison between conforming and non-conforming methods

- Reference solution is constructed in a very fine conforming mesh using DGTD-$\mathbb{P}_4$ method

DGTD-$\mathbb{P}_p$ method: conforming mesh

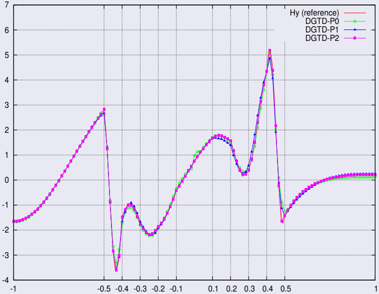
DGTD-$\mathbb{P}_p$	Error on H_y	CPU (min)	# DOF
DGTD-$\mathbb{P}_0$	8.6 %	25	28560
DGTD-$\mathbb{P}_1$	7.6 %	137	85680
DGTD-$\mathbb{P}_2$	2.2 %	286	171360
DGTD-$\mathbb{P}_3$	2.2 %	842	285600

DGTD-$\mathbb{P}_{p_i}$ method: non-conforming mesh

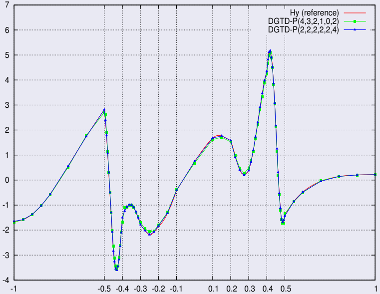
DGTD-$\mathbb{P}_{(p_1,p_2,p_3,p_4,p_5,p_6)}$	Error on H_y	CPU (min)	# DOF
DGTD-$\mathbb{P}_{(4,3,2,1,0,2)}$	5.0 %	12	49720
DGTD-$\mathbb{P}_{(4,3,2,2,0,2)}$	4.8 %	13	65080
DGTD-$\mathbb{P}_{(4,3,2,2,1,4)}$	3.5 %	17	109640
DGTD-$\mathbb{P}_{(4,2,2,4,1,4)}$	3.2 %	21	154440
DGTD-$\mathbb{P}_{(2,2,2,2,2,4)}$	2.5 %	20	169440

Comparison between conforming and non-conforming methods

DGTD-$\mathbb{P}_p$ method
Conforming mesh



DGTD-$\mathbb{P}_{p_i}$ method
Non-conforming mesh



1D distribution of H_y along $y = 0.0$

Outline

1. Time-domain electromagnetics
   - Overview of existing methods
   - Discontinuous Galerkin Time Domain (DGTD) method
   - Non-conforming DGTD-$\mathbb{P}_{p_i}$ method
   - Towards higher order time accuracy
   - Numerical treatment of grid-induced stiffness
   - 3D application: electromagnetic waves and humans
2. Frequency-domain electromagnetics
   - Numerical results in the 2D TMz case
   - Domain decomposition solver
3. High performance computing

- H. Spachmann, R. Schuhmann and T. Weiland
  Int. J. Numer. Model., 2002

Formulation: 2nd versus 4th order Leap-Frog

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

$$
\mathbf{G}_N = \left\{
\begin{array}{lll}
\mathbf{G} & \text{if} & N=2 \\
\mathbf{G}\left(\mathbf{I}-\dfrac{\Delta t^2}{24}\mathbf{M}^{-\mu}\,{}^{\mathsf{T}}\mathbf{G}\mathbf{M}^{-\varepsilon}\mathbf{G}\right) & \text{if} & N=4
\end{array}
\right.
$$

Stability analysis

- H. Fahs and S. Lanteri
  J. Comput. Appl. Math., Vol. 6, No. 2, 2009
- Discrete electromagnetic energy

$$\mathscr{E}^n = {}^{\mathrm{T}}\mathbb{E}^n \mathbf{M}^{\varepsilon} \mathbb{E}^n + {}^{\mathrm{T}}\mathbb{H}^{n+\frac{1}{2}} \mathbf{M}^{\mu} \mathbb{H}^{n-\frac{1}{2}}$$

- Condition for $\mathscr{E}^n$ being a positive definite form

$$\Delta t \leq \frac{2}{d_N}, \quad \text{with} \quad d_N = \| \, (\mathbf{M}^{-\mu})^{\frac{1}{2}} \; {}^{\mathrm{T}}\mathbf{G}_N \, (\mathbf{M}^{-\varepsilon})^{\frac{1}{2}} \, \|$$

$$\nu_N = \mathrm{CFL}(\mathrm{LF}_N)/\mathrm{CFL}(\mathrm{LF}_2)$$

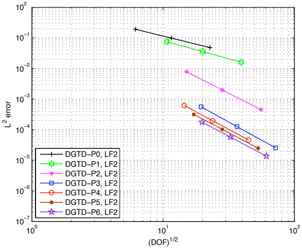
N	2	4	6	8	10	12	14	16	18	20
ν_N	1.0	2.85	3.68	3.79	5.27	4.44	6.42	7.53	7.27	8.91

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

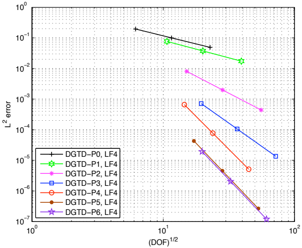
p	0	1	2	3	4	5
LF_2	1.06	1.19	2.18	2.37	2.29	2.25
LF_4	1.06	1.14	2.23	3.03	4.30	4.50

Asymptotic h-convergence orders of the DGTD-$\mathbb{P}_{p_i}$ method

LF_2 scheme



LF_4 scheme



Global (space and time) L^2 error versus the square root of # DOF

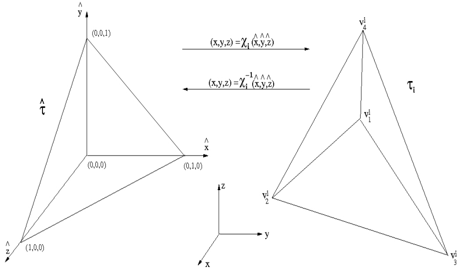
Numerical results in 2D: eigenmode in a square PEC cavity

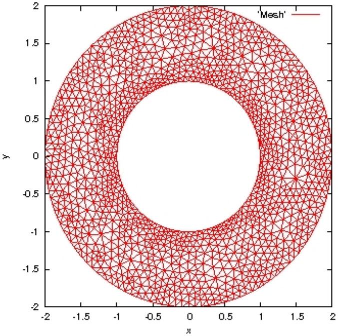
		LF_2 scheme		LF_4 scheme	
p	# DOF	Error	CPU time (min)	Error	CPU time (min)
2	4692	1.8×10^{-3}	11	5.5×10^{-4}	8
3	7820	3.1×10^{-4}	39	2.4×10^{-5}	28
4	11730	1.9×10^{-4}	98	1.5×10^{-5}	70
5	16422	1.5×10^{-4}	220	1.3×10^{-5}	155

L^2 error, CPU time in seconds and # DOF

Standard approach: affine map

- Elementary matrices (mass matrix, pseudo-stiffness matrix, etc.) are computed on the reference element for each interpolation degree p
- An affine transformation χ_i is used to obtain these matrices on a physical ement



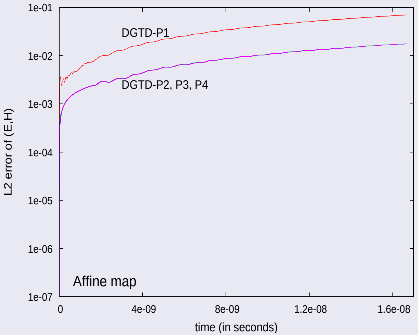


High order geometrical maps

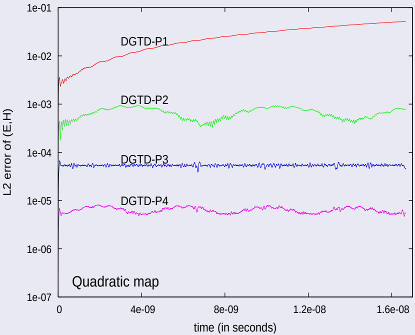
- With the affine map, for interpolation degree $p \geq 2$, the global error is then limited by the approximation error
- Curvilinear elements combined to quadratic or cubic maps are used on selected elements

2D Maxwell equations (TMz): eigenmode in a circular PEC cavity

Affine map

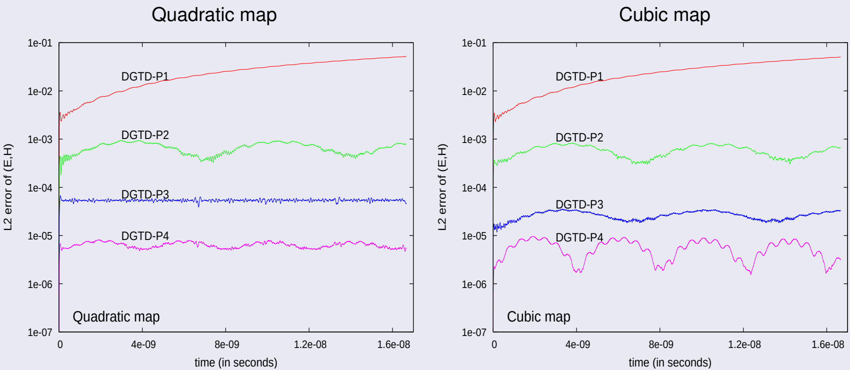


Quadratic map



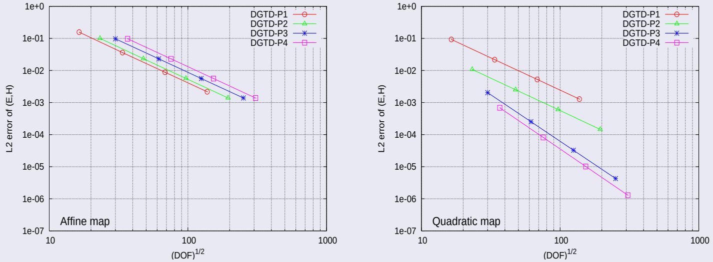
Time evolution of the L^2 error

2D Maxwell equations (TMz): eigenmode in a circular PEC cavity



Time evolution of the L^2 error

2D Maxwell equations (TMz): eigenmode in a circular PEC cavity



Global (space and time) L^2 error versus the square root of # DOF

2D Maxwell equations (TMz): eigenmode in a circular PEC cavity

p	Affine map	Quadratic map	Cubic map
1	2.00	2.01	2.02
2	2.00	2.03	2.05
3	2.00	2.90	3.01
4	2.00	2.95	3.47

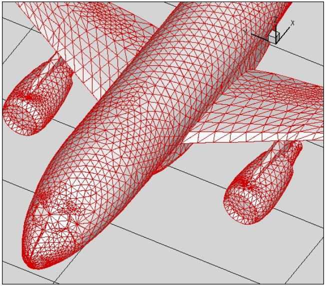
Asymptotic h-convergence orders of the DGTD-$\mathbb{P}_p$ method

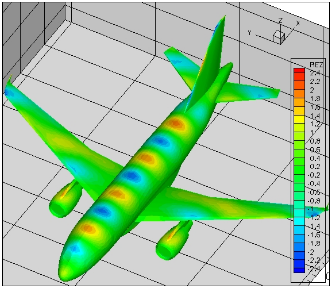
Outline

1. Time-domain electromagnetics
   - Overview of existing methods
   - Discontinuous Galerkin Time Domain (DGTD) method
   - Non-conforming DGTD-$\mathbb{P}_{p_i}$ method
   - Towards higher order time accuracy
   - Numerical treatment of grid-induced stiffness
   - 3D application: electromagnetic waves and humans
2. Frequency-domain electromagnetics
   - Numerical results in the 2D TMz case
   - Domain decomposition solver
3. High performance computing

Numerical treatment of grid-induced stiffness

- Scattering of a plane wave by an aircraft, F=1 GHz
  - Mesh: # vertices = 153,821 , # tetrahedra = 883,374
  - $L_{\text{min}} = 0.000601$ m , $L_{\text{max}} = 0.121290$ m $(\approx \dfrac{\lambda}{2.5})$, $L_{\text{avg}} = 0.039892$ m
  - $\Delta t_{\text{min}} =$ 0.24 picosec and $\Delta t_{\text{max}} = 40.50$ picosec





Possible routes to overcome grid-induced stiffness

- Local time-step strategies with explicit time integration
- Locally implicit (hybrid explicit/implicit) time integration

- S. Piperno, ESAIM: M2AN, Vol. 40, No. 5, 2006
  - Explict scheme: Verlet method (i.e. three-step Leap-Frog method with **E** and **H** computed at the same time stations)
  - Implicit scheme: Crank-Nicolson scheme
- Partitioning of the mesh elements (triangles/tetrahedra) into two subsets
  - $\mathscr{S}_e$: coarsest elements, treated explicitly
  - $\mathscr{S}_i$: smallest elements, treated implicitly

$$\mathbb{E} = \begin{pmatrix} \mathbb{E}_e \\ \mathbb{E}_i \end{pmatrix} \ , \ \mathbb{H} = \begin{pmatrix} \mathbb{H}_e \\ \mathbb{H}_i \end{pmatrix}$$

$$\mathbf{M}^{\varepsilon} = \begin{pmatrix} \mathbf{M}_e^{\varepsilon} & \mathbf{O} \\ \mathbf{O} & \mathbf{M}_i^{\varepsilon} \end{pmatrix} \ , \ \mathbf{M}^{\mu} = \begin{pmatrix} \mathbf{M}_e^{\mu} & \mathbf{O} \\ \mathbf{O} & \mathbf{M}_i^{\mu} \end{pmatrix}$$

$$\mathbf{K} = \begin{pmatrix} \mathbf{K}_e & \mathbf{O} \\ \mathbf{O} & \mathbf{K}_i \end{pmatrix} \ , \ \mathbf{B} = \begin{pmatrix} \mathbf{B}_e & \mathbf{O} \\ \mathbf{O} & \mathbf{B}_i \end{pmatrix}$$

- $\mathbf{A}$: matrix corresponding to fluxes at cell interfaces
  - $\mathbf{A}_{ee}$ and $\mathbf{A}_{ii}$ are symmetric matrices
  - $\mathbf{A}_{ei} = {}^{\mathsf{T}}\mathbf{A}_{ie}$

$$\mathbf{A} = \begin{pmatrix} \mathbf{A}_{ee} & \mathbf{A}_{ei} \\ \mathbf{A}_{ie} & \mathbf{A}_{ii} \end{pmatrix}$$

- $\mathbf{G}_e = \mathbf{K}_e - \mathbf{A}_{ee} - \mathbf{B}_e$
- $\mathbf{G}_i = \mathbf{K}_i - \mathbf{A}_{ii} - \mathbf{B}_i$
- $\mathbf{G}_e$ and $\mathbf{G}_i$ are symmetric matrices

$$\left\{ \begin{array}{rcl} \mathbf{M}^{\varepsilon}\dfrac{d\mathbb{E}}{dt} & = & \mathbf{G}\mathbb{H} \\ \mathbf{M}^{\mu}\dfrac{d\mathbb{H}}{dt} & = & -{}^{\mathsf{T}}\mathbf{G}\mathbb{E} \end{array} \right. \iff \begin{array}{l} \left\{ \begin{array}{rcl} \mathbf{M}^{\varepsilon}_e\dfrac{d\mathbb{E}_e}{dt} & = & \mathbf{G}_e\mathbb{H}_e - \mathbf{A}_{ei}\mathbb{H}_i \\ \mathbf{M}^{\mu}_e\dfrac{d\mathbb{H}_e}{dt} & = & -{}^{\mathsf{T}}\mathbf{G}_e\mathbb{E}_e + \mathbf{A}_{ei}\mathbb{E}_i \end{array} \right. \\ \left\{ \begin{array}{rcl} \mathbf{M}^{\varepsilon}_i\dfrac{d\mathbb{E}_i}{dt} & = & \mathbf{G}_i\mathbb{H}_i - \mathbf{A}_{ie}\mathbb{H}_e \\ \mathbf{M}^{\mu}_i\dfrac{d\mathbb{H}_i}{dt} & = & -{}^{\mathsf{T}}\mathbf{G}_i\mathbb{E}_i + \mathbf{A}_{ie}\mathbb{E}_e \end{array} \right. \end{array}$$

Formulation

$$\left\{\begin{array}{lcl}
\mathbf{M}_e^{\mu}\left(\dfrac{\mathbb{H}_e^{n+\frac{1}{2}}-\mathbb{H}_e^{n}}{\Delta t/2}\right) & = & -{}^{\mathsf{T}}\mathbf{G}_e\mathbb{E}_e^{n}+\mathbf{A}_{ei}\mathbb{E}_i^{n} \\
\mathbf{M}_e^{\varepsilon}\left(\dfrac{\mathbb{E}_e^{n+\frac{1}{2}}-\mathbb{E}_e^{n}}{\Delta t/2}\right) & = & \mathbf{G}_e\mathbb{H}_e^{n+\frac{1}{2}}-\mathbf{A}_{ei}\mathbb{H}_i^{n}
\end{array}\right.$$

$$\left\{\begin{array}{lcl}
\mathbf{M}_i^{\varepsilon}\left(\dfrac{\mathbb{E}_i^{n+1}-\mathbb{E}_i^{n}}{\Delta t}\right) & = & \mathbf{G}_i\left(\dfrac{\mathbb{H}_i^{n+1}+\mathbb{H}_i^{n}}{2}\right)-\mathbf{A}_{ie}\mathbb{H}_e^{n+\frac{1}{2}} \\
\mathbf{M}_i^{\mu}\left(\dfrac{\mathbb{H}_i^{n}-\mathbb{H}_i^{n+1}}{\Delta t}\right) & = & -{}^{\mathsf{T}}\mathbf{G}_i\left(\dfrac{\mathbb{E}_i^{n}+\mathbb{E}_i^{n+1}}{2}\right)+\mathbf{A}_{ie}\mathbb{E}_e^{n+\frac{1}{2}}
\end{array}\right.$$

$$\left\{\begin{array}{lcl}
\mathbf{M}_e^{\varepsilon}\left(\dfrac{\mathbb{E}_e^{n+1}-\mathbb{E}_e^{n+\frac{1}{2}}}{\Delta t/2}\right) & = & \mathbf{G}_e\mathbb{H}_e^{n+\frac{1}{2}}-\mathbf{A}_{ei}\mathbb{H}_i^{n+1} \\
\mathbf{M}_e^{\mu}\left(\dfrac{\mathbb{H}_e^{n+1}-\mathbb{H}_e^{n+\frac{1}{2}}}{\Delta t/2}\right) & = & -{}^{\mathsf{T}}\mathbf{G}_e\mathbb{E}_e^{n+1}+\mathbf{A}_{ei}\mathbb{E}_i^{n+1}
\end{array}\right.$$

Stability analysis

- V. Dolean, H. Fahs, L. Fezoui and S. Lanteri
  J. Comput. Phys., Vol. 229, No. 2, 2010
- Discrete electromagnetic energy

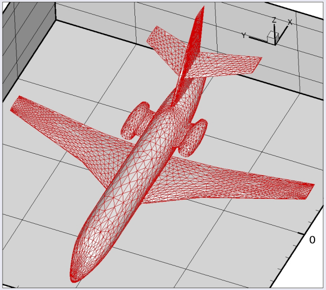
$$\mathscr{E}^n = \mathscr{E}_e^n + \mathscr{E}_i^n + \mathscr{E}_h^n \quad \text{with} \quad \left\{ \begin{array}{lcl} \mathscr{E}_e^n & = & {}^{\mathrm{T}}\mathbb{E}_e^n \mathbf{M}_e^{\varepsilon} \mathbb{E}_e^n + {}^{\mathrm{T}}\mathbb{H}_e^{n+\frac{1}{2}} \mathbf{M}_e^{\mu} \mathbb{H}_e^{n-\frac{1}{2}} \\ \mathscr{E}_i^n & = & {}^{\mathrm{T}}\mathbb{E}_i^n \mathbf{M}_i^{\varepsilon} \mathbb{E}_i^n + {}^{\mathrm{T}}\mathbb{H}_i^n \mathbf{M}_i^{\mu} \mathbb{H}_i^n \\ \mathscr{E}_h^n & = & -\dfrac{\Delta t^2}{4} {}^{\mathrm{T}}\mathbb{H}_i^n {}^{\mathrm{T}}\mathbf{A}_{\mathbf{ei}} (\mathbf{M}_{\mathbf{e}}^{\varepsilon})^{-\mathbf{1}} \mathbf{A}_{\mathbf{ei}} \mathbb{H}_{\mathbf{i}}^{\mathbf{n}} \end{array} \right.$$

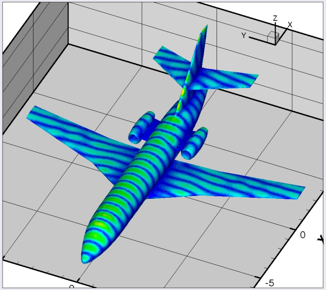
- Condition for $\mathscr{E}^n$ being a positive definite form

$$\Delta t \leq \frac{2}{\alpha_e + \max(\beta_{ei}, \gamma_{ei})} \quad \text{with} \quad \left\{ \begin{array}{lcl} \alpha_e & = & \| \, (\mathbf{M}_e^{\varepsilon})^{-\frac{1}{2}} \mathbf{G}_e (\mathbf{M}_e^{\mu})^{-\frac{1}{2}} \, \| \\ \beta_{ei} & = & \| \, (\mathbf{M}_e^{\varepsilon})^{-\frac{1}{2}} \mathbf{A}_{ei} (\mathbf{M}_i^{\mu})^{-\frac{1}{2}} \, \| \\ \gamma_{ei} & = & \| \, (\mathbf{M}_e^{\mu})^{-\frac{1}{2}} \mathbf{A}_{ei} (\mathbf{M}_i^{\varepsilon})^{-\frac{1}{2}} \, \| \end{array} \right.$$

Numerical results

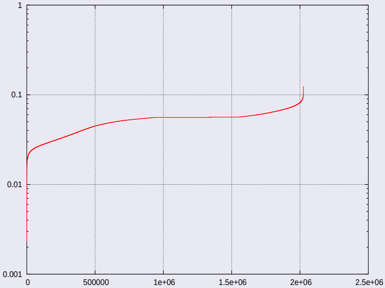
- Scattering of plane wave (F=200 MHz, $\lambda = 1.5$ m) by an aircraft
- # vertices=360,495 and # elements=2,024,924
- Edges length: L_m=9.166 10^{-3} m ($\approx \lambda/163$ m) and L_M=6.831 10^{-1} m ($\approx \lambda/2.2$ m)

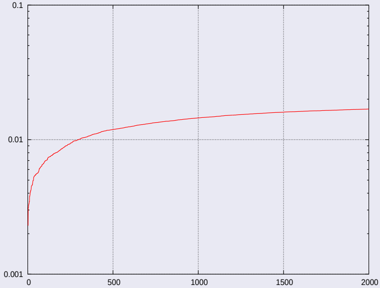




Numerical results

- Scattering of plane wave (F=200 MHz, $\lambda = 1.5$ m) by an aircraft
- Geometric criterion: $\mathscr{C}(\tau_i) = 4 \min_{j \in \mathscr{V}_i} \sqrt{\dfrac{V_i V_j}{P_i P_j}}$
  - V_i , P_i: volume and perimeter of τ_i





Distribution of the geometric criterion $\mathscr{C}$

Numerical results

- Scattering of plane wave (F=200 MHz, $\lambda = 1.5$ m) by an aircraft

$\mathscr{C}$max	$\lvert\mathscr{S}_e\rvert$	$\lvert\mathscr{S}_i\rvert$
0.0125	2,024,320	604 (0.03 %)
0.0175	2,022,464	2,460 (0.12 %)
0.02	2,018,543	6,381 (0.31 %)

Definition of the subsets of explicit and implicit elements

$\mathscr{C}$max	RAM (LU)	Time (LU)	Time (total)
0.0125 m	12 MB	0.3 sec	6 h 39 mn
0.0175 m	48 MB	1.5 sec	4 h 44 mn
0.02 m	117 MB	4.2 sec	4 h 08 mn

Hybrid explicit-implicit DGTD-$\mathbb{P}_1$ method (Intel Xeon/2.33 GHz workstation)

Fully explicit DGTD-$\mathbb{P}_1$ method: 25 h 3 mn

Outline

1. Time-domain electromagnetics
   - Overview of existing methods
   - Discontinuous Galerkin Time Domain (DGTD) method
   - Non-conforming DGTD-$\mathbb{P}_{p_i}$ method
   - Towards higher order time accuracy
   - Numerical treatment of grid-induced stiffness
   - 3D application: electromagnetic waves and humans

2. Frequency-domain electromagnetics
   - Numerical results in the 2D TMz case
   - Domain decomposition solver

3. High performance computing

3D application: electromagnetic waves and humans

EM waves in our environment

- Natural sources (earth magnetic field, etc.)
- Manmade sources
  - Domestic appliances: TV, radio, microwave ovens, hairdryers, fridges, etc.
  - Technological devices: mobile phones, Wi-Fi, etc.

Characterization of EM fields and related effects

- An EM field is characterized by its frequency (Hz, MHz, GHz)
- Ionising radiation
  - Upper part of the frequency spectrum
  - Can induce changes at the molecular level
  - x-rays and gamma rays
- Non-ionising radiation
  - Lower part of the frequency spectrum
  - Static and power frequency fields, radiofrequencies, microwaves and infrared radiation

3D application: electromagnetic waves and humans

Basic physiological processes

- Energy from RF and MW is absorbed into the body

  SAR (Specific Absorption Rate): $\dfrac{\sigma |\mathbf{E}|^2}{\rho}$

- Energy is converted to heat
- Energy is dissipated by the body's normal thermoregulatory process

Health issues related to hand-held mobile phones

- Biological effects versus sanitary effects
  - Biological effects: physiological, biochemical or behavioral changes induced in a body, tissue or cell by an external source
  - A biological effect does not necessarily represent a risk for human health
  - Sanitary effects: consequences of biological effects that change the normal behavior of a body
- Thermal effects versus non-thermal effects
  - A thermal effect results from a local or systemic heating of a tissue
  - Thermal effects are relatively well known
  - Ongoing studies are concerned with non-thermal effects

Basic physiological processes

- Energy from RF and MW is absorbed into the body

SAR (Specific Absorption Rate): $\dfrac{\sigma|\mathbf{E}|^2}{\rho}$

- Energy is converted to heat
- Energy is dissipated by the body's normal thermoregulatory process

Health issues related to hand-held mobile phones

- Biological effects versus sanitary effects
  - Biological effects: physiological, biochemical or behavioral changes induced in a body, tissue or cell by an external source
  - A biological effect does not necessarily represent a risk for human health
  - Sanitary effects: consequences of biological effects that change the normal behavior of a body
- Thermal effects versus non-thermal effects
  - A thermal effect results from a local or systemic heating of a tissue
  - Thermal effects are relatively well known
  - Ongoing studies are concerned with non-thermal effects

Societal context

Year	2003	2004	2005	2006	2007	2008	2009
# users ($\times 10^6$)	41.6	43.8	48.1	51.7	55.4	58.1	61.4
% active population	69.1	72.6	78.4	80.8	85.6	89.1	95.8

As of 2008,

- 71% of 12-14 years old kids owned a mobile phone,
- 95% coverage of the 15-17 years olds.

Health issues related to hand-held mobile phones

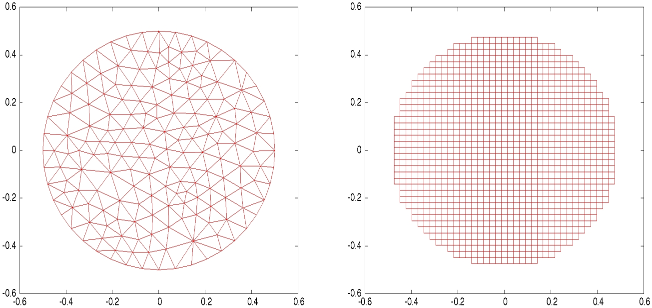
- Epidemiological studies
  - Possible links with various cancers
- Experimental studies
  - Dosimetry of animal exposure
  - In vivo and in vitro studies
- Computer simulation studies
  - Numerical dosimetry of EM fields
  - Evaluation of temperature elevation in tissues
    - ε, σ and ρ are varying from one tissue to the other
    - They also depend on the frequency of the signal
    - Discontinuities of **E** and **H** occur at interfaces between different tissues

FDM (Finite Difference Methods)

- Advantages
  - Easy computer implementation
  - Computationally efficient (very low algorithmic complexity)
  - Mesh generation is straightforward (medical images are voxel based)
  - 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 (staircase representation)
- Numerical dosymetry analysis of mobile phones radiation most often relies on the FDTD method
  - P. Bernardi *et al.* (U. La Sapienza, Roma, Italy)
  - O.P. Gandhi *et al.* (U. of Utah, USA)
  - J. Wiart *et al.* (FTR&D, France)
  - etc.

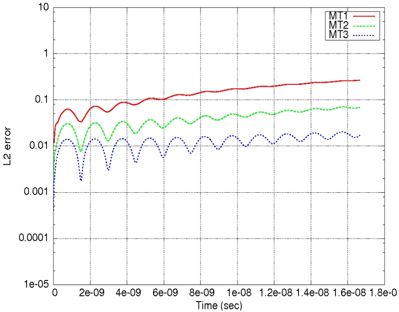
Numerical assessment of the staircasing effect

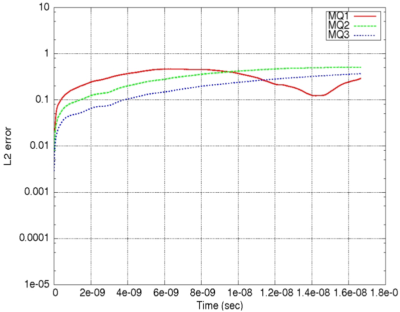
- Standing wave in a 2D circular PEC cavity
- Triangular mesh: DGTD-$\mathbb{P}_p$ method
- Quadrangular mesh: DGTD-$\mathbb{Q}_p$ method



Numerical assessment of the staircasing effect

- Standing wave in a 2D circular PEC cavity
- Triangular mesh: DGTD-$\mathbb{P}_p$ method
- Quadrangular mesh: DGTD-$\mathbb{Q}_p$ method

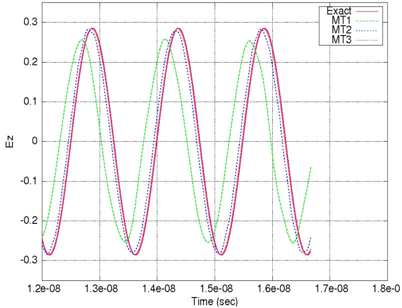


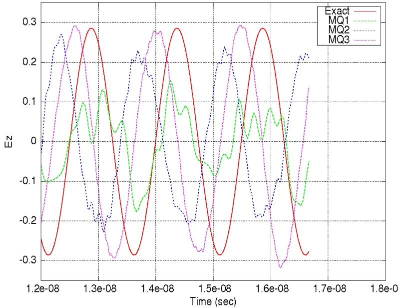


Time evolution of the L^2 error

Numerical assessment of the staircasing effect

- Standing wave in a 2D circular PEC cavity
- Triangular mesh: DGTD-$\mathbb{P}_p$ method
- Quadrangular mesh: DGTD-$\mathbb{Q}_p$ method

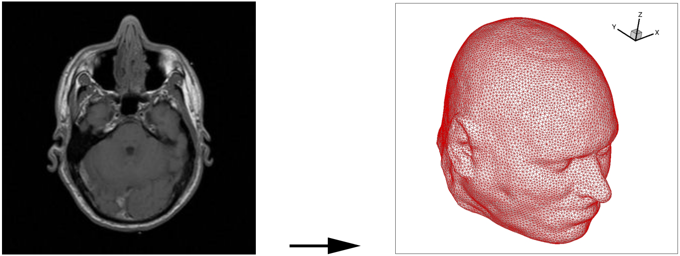




Time evolution of the E_z component at a selected point

3D application: electromagnetic waves and humans

Numerical dosimetry of EM fields



Geometric models

- Built from segmented medical images
- Extraction of surfacic (triangular) meshes of the tissue interfaces using specific tools
    - Marching cubes + adaptive isotropic surface remeshing
    - Delaunay refinement
- Generation of tetrahedral meshes using a Delaunay/Voronoi tool

Coarse mesh (M1)

- # vertices = 135,633 and # tetrahedra = 781,742

Tissue	L_{min} (mm)	L_{max} (mm)	L_{moy} (mm)	λ (mm)
Skin	1.339	8.055	4.070	26.73
Skull	1.613	7.786	4.069	42.25
CSF	0.650	7.232	4.059	20.33
Brain	0.650	7.993	4.009	25.26

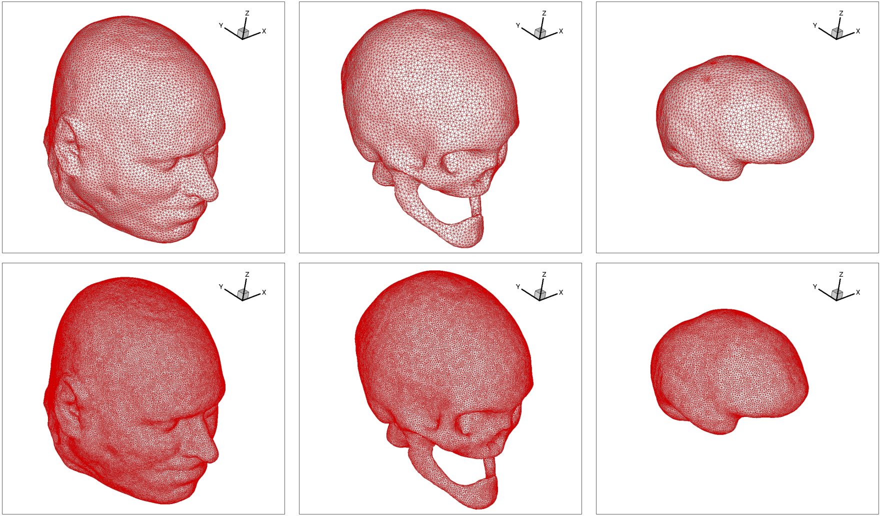
Fine mesh (M2)

- # vertices = 889,960 and # tetrahedra = 5,230,947

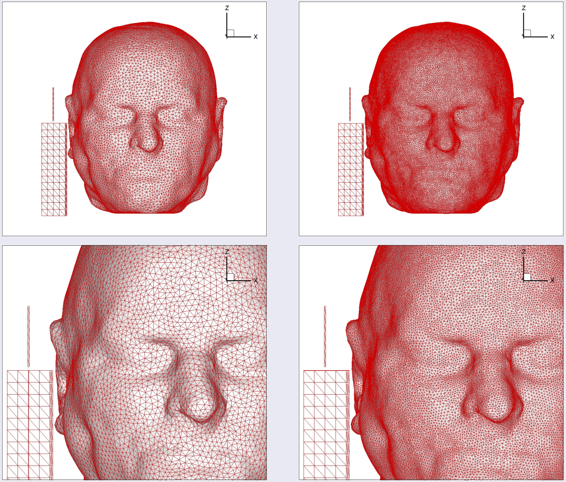
Tissue	L_{min} (mm)	L_{max} (mm)	L_{moy} (mm)	λ (mm)
Skin	0.821	5.095	2.113	26.73
Skull	0.776	4.265	2.040	42.25
CSF	0.909	3.701	1.978	20.33
Brain	0.915	5.509	2.364	25.26

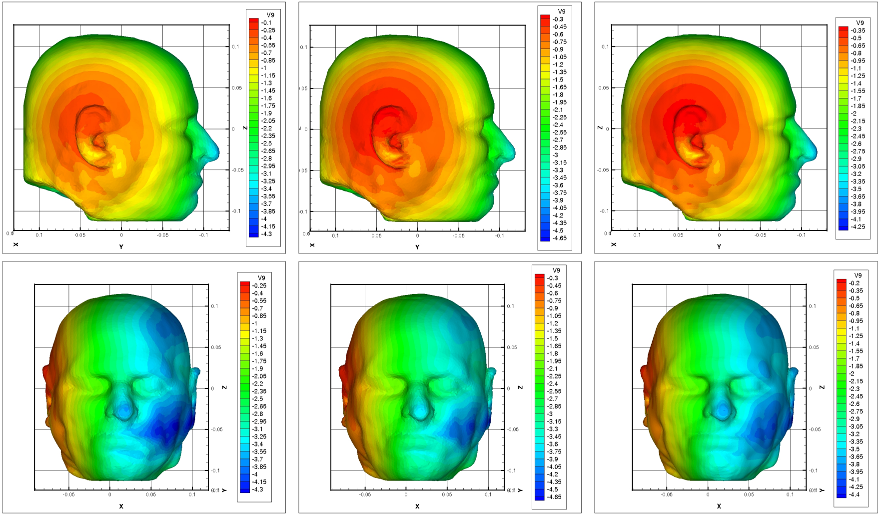
3D application: electromagnetic waves and humans

Numerical dosimetry of EM fields

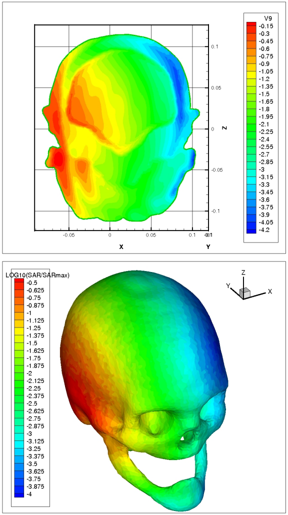


Head + simplified phone model

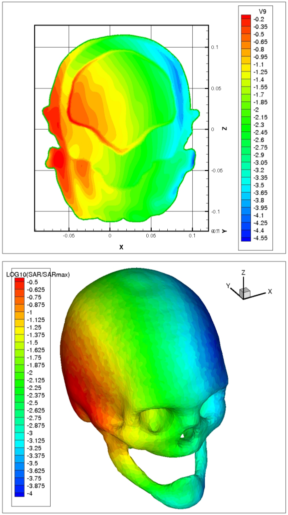




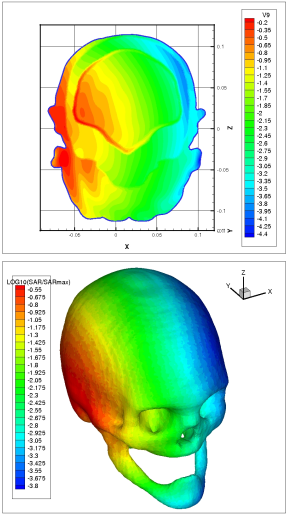
M1, DGTD-$\mathbb{P}_1$ M1, DGTD-$\mathbb{P}_2$ M2, DGTD-$\mathbb{P}_1$



M1, DGTD-$\mathbb{P}_1$



M1, DGTD-$\mathbb{P}_2$



M2, DGTD-$\mathbb{P}_1$

Outline

1. Time-domain electromagnetics
   - Overview of existing methods
   - Discontinuous Galerkin Time Domain (DGTD) method
   - Non-conforming DGTD-$\mathbb{P}_{p_i}$ method
   - Towards higher order time accuracy
   - Numerical treatment of grid-induced stiffness
   - 3D application: electromagnetic waves and humans
2. Frequency-domain electromagnetics
   - Numerical results in the 2D TMz case
   - Domain decomposition solver
3. High performance computing

$$\varepsilon \mathrm{i}\omega \mathbf{E} - \mathrm{rot}(\mathbf{H}) = -z_0 \mathbf{J} \ , \ \ \mu \mathrm{i}\omega \mathbf{H} + \mathrm{rot}(\mathbf{E}) = 0$$

- $\mathbf{E} = \mathbf{E}(\mathbf{x})$ is the electric field and $\mathbf{H} = \mathbf{H}(\mathbf{x})$ is the magnetic field
- $\mathbf{J} = \mathbf{J}(\mathbf{x})$ is the conductive current : $\mathbf{J} = \sigma \mathbf{E}$ $\left(z_0 = \sqrt{\dfrac{\mu_0}{\varepsilon_0}}\right)$
- $\varepsilon = \varepsilon(\mathbf{x})$: (relative) electric permittivity
- $\mu = \mu(\mathbf{x})$: (relative) magnetic permeability
- $\sigma = \sigma(\mathbf{x})$: electric conductivity
- Boundary conditions
  - PEC boundary : $\mathbf{n} \times \mathbf{E} = 0$
  - Absorbing boundary : $\mathbf{n} \times \mathbf{E} + z\mathbf{n} \times (\mathbf{n} \times \mathbf{H}) = \mathbf{n} \times \mathbf{E}^{\infty} + z\mathbf{n} \times (\mathbf{n} \times \mathbf{H}^{\infty})$

Pseudo-conservative system form

$$\mathrm{i}\omega Q\mathbf{W} + \nabla \cdot F(\mathbf{W}) = \mathbf{S} \ \text{ with } \ \mathbf{W} = {}^t(\mathbf{E},\mathbf{H}) \ \text{ and } \ \mathbf{S} = {}^t(-z_0\mathbf{J}, 0_{3\times 1})$$

- Triangulation: $\mathscr{T}_h = \bigcup_{i=1}^{N} \tau_i$
- Assume $\mathbf{J} = 0$ for simplicity of the presentation
- $\mathbf{W}_i(\mathbf{x}) \in \mathscr{P}_i = \mathbb{P}_m[\tau_i]$ and $\mathbf{W}_i(\mathbf{x}) = \sum_{j=1}^{d_i} \mathbf{W}_{ij}\varphi_{ij}(\mathbf{x})$ with $\mathbf{W}_{ij} \in \mathbb{C}^6$

$$\int_{\tau_i} \varphi \left(\mathbf{i}\omega Q\mathbf{W} + \nabla \cdot F(\mathbf{W}) \right) d\mathbf{x} = 0$$

$$\Leftrightarrow \quad \int_{\tau_i} \mathbf{i}\omega Q\mathbf{W}\varphi d\mathbf{x} - \int_{\tau_i} \nabla\varphi \cdot F(\mathbf{W}) d\mathbf{x} + \int_{\partial\tau_i} \left(F(\mathbf{W}) \cdot \mathbf{n} \right) \varphi d\sigma = 0$$

- Calculation of the boundary term on $\partial\tau_i$: centered or upwind numerical flux

Outline

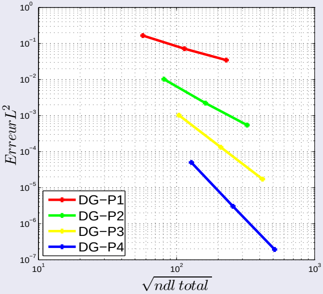
1. Time-domain electromagnetics
   - Overview of existing methods
   - Discontinuous Galerkin Time Domain (DGTD) method
   - Non-conforming DGTD-$\mathbb{P}_{p_i}$ method
   - Towards higher order time accuracy
   - Numerical treatment of grid-induced stiffness
   - 3D application: electromagnetic waves and humans
2. Frequency-domain electromagnetics
   - Numerical results in the 2D TMz case
   - Domain decomposition solver
3. High performance computing

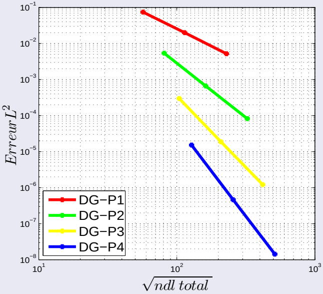
$$
\left\{
\begin{aligned}
&\mu \mathbf{i}\omega H_x + \frac{\partial E_z}{\partial y} = 0\\
&\mu \mathbf{i}\omega H_y - \frac{\partial E_z}{\partial x} = 0\\
&\varepsilon \mathbf{i}\omega E_z - \frac{\partial H_y}{\partial x} + \frac{\partial H_x}{\partial y} = 0
\end{aligned}
\right.
$$

- DGFD-$\mathbb{P}_p$ method based on Lagrange (nodal) interpolation
  - Triangular mesh
  - Sparse block matrix, $3n_p \times 3n_p$ (with $n_p = ((p+1)(p+2))/2$)
  - MUMPS multifrontal sparse matrix solver
    (P.R. Amestoy, I.S. Duff and J.-Y. L'Excellent, CMAME, Vol. 184, 2000)

Numerical convergence of the DGFD-$\mathbb{P}_p$ method

- Plane wave in vacuum, F=300 MHz
- Non-uniform triangular meshes

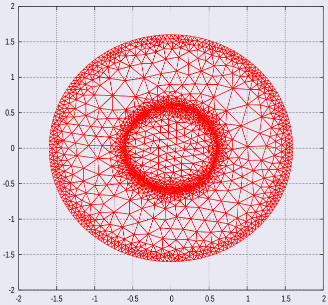




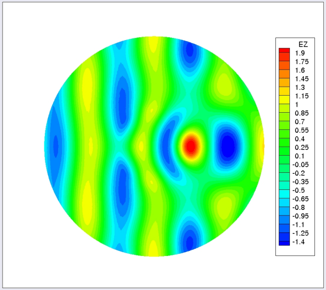
	$\mathbb{P}_1$	$\mathbb{P}_2$	$\mathbb{P}_3$	$\mathbb{P}_4$
Centered flux	1.1	2.1	2.9	4.0
Upwind flux	1.9	3.0	4.0	5.0

Scattering of a plane wave by a dielectric cylinder, F=300 MHz

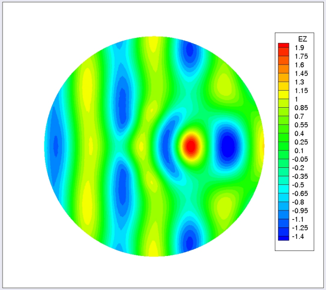
- # vertices = 2078 and # elements = 3958
- Comparison between conforming DGFD-$\mathbb{P}_p$ and DGFD-$\mathbb{P}_{p_i}$ methods



Scattering of a plane wave by a dielectric cylinder, F=300 MHz



$\mathbb{P}_4$



$\mathbb{P}_{p_i}$ $(i = 1,2,3,4)$

DGFD-$\mathbb{P}_4$ and non-conforming DGFD-$\mathbb{P}_{p_i}$ methods
Contour lines of E_z

- Scattering of a plane wave by a dielectric cylinder, F=300 MHz
- Centered numerical flux

Conforming DGFD-$\mathbb{P}_p$ methods

n_z	Method	L2 error on E_z	CPU	RAM LU
390,274	DGFD-$\mathbb{P}_1$	0.37977	1.3 sec	97 MB
1,186,224	DGFD-$\mathbb{P}_2$	0.05830	4.1 sec	255 MB
3,225,808	DGFD-$\mathbb{P}_3$	0.05527	7.9 sec	547 MB
7,033,834	DGFD-$\mathbb{P}_4$	0.05522	15.7 sec	954 MB

Non-conforming DGFD-$\mathbb{P}_{p_i}$ method

n_z	Method	L2 error on E_z	CPU	RAM LU
1,267,878	DGFD-$\mathbb{P}_{1,4}$	0.05586	3.7 sec	252 MB

$\mathbb{P}_1$	$\mathbb{P}_2$	$\mathbb{P}_3$	$\mathbb{P}_4$
1495	2037	243	183

Local definition of p_i based on the value of a triangle area

- Scattering of a plane wave by a dielectric cylinder, F=300 MHz
- Centered numerical flux

Conforming DGFD-$\mathbb{P}_p$ methods

n_z	Method	L2 error on E_z	CPU	RAM LU
390,274	DGFD-$\mathbb{P}_1$	0.37977	1.3 sec	97 MB
1,186,224	DGFD-$\mathbb{P}_2$	0.05830	4.1 sec	255 MB
3,225,808	DGFD-$\mathbb{P}_3$	0.05527	7.9 sec	547 MB
7,033,834	DGFD-$\mathbb{P}_4$	0.05522	15.7 sec	954 MB

Non-conforming DGFD-$\mathbb{P}_{p_i}$ method

n_z	Method	L2 error on E_z	CPU	RAM LU
1,267,878	DGFD-$\mathbb{P}_{1,4}$	0.05586	3.7 sec	252 MB

$\mathbb{P}_1$	$\mathbb{P}_2$	$\mathbb{P}_3$	$\mathbb{P}_4$
1495	2037	243	183

Local definition of p_i based on the value of a triangle area

Outline

1. Time-domain electromagnetics
   - Overview of existing methods
   - Discontinuous Galerkin Time Domain (DGTD) method
   - Non-conforming DGTD-$\mathbb{P}_{p_i}$ method
   - Towards higher order time accuracy
   - Numerical treatment of grid-induced stiffness
   - 3D application: electromagnetic waves and humans

2. Frequency-domain electromagnetics
   - Numerical results in the 2D TMz case
   - **Domain decomposition solver**

3. High performance computing

Time harmonic Maxwell system

$$\mathscr{L}\mathbf{W} = \mathrm{i}\omega G_0\mathbf{W} + G_x\partial_x\mathbf{W} + G_y\partial_y\mathbf{W} + G_z\partial_z\mathbf{W} - \mathbf{S} = 0$$

- Flux matrices

$$G_l = \begin{bmatrix} 0_{3\times 3} & N_l \\ -N_l & 0_{3\times 3} \end{bmatrix} \text{ for } l = x, y, z \text{ and with } {}^tN_l = -N_l$$

- Property : for any $\mathbf{n} = {}^t(n_x, n_y, n_z)$ with $\| \mathbf{n} \| = 1$,

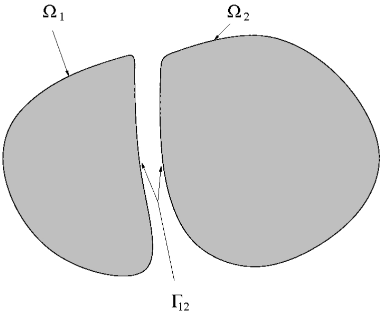
$$C(\mathbf{n}) = G_0^{-1}\left(n_x G_x + n_y G_y + n_z G_z\right) \text{ is diagonalizable}$$

$$C(\mathbf{n}) = T(\mathbf{n})\Lambda(\mathbf{n})T^{-1}(\mathbf{n})$$

$$\text{Eigenvalues : } \lambda_{1,2} = -c\,,\ \lambda_{3,4} = 0\,,\ \lambda_{5,6} = c \text{ with } c = \frac{1}{\sqrt{\varepsilon\mu}}$$

Schwarz algorithm

- $\Omega = \bigcup_{j=1}^{N_s} \Omega_j$, $\mathbf{W}^j = \mathbf{W}|_{\Omega_j}$
- $\Gamma = \Gamma_a$ (for the presentation)
- Overlapping subdomains



Schwarz algorithm

- $\Omega = \bigcup_{j=1}^{N_s} \Omega_j$, $\mathbf{W}^j = \mathbf{W}|_{\Omega_j}$
- $\Gamma = \Gamma_a$ (for the presentation)
- Overlapping subdomains

$$\left\{\begin{array}{lcl} \mathscr{L}\mathbf{W}^{j,p+1} & = & 0 \ \text{ in } \Omega_j \\ \mathscr{B}_{\mathbf{n}_{jl}}\mathbf{W}^{j,p+1} & = & \mathscr{B}_{\mathbf{n}_{jl}}\mathbf{W}^{l,p} \ \text{ on } \Gamma_{jl} = \partial\Omega_j \cap \bar{\Omega}_l \\ G_{\mathbf{n}}^{-}\mathbf{W}^{j,p+1} & = & G_{\mathbf{n}}^{-}\mathbf{W}_{\rm inc} \ \text{ on } \Omega_j \cap \Gamma_a \end{array}\right.$$

Classical (natural) interface conditions

$$\mathscr{B}_{\mathbf{n}} \equiv G_{\mathbf{n}}^{-}$$

$$G_{\mathbf{n}}^{-}\mathbf{W} \iff \mathbf{n} \times \mathbf{E} + z\mathbf{n} \times (\mathbf{n} \times \mathbf{H}) \ \text{(impedance condition)}$$

Schwarz algorithm

- $\Omega = \bigcup_{j=1}^{N_s} \Omega_j$, $\mathbf{W}^j = \mathbf{W}|_{\Omega_j}$
- $\Gamma = \Gamma_a$ (for the presentation)
- Overlapping subdomains

$$\left\{ \begin{array}{lcl} \mathscr{L}\mathbf{W}^{j,p+1} & = & 0 \text{ in } \Omega_j \\ \mathscr{B}_{\mathbf{n}_{jl}}\mathbf{W}^{j,p+1} & = & \mathscr{B}_{\mathbf{n}_{jl}}\mathbf{W}^{l,p} \text{ on } \Gamma_{jl} = \partial\Omega_j \cap \bar{\Omega}_l \\ G_{\mathbf{n}}^{-}\mathbf{W}^{j,p+1} & = & G_{\mathbf{n}}^{-}\mathbf{W}_{\text{inc}} \text{ on } \Omega_j \cap \Gamma_a \end{array} \right.$$

Classical (natural) interface conditions

$$\mathscr{B}_{\mathbf{n}} \equiv G_{\mathbf{n}}^{-}$$

$$G_{\mathbf{n}}^{-}\mathbf{W} \iff \mathbf{n} \times \mathbf{E} + z\mathbf{n} \times (\mathbf{n} \times \mathbf{H}) \text{ (impedance condition)}$$

- Convergence result
  - V. Dolean, M.J. Gander and L. Gerardo-Giorda, SISC, Vol. 31, No. 3 (2009)
  - Fourier analysis (for constant ε and μ)
  - $\Omega_1 =]-\infty, b[\times\mathbb{R}^2$ and $\Omega_2 =]a, +\infty[\times\mathbb{R}^2$ with $a \leq b$

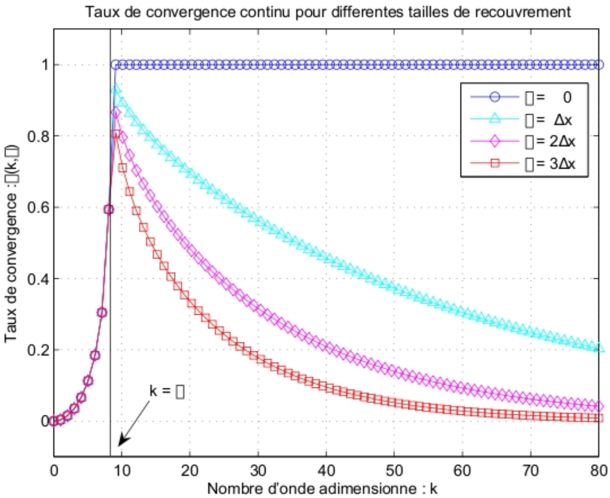
Convergence rate (non-conductive case)

$$\rho(\mathbf{k},\delta) = \left| \left(\frac{\sqrt{\mathbf{k}^2 - \tilde{\omega}^2} - \mathrm{i}\tilde{\omega}}{\sqrt{\mathbf{k}^2 - \tilde{\omega}^2} + \mathrm{i}\tilde{\omega}} \right) e^{-\delta\sqrt{\mathbf{k}^2 - \tilde{\omega}^2}} \right|$$

with $\delta = b - a$ and $\tilde{\omega} = \omega\sqrt{\varepsilon\mu}$

$$\rho(k,\delta) = \begin{cases} \left| \dfrac{\sqrt{\tilde{\omega}^2 - \mathbf{k}^2} - \tilde{\omega}}{\sqrt{\tilde{\omega}^2 - \mathbf{k}^2} + \tilde{\omega}} \right| & \text{if } |\mathbf{k}|^2 \leq \tilde{\omega}^2 \text{ (propagative modes)} \\ e^{-\delta\sqrt{\mathbf{k}^2 - \tilde{\omega}^2}} & \text{if } |\mathbf{k}|^2 > \tilde{\omega}^2 \text{ (evanescent modes)} \end{cases}$$

Convergence rate as a function of the frequency parameter



Schwarz algorithm with optimized interface conditions

- V. Dolean, M.J. Gander and L. Gerardo-Giorda, SISC, Vol. 31, No. 3 (2009)
- $\mathscr{S}_j$ for $j = 1, \cdots, N_s$: tangential operator
  Interface condition : $\left(\mathscr{B}_{\mathbf{n}_{jl}} + \mathscr{S}_j \mathscr{B}_{\mathbf{n}_{lj}}\right) \mathbf{W}^{j,p+1} = \left(\mathscr{B}_{\mathbf{n}_{jl}} + \mathscr{S}_j \mathscr{B}_{\mathbf{n}_{lj}}\right) \mathbf{W}^{l,p}$

Optimal interface operators

$$\mathscr{S}_j = \alpha_j = (\mathrm{i}\tilde{\omega})^{-1}(p_j - \mathrm{i}p_j) \;\text{ for }\; j = 1, 2$$

Case	p_1	p_2	Asymptotic ρ
1	0	0	1
2	$\dfrac{\sqrt{C}C_{\tilde{\omega}}^{\frac{1}{4}}}{\sqrt{2}\sqrt{h}}$	$\dfrac{\sqrt{C}C_{\tilde{\omega}}^{\frac{1}{4}}}{\sqrt{2}\sqrt{h}}$	$1 - \dfrac{\sqrt{2}C_{\tilde{\omega}}^{\frac{1}{4}}}{\sqrt{C}}\sqrt{h}$
3	$\dfrac{C^{\frac{1}{4}}C_{\tilde{\omega}}^{\frac{3}{8}}}{2h^{\frac{1}{4}}}$	$\dfrac{C^{\frac{3}{4}}C_{\tilde{\omega}}^{\frac{1}{8}}}{h^{\frac{3}{4}}}$	$1 - \dfrac{C_{\tilde{\omega}}^{\frac{1}{8}}}{C^{\frac{1}{4}}}h^{\frac{1}{4}}$

- Numerical results in 2D (TMz mode)
  - Scattering of a plane wave by a dielectric cylinder, F=300 MHz
  - # vertices = 2078 and # elements = 3958 - Upwind flux

Classical Schwarz method

Method	L2 error on E_z	N_s	# iter BiCGStab ($\varepsilon = 10^{-6}$)
DGFD-$\mathbb{P}_1$	0.16400	4	317
-	0.16400	16	393
DGFD-$\mathbb{P}_2$	0.05701	4	650
-	0.05701	16	734
DGFD-$\mathbb{P}_3$	0.05519	4	1067
-	0.05519	16	1143
DGFD-$\mathbb{P}_4$	0.05428	4	1619
-	0.05427	16	1753
DGFD-$\mathbb{P}_i$	0.05487	4	352
-	0.05487	16	414

- Numerical results in 2D (TMz mode)
  - Scattering of a plane wave by a dielectric cylinder, F=300 MHz
  - # vertices = 2078 and # elements = 3958 - Upwind flux
  - DGFD-$\mathbb{P}_i$ ($N_s = 4$): 25.8 sec (classical) / 3.6 sec (optimized)

Optimized Schwarz method (case 1)

Method	L2 error on E_z	N_s	# iter BiCGStab ($\varepsilon = 10^{-6}$)
DGFD-$\mathbb{P}_1$	0.16457	4	52 (6.1) [a]
-	0.16467	16	83 (4.7)
DGFD-$\mathbb{P}_2$	0.05705	4	61 (10.7)
-	0.05706	16	109 (6.7)
DGFD-$\mathbb{P}_3$	0.05519	4	71 (15.0)
-	0.05519	16	139 (8.2)
DGFD-$\mathbb{P}_4$	0.05427	4	83 (19.5)
-	0.05527	16	170 (10.3)
DGFD-$\mathbb{P}_i$	0.05486	4	49 (7.2)
-	0.05491	16	81 (5.1)

[a] # iter classical/# iter optimized

- Numerical results in 2D (TMz mode)
  - Scattering of a plane wave by a dielectric cylinder, F=300 MHz
  - # vertices = 2078 and # elements = 3958
  - $N_s = 4$ subdomains

Optimized Schwarz method (case 1)

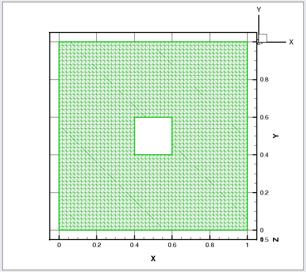
Method	Flux	L2 error on E_z	# iter BiCGStab ($\varepsilon = 10^{-6}$)	RAM LU (min/max)
DGFD-$\mathbb{P}_1$	Upwind	0.16457	52	26 MB/ 27 MB
-	Centered	0.35274	53	15 MB/ 15 MB
DGFD-$\mathbb{P}_2$	Upwind	0.05705	61	69 MB/ 71 MB
-	Centered	0.05823	61	39 MB/ 41 MB
DGFD-$\mathbb{P}_3$	Upwind	0.05519	71	140 MB/147 MB
-	Centered	0.05520	77	86 MB/ 90 MB
DGFD-$\mathbb{P}_4$	Upwind	0.05427	83	237 MB/249 MB
-	Centered	0.05527	85	156 MB/161 MB
DGFD-$\mathbb{P}_i$	Upwind	0.05486	49	54 MB/ 69 MB
-	Centered	0.05583	49	33 MB/ 42 MB

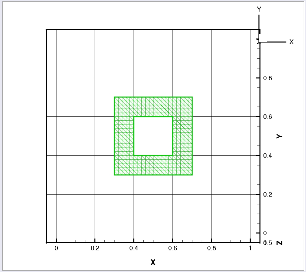
- Classical Schwarz method
- Solvers
  - Interface system
    - BiCGstab(ℓ) (G.L.G. Sleijpen and D.R. Fokkema, ETNA, Vol.1, 1993)
    - No preconditioner, $\ell = 6$
  - Local systems
    - MUMPS multifrontal sparse matrix solver
      (P.R. Amestoy, I.S. Duff and J.-Y. L'Excellent, CMAME, Vol. 184, 2000)
    - Mixed arithmetic strategy: LU in 32 bit + iterative refinement
- Hardware platform
  - Bull Novascale 3045 system of the CEA/CCRT center
    (Centre de Calcul Recherche et Technologie)
  - Intel Itanium 2/1.6 GHz, InfiniBand

- Scattering of a plane wave by a coated perfectly conducting cube
- F=900 MHz, $\Omega = [0,1]^3$

Characteristics of the tetrahedral meshes

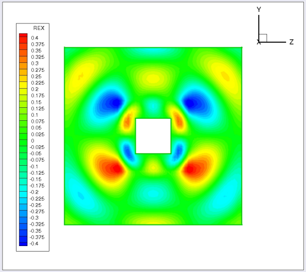
Mesh	# vertices	# tetrahedra
M1	131,922	744,000
M2	355,714	2,041,536

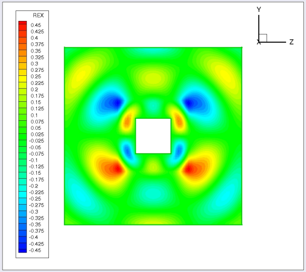




- Scattering of a plane wave by a coated perfectly conducting cube
- F=900 MHz, $\Omega = [0,1]^3$

Contour lines of E_x for $x = 0.5$, DGTH-$\mathbb{P}_1$ method



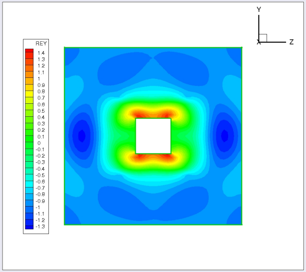


Mesh M1, # vertices = 131,922 Mesh M2, # vertices = 355,714

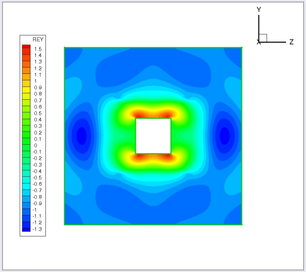
Centered flux

- Scattering of a plane wave by a coated perfectly conducting cube
- F=900 MHz, $\Omega = [0,1]^3$

Contour lines of E_y for $x = 0.5$, DGTH-$\mathbb{P}_1$ method



Mesh M1, # vertices = 131,922

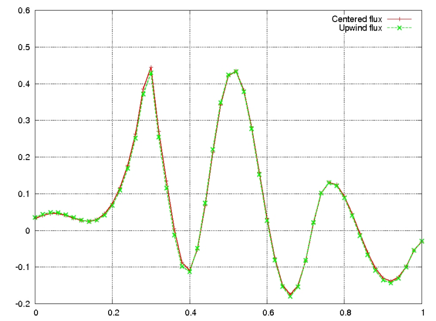


Mesh M2, # vertices = 355,714

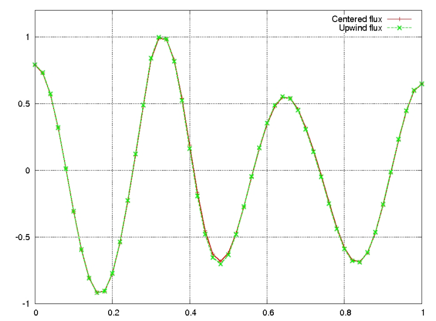
Centered flux

- Scattering of a plane wave by a coated perfectly conducting cube
- F=900 MHz, $\Omega = [0,1]^3$

x-wise distributions for $y = z = 0.3$, DGTH-$\mathbb{P}_1$ method



E_x component

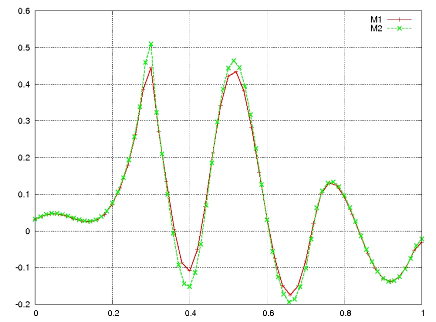


E_y component

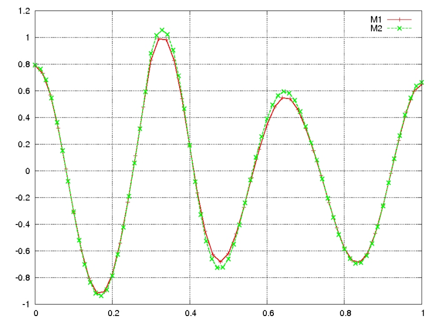
Mesh M1, # vertices = 131,922

- Scattering of a plane wave by a coated perfectly conducting cube
- F=900 MHz, $\Omega = [0,1]^3$

x-wise distributions for $y = z = 0.3$, DGTH-$\mathbb{P}_1$ method



E_x component

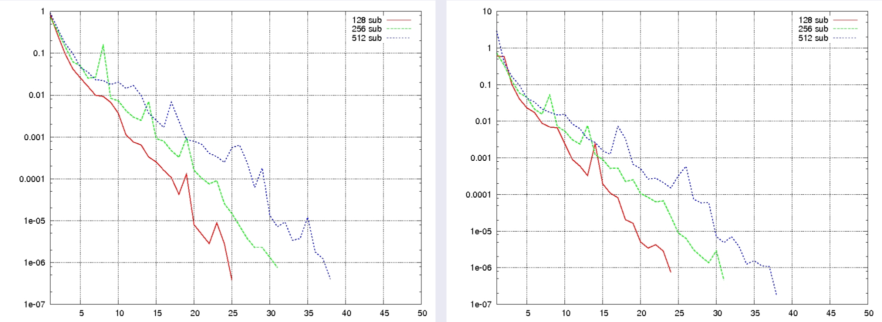


E_y component

Centered flux

- Scattering of a plane wave by a coated perfectly conducting cube
- F=900 MHz, $\Omega = [0,1]^3$

Solution of the interface system, DGTH-$\mathbb{P}_1$ method



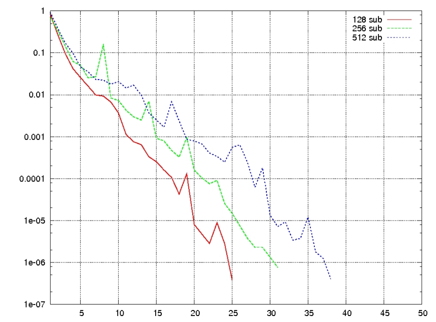
Centered flux

Upwind flux

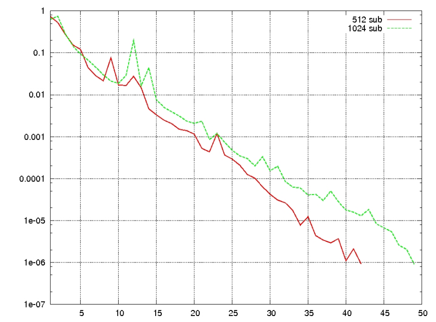
Mesh M1, # vertices = 131,922

- Scattering of a plane wave by a coated perfectly conducting cube
- F=900 MHz, $\Omega = [0,1]^3$

Solution of the interface system, DGTH-$\mathbb{P}_1$ method



Mesh M1, # vertices = 131,922



Mesh M2, # vertices = 355,714

Centered flux

- Scattering of a plane wave by a coated perfectly conducting cube
- F=900 MHz, $\Omega = [0, 1]^3$
- Mesh M1, # vertices = 131,922

Performance results, DGTH-$\mathbb{P}_1$ method

Flux	# d.o.f	N_s	# it	CPU (min/max)	Elapsed time
Centered	17,856,000	128	25	650 sec/651 sec	652 sec
-	-	256	31	401 sec/402 sec	403 sec (1.60)[a]
-	-	512	38	180 sec/183 sec	184 sec (3.55)
Upwind	17,856,000	128	24	557 sec/558 sec	559 sec
-	-	256	31	318 sec/319 sec	320 sec (1.75)
-	-	512	38	142 sec/143 sec	144 sec (3.90)

[a] Parallel speedup

- Scattering of a plane wave by a coated perfectly conducting cube
- F=900 MHz, $\Omega = [0,1]^3$
- Mesh M1, # vertices = 131,922

Performance results, DGTH-$\mathbb{P}_1$ method

Flux	N_s	RAM LU (min/max)	CPU LU (min/max)	Elapsed time LU
Centered	128	1.17 GB/1.58 GB	180 sec/181 sec	182 sec
-	256	0.42 GB/0.64 GB	47 sec/ 48 sec	49 sec (3.7)[a]
-	512	0.16 GB/0.24 GB	12 sec/ 13 sec	14 sec (13.0)
Upwind	128	1.29 GB/1.77 GB	214 sec/215 sec	216 sec
-	256	0.46 GB/0.70 GB	55 sec/ 56 sec	57 sec (3.8)
-	512	0.18 GB/0.27 GB	14 sec/ 15 sec	16 sec (13.5)

[a] Parallel speedup

- Scattering of a plane wave by a coated perfectly conducting cube
- F=900 MHz, $\Omega = [0,1]^3$
- Centered flux

Performance results, DGTH-$\mathbb{P}_1$ method

Mesh	# d.o.f	N_s	# it	CPU (min/max)	Elapsed time
M1	17,856,000	128	25	650 sec/651 sec	652 sec
M2	48,996,864	512	42	705 sec/710 sec	711 sec
-	-	1024	49	380 sec/383 sec	384 sec (1.85)[a]

Mesh	N_s	RAM LU (min/max)	CPU LU (min/max)	Elapsed time LU
M1	128	1.17 GB/1.58 GB	180 sec/181 sec	182 sec
M2	512	0.61 GB/0.97 GB	92 sec/ 93 sec	93 sec
-	1024	0.23 GB/0.38 GB	23 sec/ 25 sec	27 sec (3.4)

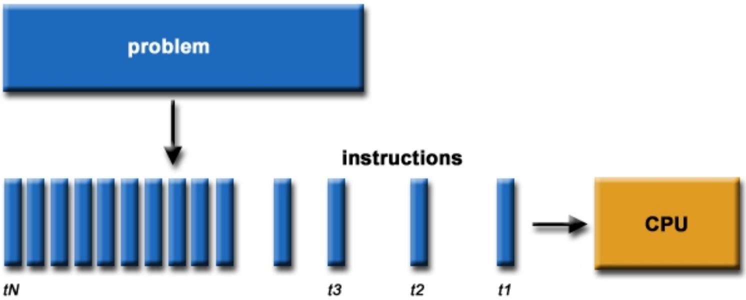
[a] Parallel speedup

Outline

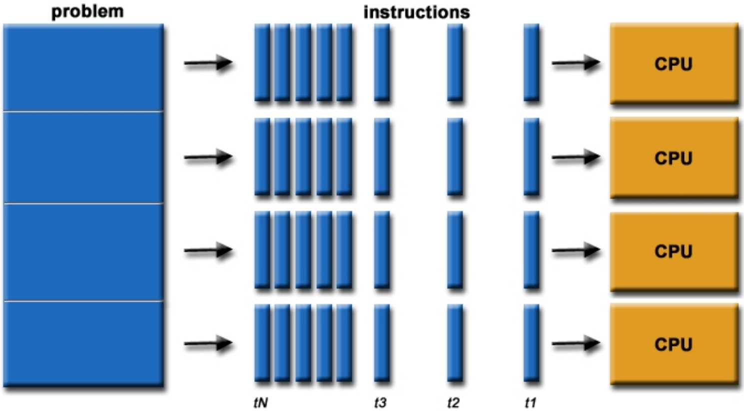
1. Time-domain electromagnetics
   - Overview of existing methods
   - Discontinuous Galerkin Time Domain (DGTD) method
   - Non-conforming DGTD-$\mathbb{P}_{p_i}$ method
   - Towards higher order time accuracy
   - Numerical treatment of grid-induced stiffness
   - 3D application: electromagnetic waves and humans
2. Frequency-domain electromagnetics
   - Numerical results in the 2D TMz case
   - Domain decomposition solver
3. High performance computing

What is parallel computing ?

- Traditionally, software has been written for serial computation:
  - a problem is run on a single computer having a single Central Processing Unit (CPU),
  - it is is broken into a discrete series of instructions,
  - instructions are executed one after another,
  - only one instruction may execute at any moment in time.

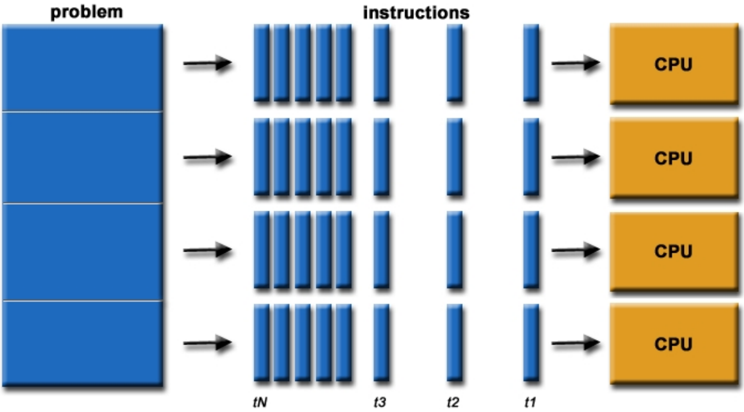


- In the simplest sense, parallel computing is the simultaneous use of multiple compute resources to solve a computational problem:
  - to be run using multiple CPUs,
  - a problem is broken into discrete parts that can be solved concurrently,
  - each part is further broken down to a series of instructions,
  - instructions from each part execute simultaneously on different CPUs.

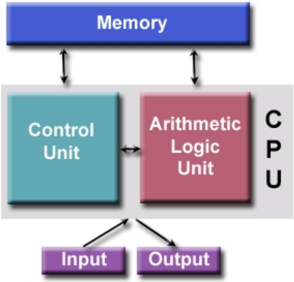


Why use parallel computing ?

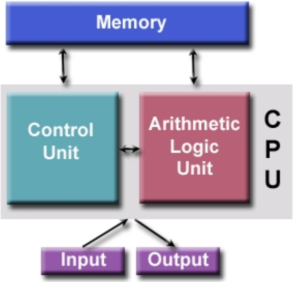
- Save time: in theory, throwing more resources at a task will shorten its time to completion
- Solve larger problems: many problems are so large and/or complex that it is impractical or impossible to solve them on a single computer, especially given limited computer memory



- Named after the Hungarian mathematician John von Neumann who first authored the general requirements for an electronic computer in his 1945 papers
- Since then, virtually all computers have followed this basic design, which differed from earlier computers programmed through hard wiring

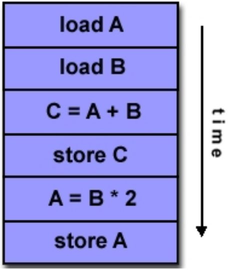


- Comprised of four main components: memory, control unit, arithmetic logic unit, input/output
- Read/write, random access memory is used to store both program instructions and data:
  - program instructions are coded data which tell the computer to do something,
  - data is simply information to be used by the program.
- Control unit fetches instructions/data from memory, decodes the instructions and then sequentially coordinates operations to accomplish the programmed task
- Aritmetic unit performs basic arithmetic operations
- Input/output is the interface to the human operator



- There are different ways to classify parallel computers
- One of the more widely used classifications, in use since 1966, is called Flynn's Taxonomy
- Flynn's taxonomy distinguishes multi-processor computer architectures according to how they can be classified along the two independent dimensions of instruction and data
- Each of these dimensions can have only one of two possible states: single or multiple
- The matrix below defines the 4 possible classifications according to Flynn:
  - SISD: Single Instruction Single Data
  - SIMD: Single Instruction Multiple Data
  - MISD: Multiple Instruction Single Data
  - MIMD: Multiple Instruction Multiple Data

- A serial (non-parallel) computer
- Single instruction: only one instruction stream is being acted on by the CPU during any one clock cycle
- Single data: only one data stream is being used as input during any one clock cycle:
  - deterministic execution,
  - this is the oldest and even today the most common type of computer.
- Examples: older generation mainframes, minicomputers and workstations most modern day PCs



CDC 7600

CRAY 1

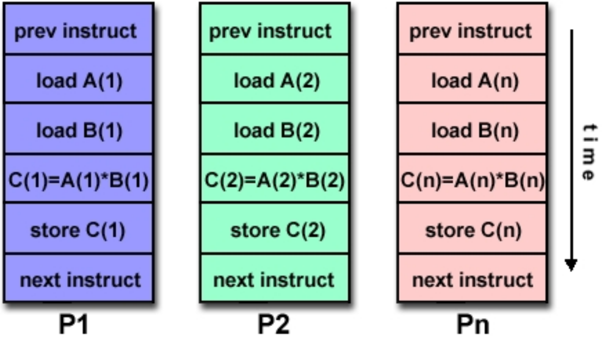
PDP 1

IBM 360

UNIVAC 1

DELL laptop

- A type of parallel computer
- Single instruction: all processing units execute the same instruction at any given clock cycle
- Multiple data: each processing unit can operate on a different data element
- Best suited for specialized problems characterized by a high degree of regularity, such as graphics/image processing
- Synchronous (lockstep) and deterministic execution
- Two varieties: processor arrays and vector pipelines
  - Processor arrays: Connection Machine CM-2, MasPar MP-1 and MP-2, ILLIAC IV
  - Vector Pipelines: IBM 9000, Cray X-MP, Y-MP and C90, Fujitsu VP, NEC SX-2, Hitachi S820, ETA10
- Most modern computers, particularly those with graphics processor units (GPUs) employ SIMD instructions and execution units

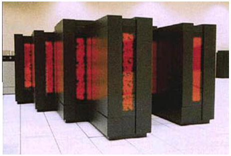

prev instruct
load A(1)
load B(1)
C(1)=A(1)*B(1)
store C(1)
next instruct
P1
prev instruct
load A(2)
load B(2)
C(2)=A(2)*B(2)
store C(2)
next instruct
P2
prev instruct
load A(n)
load B(n)
C(n)=A(n)*B(n)
store C(n)
next instruct
Pn
time

MasPar

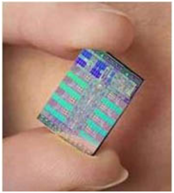
Cray X-MP

Cray Y-MP

ILLIAC IV

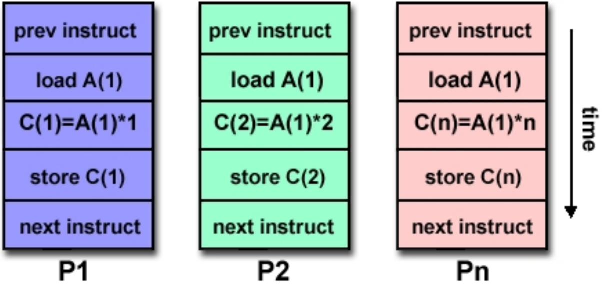


CM-2

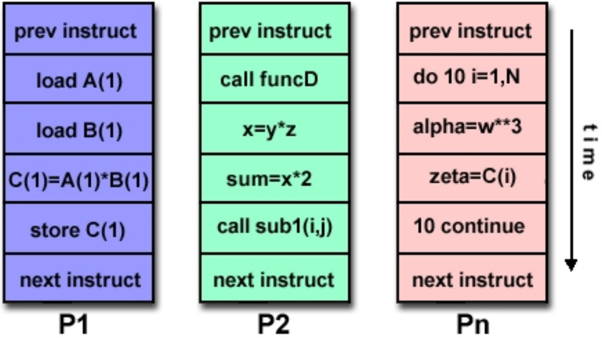


Cell Processor (GPU)

- A single data stream is fed into multiple processing units
- Each processing unit operates on the data independently via independent instruction streams
- Few actual examples of this class of parallel computer have ever existed (one is the experimental Carnegie-Mellon C.mmp computer (1971))
- Some conceivable uses might be:
  - multiple frequency filters operating on a single signal stream,
  - multiple cryptography algorithms attempting to crack a single coded message.



- Currently, the most common type of parallel computer
- Most modern computers fall into this category
- Multiple instruction: every processor may be executing a different instruction stream
- Multiple data: every processor may be working with a different data stream
- Execution can be synchronous or asynchronous, deterministic or non-deterministic
- Examples: most current supercomputers, networked parallel computer clusters and grids, multi-processor SMP computers, multi-core PCs
- Note: many MIMD architectures also include SIMD execution sub-components


prev instruct
load A(1)
load B(1)
C(1)=A(1)*B(1)
store C(1)
next instruct
P1
prev instruct
call funcD
x=y*z
sum=x*2
call sub1(i,j)
next instruct
P2
prev instruct
do 10 i=1,N
alpha=w**3
zeta=C(i)
10 continue
next instruct
Pn
time

HP/Compaq Alphaserver

Intel IA32 cluster

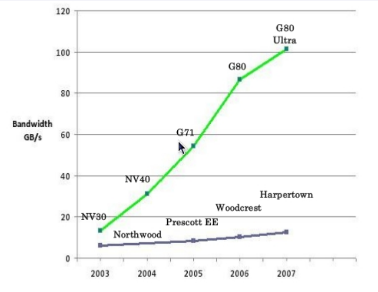
IBM POWER5

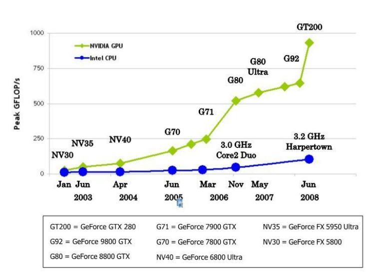
AMD Opteron cluster

Cray XT3

IBM BG/L

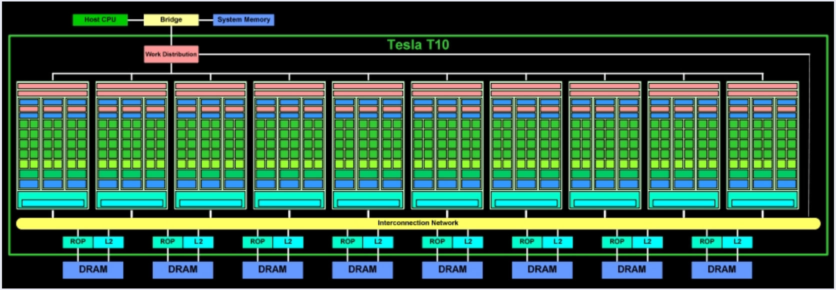
CPUs vs GPUs performance evolution





Intel Core i7 – 975 XE	106 GFLOPS	25.6 GB/s
NVIDIA GeForce GTX 280	933 GFLOPS	113 GB/s

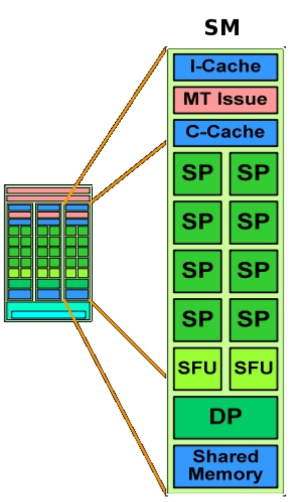
GPU200 global architecture



Tesla C1060

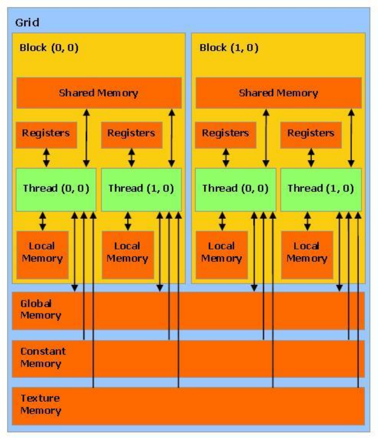
- 10 Texture Processor Clusters
- 30 Streaming Multiprocessors (SM) - SIMD processors
- 4GB of DRAM

SM Architecture



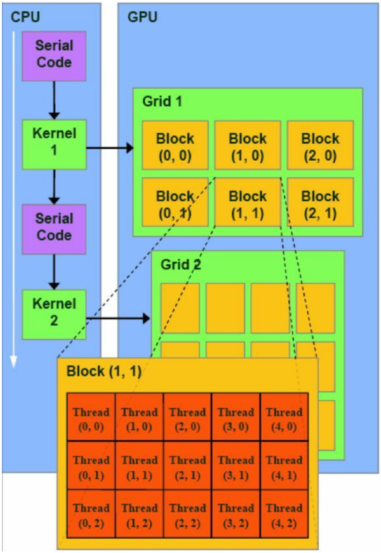
- 1 Multithreaded instruction Unit
- 8 Single Precision Processors
- 2 Special Function Units
- 1 Double Precision Unit
- 16k 32-bit registers
- 16k Shared memory

Memory architecture



- Fast
  - Registers (belongs to threads)
  - Shared memory (belongs to block)
- With cache
  - Texture memory (Read mode only)
  - Constant memory (Read mode only)
- Slow
  - Global memory 400 to 600 cycles of latency

CUDA Programming model



- Thread
  - Base element
  - All threads run the same code
  - Thread index is a built-in variable
  - Has a set of registers containing program context
- Block
  - Set of threads
  - All threads of a block runs in the same SM
  - Have common shared memory
  - Threads can be synchronize in a block
- Grid
  - set of blocks

GPU constraints

- A GPU works in a SIMD way
- This implies that a certain number of consecutives threads have to execute the same code at the same time to get the most of GPU's processing capabilities
- In the CUDA nodel, this number is 32 and such a group of threads is called a warp
- There are also rules to read and write in global GPU's memory allowing coalesced access and impressive bandwidth
- At last, the programmer has to remember that GPU programming is not high-level; in particular, he as to think about the number of registers available for a given kernel

Data distribution between blocks

- We have adopted an approach in which each block deals with a certain number of elements (tetrahedron)
- This number is defined according to the order of the computations
- For example, it is equal to 32 for the DGTD-$\mathbb{P}_1$ method and 16 for the DGTD-$\mathbb{P}_2$ and DGTD-$\mathbb{P}_3$ methods
- Indeed, the number of DOF (degrees of freedom) changes with order but not the available hardware resources i.e. the number of registers in a SM and the shared memory size
- This strategy works fine for order below 4; for higher interpolation orders, register pressures begin to be too important for actual hardware and a new approach will have to be defined

Implementation

The DGTD method is an iterative algorithm that computes at each time step the evolution of the electric and magnetic fields

Each iteration can be decomposed into 4 steps applied at the tetrahedron level

intVolume : computes the volume integral,

$$\frac{1}{2}\iiint_{\tau_i}(\nabla\times\vec{\varphi}\cdot\mathbf{H}_i+\nabla\times\mathbf{H}_i\cdot\vec{\varphi})d\omega$$

intSurface : computes the surface integral for internal faces $a_{ik}=\tau_i\cap\tau_j$,

$$\frac{1}{2}\sum_{k\in\mathscr{V}_i}\iint_{a_{ik}}\vec{\varphi}\cdot(\mathbf{H}_k\times\vec{n}_{ik})ds$$

IntSurfaceBdry : computes the surface integral (same as above) for boundary faces

UpdateEM : updates the electromagnetic field

Parallelization strategy for clusters of CPUs

Domain partitioning + message passing programming (MPI)

Computing platform

HPC resource mad available by GENCI (Grand Equipement National de Calcul Intensif)
Allocation 2010-t2010065004

Hybrid CPU-GPU cluster of the CCRT (Centre de Calcul Recherche et Technologie)
in Bruyères-le-Châtel, France

1068 Intel CPU nodes with two quad-core Intel Xeon X5570 Nehalem processors
operating at 2.93 GHz each

48 Teslas S1070 GPU systems with four GT200 GPUs and two PCI Express-2 buses each

The network is a non-blocking, symmetric, full duplex Voltaire InfiniBand double data rate
organized as a fat tree

The original DGTD software is developed in Fortran 90

Simulations are performed in single precision arithmetic

Model test problem and configurations

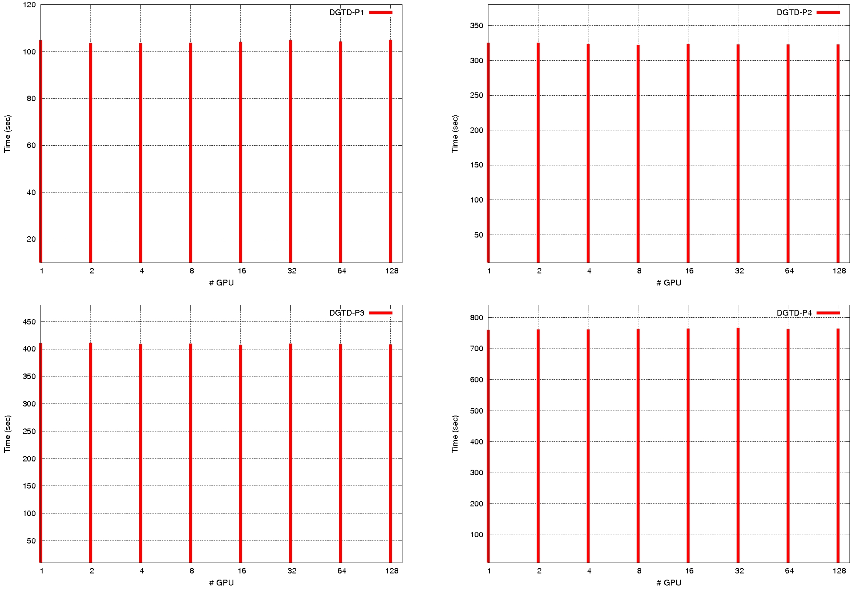
Propagation of a standing wave in a perfectly conducting unitary cubic cavity

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

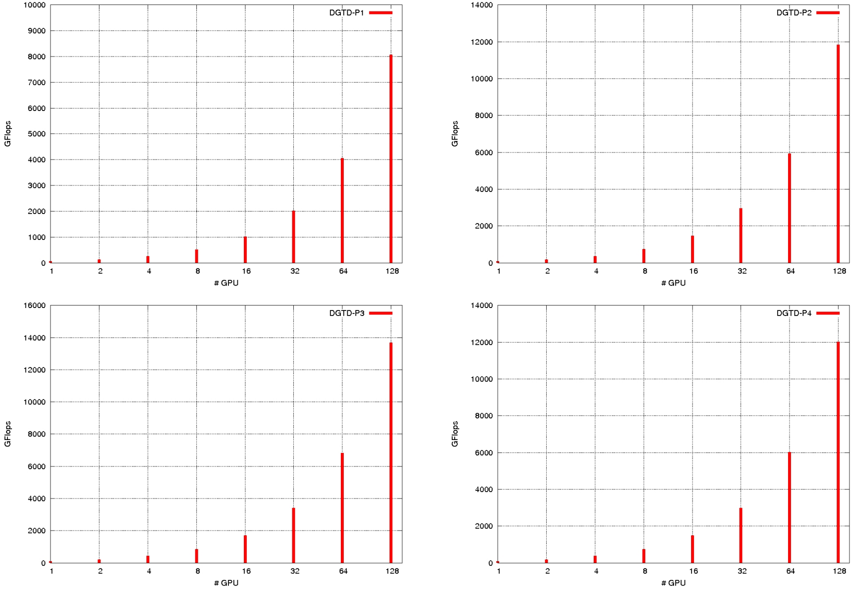
Boxwise domain decompositions with optimal computational load balance

Timings for 1000 iterations and up to 128 GPUs

Weak scalability: timings



Weak scalability: GFlops rates



Weak scalability

Model test problem and congiurations

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

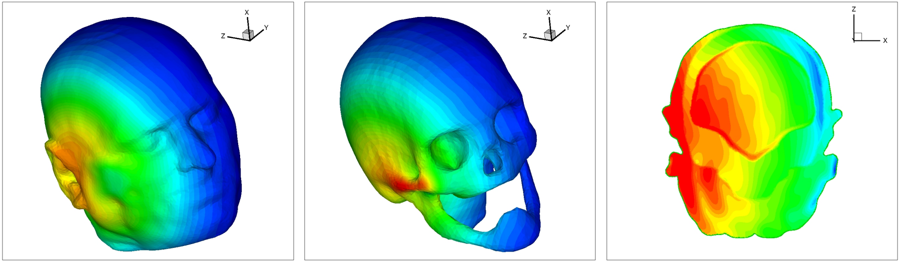
Regular uniform tetrahedral meshes respectively containing 3,072,000 elements for the DGTD-$\mathbb{P}_1$ and DGTD-$\mathbb{P}_2$ methods, 1,296,000 elements for the DGTD-$\mathbb{P}_3$ method and 750,000 elements for the DGTD-$\mathbb{P}_4$ method

Boxwise domain decomposition with optimal computational load balance

Timings for 1000 iterations and up to 128 GPUs

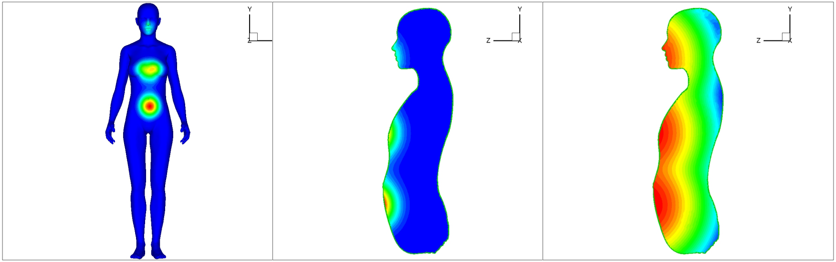
Computational performances

# GPU	DGTD-$\mathbb{P}_1$	DGTD-$\mathbb{P}_2$	DGTD-$\mathbb{P}_3$	DGTD-$\mathbb{P}_4$
1	63 GFlops	92 GFlops	106 GFlops	94 GFlops
128	8072 GFlops	11844 GFlops	13676 GFlops	12009 GFlops



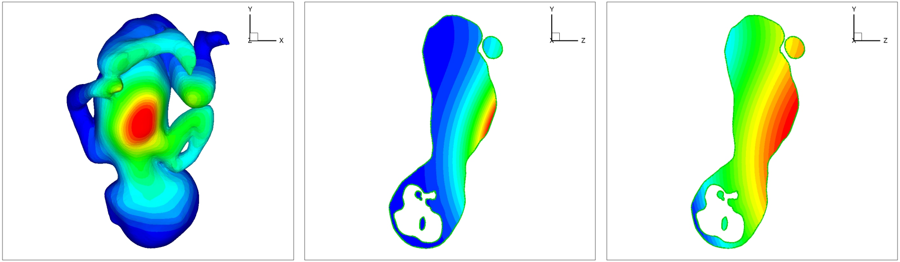
- Mesh: # elements = 7,894,172
- Total # dof is 189,45,8688 (DGTD-$\mathbb{P}_1$ method) and 473,646,720 (DGTD-$\mathbb{P}_2$ method)
- Time on 128 CPU cores: 2786 sec (DGTD-$\mathbb{P}_1$ method) and 6057 sec (DGTD-$\mathbb{P}_2$ method)

# GPU	DGTD-$\mathbb{P}_1$			DGTD-$\mathbb{P}_2$		
	Time	GFlops	Speedup	Time	GFlops	Speedup
32	162 sec	146	-	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



- Mesh: # elements = 5,536,852
- Total # dof is 132,884,448 (DGTD-$\mathbb{P}_1$ method) and 332,211,120 (DGTD-$\mathbb{P}_2$ method)
- Time on 64 CPU cores for the DGTD-$\mathbb{P}_1$ method: 7 h 10 mn

# GPU	DGTD-$\mathbb{P}_1$			DGTD-$\mathbb{P}_2$		
	Time	GFlops	Speedup	Time	GFlops	Speedup
64	12 mn	2762	-	59 mn	4525	-
128	7 mn	4643	1.7	30 mn	8865	1.95



- Mesh: # elements = 5,536,852
- Total # dof is 132,884,448 (DGTD-$\mathbb{P}_1$ method) and 332,211,120 (DGTD-$\mathbb{P}_2$ method)
- Time on 64 CPU cores for the DGTD-$\mathbb{P}_1$ method: 7 h 10 mn

# GPU	DGTD-$\mathbb{P}_1$			DGTD-$\mathbb{P}_2$		
	Time	GFlops	Speedup	Time	GFlops	Speedup
64	12 mn	2762	-	59 mn	4525	-
128	7 mn	4643	1.7	30 mn	8865	1.95