Integral Equations, Fast Algorithms, and Parallelization Strategies for the Solution of Extremely Large Problems in Computational Electromagnetics

Lecture Notes

May 17, 2015

ExpoMeloneras Convention Centre
Gran Canaria, Spain
Integral Equations, Fast Algorithms, and Parallelization Strategies for the Solution of Extremely Large Problems in Computational Electromagnetics

Lecture Notes

May 17, 2015

ExpoMeloneras Convention Centre
Gran Canaria, Spain

*This School is organized during the “2015 URSI Atlantic Radio Science Conference” (URSI AT-RASC 2015), May 16-24, 2015, Gran Canaria, Spain.
Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preface</td>
<td>1</td>
</tr>
<tr>
<td>Program</td>
<td>3</td>
</tr>
<tr>
<td>Lecture Abstract</td>
<td>5</td>
</tr>
<tr>
<td>Biographical Sketch of Course Instructor</td>
<td>6</td>
</tr>
<tr>
<td>Integral Equations, Fast Algorithms, and Parallelization Strategies for the Solution of Extremely Large Problems in Computational Electromagnetics By Prof. Levent Gurel, CEO, ABAKUS Computing Technologies, Turkey; Adjunct Professor, ECE, University of Illinois at Urbana-Champaign, USA</td>
<td>7</td>
</tr>
<tr>
<td>Part 1</td>
<td>9</td>
</tr>
<tr>
<td>Part 2</td>
<td>47</td>
</tr>
</tbody>
</table>
Preface

The “2015 URSI Commission B School for Young Scientists” is organized by URSI Commission B and is arranged on the occasion of the “2015 URSI Atlantic Radio Science Conference” (URSI AT-RASC 2015), May 16-24, 2015, Gran Canaria, Spain. This School is a one-day event held during URSI AT-RASC 2015, and is sponsored jointly by URSI Commission B and the URSI AT-RASC 2015 Organizing Committee. The School offers a short, intensive course, where a series of lectures will be delivered by a leading scientist in the Commission B community. Young scientists are encouraged to learn the fundamentals and future directions in the area of electromagnetic theory from these lectures.
Program

1. Course Title

Integral Equations, Fast Algorithms, and Parallelization Strategies for the Solution of Extremely Large Problems in Computational Electromagnetics

2. Course Instructor

Prof. Levent Gurel
CEO, ABAKUS Computing Technologies, Turkey;
Adjunct Professor, ECE, University of Illinois at Urbana-Champaign, USA

3. Course Program

Lecture 1
- Date and Time: 9:00-13:00, Sunday, May 17, 2015
- Venue: ExpoMeloneras Convention Centre, Gran Canaria, Spain
- Lecture Topics:
  - Introduction
  - Computational electromagnetics
  - Maxwell’s equations
  - Integral equations
  - Method of moments
  - Fast multipole method (FMM)
  - Clustering, aggregation, translation, and disaggregation
  - Complexity of FMM
  - Multilevel fast multipole algorithm (MLFMA)
  - Recursive clustering and tree structure
  - Interpolation and anterpolation
  - Multilevel aggregation, translation, and disaggregation
  - Complexity of MLFMA

Lecture 2
- Date and Time: 14:00-18:00, Sunday, May 17, 2015
- Venue: ExpoMeloneras Convention Centre, Gran Canaria, Spain
- Lecture Topics:
  - Parallelization of MLFMA
  - Load balancing
  - Simple parallelization
  - Hybrid parallelization
  - Hierarchical parallelization
  - Iterative methods
  - Preconditioners
  - Solution of large problems
  - Application examples
  - Conclusions
Lecture Abstract

Integral Equations, Fast Algorithms, and Parallelization Strategies for the Solution of Extremely Large Problems in Computational Electromagnetics

Prof. Levent GUREL, PhD, FIEEE, FEMA, FACES
CEO, ABAKUS Computing Technologies, Turkey
http://abakus.computing.technology
Adjunct Professor, ECE, University of Illinois at Urbana-Champaign, USA
www.ece.illinois.edu/directory/profile.asp?lgurel
Professor Emeritus, Bilkent University, Turkey
Founder, Computational Electromagnetics Research Center (BiLCEM)
Email: lgurel@gmail.com

2015 edition of the URSI Commission B School for Young Scientists lectures by Prof. Levent Gurel focuses on the solution of extremely large problems in electromagnetics. Fast solvers, such as the fast multipole method (FMM) and the multilevel fast multipole algorithm (MLFMA), will be considered. These methods can be applied to scattering, radiation, propagation, resonance, guidance, and transmission problems in electromagnetics. Furthermore, they can also be applied to the solution of problems from other disciplines, such as quantum mechanics, astrophysics, molecular dynamics, etc. Wave phenomena in electrodynamics, acoustics, elastics and seismic problems can be studied with these methods. As such, applications can be derived from a very wide portfolio, including, but not limited to optics, nanotechnology, metamaterials, antennas, radars, remote sensing, imaging, biomedical, bioelectromagnetics, stealth technology and radar-cross-section (RCS) computations.

Following a general introduction to computational electromagnetics, this course focuses on the fast and accurate solutions of large-scale electromagnetic modeling problems involving three-dimensional geometries with arbitrary shapes using FMM, MLFMA, and parallel MLFMA. Accurate simulations of real-life electromagnetics problems with integral equations require the solution of dense matrix equations involving millions of unknowns. Solutions of those extremely large problems cannot be achieved easily, even when using the most powerful computers with state-of-the-art technology. Nevertheless, some of the world’s largest integral-equation problems in computational electromagnetics can be solved by employing fast algorithms implemented on parallel computers. Most recently, we have achieved the solution of 1,000,000,000x1,000,000,000 (one billion!) dense matrix equations! This achievement requires a multidisciplinary study involving physical understanding of electromagnetics problems, novel parallelization strategies, constructing parallel clusters, advanced mathematical methods for integral equations, fast solvers, iterative methods, preconditioners, and linear algebra. In this course, various examples of CEM problems derived from real-life applications are considered.
Levent Gürel received the B.Sc. degree from the Middle East Technical University (METU) in 1986, and the M.S. and Ph.D. degrees from the University of Illinois at Urbana-Champaign (UIUC) in 1988 and 1991, respectively, in electrical and computer engineering. After spending 3 years at the Thomas. J. Watson Research Center of IBM in Yorktown Heights, New York, where he worked on the solution of electromagnetics problems relevant to the computer industry, he moved to Bilkent University, Ankara, Turkey. During his 20 years with Bilkent University (1994-2014), he served as the Founding Director of the Computational Electromagnetics Research Center (BiLCEM) and a professor of electrical engineering. Currently, Prof. Gürel is the Founder and CEO of the ABAKUS Computing Technologies company that is geared towards providing advanced solutions and creating cutting-edge technologies through R&D projects in computational sciences. He is also an adjunct professor at the Electrical and Computer Engineering Department of UIUC, a consultant to industry, and a member of the Board of Trustees of Izmir University of Economics.

Prof. Gürel is named an IEEE Distinguished Lecturer for 2011-2014. In 2013, his contributions to science have been recognized by the Electrical and Computer Engineering Department of the University of Illinois with the Distinguished Alumni Award.

Prof. Gürel was named an IEEE Fellow “on the basis of his contributions to fast methods and algorithms for computational electromagnetics” in 2009. Also, he was elevated to the Fellow grade by the Electromagnetics Academy in 2007 and elected to become a Fellow of the Applied Computational Electromagnetic Society (ACES) in 2011. He received two prestigious awards from the Turkish Academy of Sciences (TUBA) in 2002 and the Scientific and Technical Research Council of Turkey (TUBITAK) in 2003. He served as a member of the ACES Board of Directors during 2011-2014.

He has been organizing and serving as the General Chairman and Editor of the biennial Computational Electromagnetics International Workshops held in 2007-2015. Since 2003, Prof. Gürel has been serving as an associate editor for Radio Science, IEEE Antennas and Wireless Propagation Letters, IET Microwaves, Antennas & Propagation, JEMWA, PIER, ACES Journal, and ACES Express.

Prof. Gürel was invited to address the 2011 ACES Conference as a Plenary Speaker and a TEDx conference in 2014. He served as the Chairman of the AP/MTT/ED/EMC Chapter of the IEEE Turkey Section in 2000-2003. He founded the IEEE EMC Chapter in Turkey in 2000. He served as the Cochairman of the 2003 IEEE International Symposium on Electromagnetic Compatibility.
Integral Equations, Fast Algorithms, and Parallelization Strategies for the Solution of Extremely Large Problems in Computational Electromagnetics

May 17, 2015

Prof. Levent Gurel
CEO, ABAKUS Computing Technologies, Turkey; Adjunct Professor, ECE, University of Illinois at Urbana-Champaign, USA
Integral Equations, Fast Algorithms, and Parallelization Strategies for the Solution of Extremely Large Problems in Computational Electromagnetics

Prof. Levent Gürel
CEO, ABAKUS Computing Technologies
Adjunct Professor, ECE, Univ. of Illinois at Urbana-Champaign

May 2015

ABAKÜS

Radiation into Living Organisms
Integral Equations, Fast Algorithms, and Parallelization Strategies for the Solution of Extremely Large Problems in Computational Electromagnetics - Part 1

Microwave Imaging

Antennas

Double-Ridged Horn  Conical-Corrugated Horn  Patch+SRRs  Archimedian

Fractal  Spiral  Vivaldi  Patch

Log-Periodic  Conical Spiral

Prof. Levent Gürel
http://abakus.computing.technology/
Antennas Mounted on Platforms

- Interaction of multiple antennas
- Characteristics of mounted antennas (different from isolated antennas)
- Optimization of the placement of the antennas

Maxwell’s Equations

\[ \nabla \times \vec{E}(\vec{r}, t) = - \frac{\partial}{\partial t} \vec{B}(\vec{r}, t) \]

\[ \nabla \times \vec{H}(\vec{r}, t) = \frac{\partial}{\partial t} \vec{D}(\vec{r}, t) + \vec{J}(\vec{r}, t) \]

\[ \nabla \cdot \vec{B}(\vec{r}, t) = 0 \]

\[ \nabla \cdot \vec{D}(\vec{r}, t) = \rho(\vec{r}, t) \]
Surface Integral Equations

- **Electric-Field Integral Equation (EFIE):**
  \[ -\mathbf{\hat{t}}(r) \cdot ik \int_{S'} dr' \left( \mathbf{I} - \frac{\nabla \nabla'}{k^2} \right) g(r, r') \cdot \mathbf{J}(r') = \frac{1}{\eta} \mathbf{\hat{t}}(r) \cdot \mathbf{E}^{inc}(r) \]

- **Magnetic-Field Integral Equation (MFIE):**
  \[ \mathbf{J}(r) - \mathbf{\hat{n}}(r) \times \int_{S'} dr' \mathbf{J}(r') \times \nabla' g(r, r') = \mathbf{\hat{n}}(r) \times \mathbf{H}^{inc}(r) \]

- **Combined-Field Integral Equation (CFIE):**
  \[ \text{CFIE} = \alpha \text{EFIE} + (1 - \alpha) \text{MFIE} \]

- **Hybrid-Field Integral Equation (HFIE):**
  \[ \text{HFIE} = \alpha(r) \text{EFIE} + \left[ 1 - \alpha(r) \right] \text{MFIE} \]
Geometry Discretization

Mesh Size: $\lambda/10$

Current Discretization (Expansion)

$$J(r) = \sum_{n=1}^{N} a_n b_n(r)$$

Number of unknowns

Basis functions
Discretization of EFIE

- **Electric-Field Integral Equation (EFIE):**

\[ -\hat{e}(r) \cdot i k \int_{S'} dr' \left( \hat{I} - \frac{\nabla \nabla'}{k^2} \right) g(r, r') \cdot J(r') = \frac{1}{\eta} \hat{e}(r) \cdot E^{inc}(r) \]

- Matrix equation:

\[ \sum_{n=1}^{N} Z_{mn}^E a_n = v_m^E, \quad m = 1, 2, \ldots N \]

- Matrix elements:

\[ Z_{mn}^E = \int_{S_m} dr \int_{S_n} dr' t_m(r) \bar{G}(r, r') \cdot b_n(r') \]

Discretization of MFIE

- **Magnetic-Field Integral Equation (MFIE):**

\[ J(r) - \hat{n}(r) \times \int_{S'} dr' J(r') \times \nabla' g(r, r') = \hat{n}(r) \times H^{inc}(r) \]

- Matrix equation:

\[ \sum_{n=1}^{N} Z_{mn}^M a_n = v_m^M, \quad m = 1, 2, \ldots N \]

- Matrix elements:

\[ Z_{mn}^M = \int_{S_m} dr \int_{S_n} dr' t_m(r) b_n(r') - \int_{S_m} dr \int_{S_n} dr' \hat{n} \times \int_{S_n} dr' b_n(r') \times \nabla' g(r, r') \]
CFIE
Combined-Field Integral Equation

- Matrix elements:

\[ Z_{mn}^C = \alpha Z_{mn}^E + (1 - \alpha) \frac{i}{k} Z_{mn}^M \]

- Matrix equation:

\[ \sum_{n=1}^{N} Z_{mn}^C a_n = v_m^C, \quad m = 1, 2, \ldots N \]

Matrix Elements...
...are electromagnetic interactions

\[ Z_{mn}^E = \int_{S_m} dr \ t_m(r) \cdot \int_{S_n} d\mathbf{r}' \ \tilde{G}(\mathbf{r}, \mathbf{r}') \cdot b_n(\mathbf{r}') \]

\[ \sum_{n=1}^{N} Z_{mn}^E a_n = v_m^E, \quad m = 1, 2, \ldots N \]
MOM Interactions

- **MOM**: Perform interactions one by one (for each basis and testing functions):

\[
\sum_{n=1}^{N} Z_{mn}^{M} a_n = - \int_{S_m} dr \mathbf{t}_m(r) \cdot \hat{n} \times \int_{S_n} dr' \mathbf{b}_n(r') \times \nabla' g(r, r') a_n
\]

- Basis domain
- Testing domain

Matrix Equation

System of Linear Equations

\[
A \cdot x = b
\]

\[
Z \cdot \alpha = \nu
\]
Iterative Solutions

- Large matrix equations

\[ \overline{Z} \cdot \alpha = v \]

Matrix-Vector Multiplications

\[ \overline{Z} \cdot x = y \]

Iterative Algorithm

\[ \alpha \]

Preconditioner

\[ \overline{M} \cdot z = w \]

Parallel Multilevel Fast Multipole Algorithm (MLFMA)

\[ \overline{Z} \cdot x = y \]

- CG and CGS
- BiCG and BiCGStab
- QMR and TFQMR
- GMRES
- LSQR

- BDP
- NFP
- Filtered NFP
- ILU and ILUT
- SAI
- Nested PCs

\[ \overline{M} \cdot z = w \]
Fast Multipole Method (FMM)

- **FMM**: Perform matrix-vector multiplications (for the iterative method) by using the factorization of the Green’s function.

- Calculate the interactions in group-by-group manner:
  - Basis domain
  - Testing domain

Translation
Fast Multipole Method

Cluster of basis functions

Cluster of testing functions

Aggregation

Translation

Disaggregation

Complexity: $O(N^{3/2})$

Fast Multipole Method

Factorize the Green’s Function:

$$g(r, r') = \frac{e^{ik|r-r'|}}{4\pi |r - r'|} = \frac{e^{ik|D+d|}}{|D + d|}$$

Using the identity:

$$4\pi i^l j_l(kd) P_l(\hat{d} \cdot \hat{D}) = \int d^2 \hat{k} e^{ik\cdot d} P_l(\hat{k} \cdot \hat{D})$$

Rewrite it:

$$\frac{e^{ik|D+d|}}{4\pi |D + d|} \approx \frac{1}{4\pi} \int d^2 \hat{k} e^{ik\cdot d} T_L(k, D, \theta)$$

$$T_L(k, D, \theta) = \frac{ik}{4\pi} \sum_{l=0}^{L} i^l (2l + 1) h_l^{(1)}(kD) P_l(cos\theta)$$
FMM: Evaluating the Interactions

\[
G(r, r') = \frac{1}{4\pi} \int d^2 \hat{k} \left( \mathbf{I} - \hat{k} \hat{k} \right) e^{ik \cdot (r_{rfm} + r_{fmc} + r_{efn} + r_{fnr'})} T_L(k, |r_{cc'}|, \hat{r}_{cc'} \cdot \hat{k})
\]

EFIE

\[
Z^E_{mn} = \frac{1}{4\pi} \int d^2 \hat{k} F^E_{fmc}(\hat{k}) T_L(k, |r_{cc'}|, \hat{r}_{cc'} \cdot \hat{k}) \cdot F^E_{fnc'}(\hat{k})
\]

MFIE

\[
Z^M_{mn} = \frac{k}{4\pi i} \int d^2 \hat{k} F^M_{fmc}(\hat{k}) T_L(k, |r_{cc'}|, \hat{r}_{cc'} \cdot \hat{k}) \cdot F^M_{fnc'}(\hat{k})
\]

EFIE

\[
F^E_{fmc}(\hat{k}) = e^{ik \cdot (r_{fm} - r_e)} \int_{S_m} dr e^{ik \cdot (r - r_{fm})} (\mathbf{I} - \hat{k} \hat{k}) \cdot t_m(r)
\]

\[
F^E_{fnc'}(\hat{k}) = e^{ik \cdot (r_{c'} - r_{fn})} \int_{S_n} dr' e^{-ik \cdot (r' - r_{fn})} (\mathbf{I} - \hat{k} \hat{k}) \cdot b_n(r')
\]

MFIE

\[
F^M_{fmc}(\hat{k}) = -\hat{k} \times e^{ik \cdot (r_{fm} - r_e)} \int_{S_m} dr e^{ik \cdot (r - r_{fm})} t_m(r) \times \hat{n}
\]

\[
F^M_{fnc'}(\hat{k}) = e^{ik \cdot (r_{c'} - r_{fn})} \int_{S_n} dr' e^{-ik \cdot (r' - r_{fn})} b_n(r')
\]
Fast Multipole Method

Evaluate the interactions in group-by-group manner:

\[
Z_{mn}^M = \int_{S_m} dr t_m(r) \cdot b_n(r) - \int_{S_m} dr t_m(r) \cdot \hat{n} \times \int_{S_n} dr' b_n(r') \times \nabla' g(r, r')
\]

\[
Z_{mn}^M = \frac{k}{4\pi} \int d^2 \hat{k} F_{fnc}^M(\hat{k}) T_L(k, |r_{cc'}|, \hat{n} \cdot \hat{k}) \cdot F_{fnc'}^M(\hat{k})
\]

\[
F_{fnc}^M(\hat{k}) = -\hat{k} \times e^{ik \cdot (r_f - r_e)} \int_{S_m} dr e^{ik \cdot (r - r_f)} t_m(r) \times \hat{n}
\]

\[
F_{fnc'}^M(\hat{k}) = e^{ik \cdot (r_{c'} - r_f)} \int_{S_n} dr' e^{-ik \cdot (r' - r_f)} b_n(r')
\]
Fast Multipole Method (FMM)

- Three steps of FMM:
  1. Aggregation:
     
     \[ F^{C'}(\hat{k}) = \int_{S_n} d\mathbf{r}' \exp(-i\mathbf{k} \cdot (\mathbf{r}' - \mathbf{r}_{C'})) (\mathbf{I} - \hat{\mathbf{k}}\hat{\mathbf{k}}) \cdot \mathbf{b}_n(\mathbf{r}') \]

     \[ F^{C'}(\hat{k}) = \sum_{n \in C'} F_{fnc'}(\hat{k}) \alpha_n \]

  2. Translation:
     
     \[ F_T^{C'}(\hat{k}) = \sum_{C' \notin N(C)} T_L(k, |\mathbf{r}_{cc'}|, \hat{\mathbf{r}}_{cc'} \cdot \hat{k}) F^{C'}(\hat{k}) \]

     \[ T_L(k, |\mathbf{r}_{cc'}|, \hat{\mathbf{r}}_{cc'} \cdot \hat{k}) = \frac{ik}{4\pi} \sum_{l=0}^{L} i^{l} (2l + 1) h_{l}^{(1)}(k | \mathbf{r}_{cc'} |) P_l(\hat{\mathbf{r}}_{cc'} \cdot \hat{k}) \]

     Error Source: Truncation of an infinite summation
Fast Multipole Method (FMM)

(3) Disaggregation and reception:

\[ \sum_{n=1}^{N} Z_{mn} a_n = \frac{1}{4\pi} \int d^2\hat{k} \cdot \mathbf{F}_{fmc}(\hat{k}) \cdot \mathbf{F}_T^C(\hat{k}) \]

Receiving pattern of the testing function

\[ \mathbf{F}_{fmc}(\hat{k}) = -\hat{k} \times \int d\mathbf{r} \exp(i\hat{k} \cdot (\mathbf{r} - \mathbf{r}_c)) t_m(\mathbf{r}) \times \hat{n} \]

Error Source: Angular integration over unit sphere
Fast Multipole Method

**Cluster of basis functions**

**Translation**

**Cluster of testing functions**

**Aggregation**

**Disaggregation**

\[
F_G'(\hat{k}) = \sum_{n \in G'} F_{nG'}(\hat{k}) a_n
\]

\[
F_G^T(\hat{k}) = \sum_{G' \notin N(G)} T_L(k, D) F_{G'}'(\hat{k})
\]

\[
\sum_{n=1}^{N} Z_{mn} a_n = \frac{1}{4\pi} \int d^2\hat{k} F_{mG}(\hat{k}) \cdot F_G^T(\hat{k})
\]

**Integral Equations, Fast Algorithms, and Parallelization Strategies for the Solution of Extremely Large Problems in Computational Electromagnetics - Part 1**

1) FMM performs matrix-vector multiplications (required by the iterative solver) with \(O(N^{3/2})\) FLOPs.

2) Only the near-field interactions are stored in the memory so that the memory requirement is also reduced to \(O(N^{3/2})\).

3) Hence, we are able to solve larger problems with the FMM.
Integral Equations, Fast Algorithms, and Parallelization Strategies for the Solution of Extremely Large Problems in Computational Electromagnetics - Part 1

Near-Field Matrix

Sparsity %24

930 unknowns

Preconditioners

Filtered:
- Stronger elements in the impedance matrix are selected
- Adjustable size
- Difficult to factorize and use

Diagonal:
- Diagonal (self-unknown) elements in the impedance matrix are selected
- Size is fixed
- Easy to factorize and use

Block Diagonal:
- Block-diagonal (self-cluster) elements in the impedance matrix are selected
- Size is fixed
- Easy to factorize and use

Prof. Levent Gürel
http://abakus.computing.technology/
Preconditioners

Sparse Near-Field Matrix:

- LU
- ILU: Incomplete LU
- SAI: Sparse Approximate Inverse
- INF: Iterative Near Field Preconditioner
- Use more than the available near-field matrix?

Multilevel Fast Multipole Algorithm (MLFMA)

Prof. Levent Gürel
CEO, ABAKUS Computing Technologies
Adjunct Professor, ECE, Univ. of Illinois at Urbana-Champaign

May 2015
Method of Moments

λ/10 triangulation

Large numbers of unknowns

\[ \mathbf{J}(\mathbf{r}) = \sum_{n=1}^{N} a_n \mathbf{b}_n(\mathbf{r}) \]

\[ \sum_{n=1}^{N} Z^E_{mn} a_n = v^E_m, \quad m = 1, 2, \ldots, N \]

Example (PEC-EFIE):

\[ Z^E_{mn} = \int_{S_m} d\mathbf{r} \mathbf{t}_m(\mathbf{r}) \cdot \int_{S_n} d\mathbf{r}' \mathbf{G}(\mathbf{r}, \mathbf{r}') \cdot \mathbf{b}_n(\mathbf{r}') \]

Solutions of Matrix-Equations

- Solve
  \[ \sum_{n=1}^{N} Z^E_{mn} a_n = v^E_m, \quad m = 1, 2, \ldots, N \]

- Iterative Solutions:
  - Matrix-Vector Multiplication
    \[ \mathbf{Z} \cdot \mathbf{x} = \mathbf{y} \]
  - Iterative Algorithm
    - CG and CGS
    - BiCG and BiCGStab
    - QMR and TFQMR
    - GMRES
    - LSQR

We need acceleration techniques.
MOM: Evaluating the Interactions

**EFIE**

\[
Z_{mn}^E = \int_{S_m} dr t_m(r) \cdot \int_{S_n} dr' \mathcal{G}(r, r') \cdot b_n(r')
\]

**MFIE**

\[
Z_{mn}^M = \int_{S_m} dr t_m(r) \cdot b_n(r)
\]

\[
- \int_{S_m} dr t_m(r) \cdot \hat{n} \times \int_{S_n} dr' b_n(r') \times \nabla' g(r, r')
\]

Fast Multipole Method (FMM)

- Addition theorem (factorization of Green’s function)

\[
g(r, r') = \frac{\exp(ik|D + d|)}{4\pi|D + d|} = \frac{ik}{4\pi} \sum_{l=0}^{\infty} (-1)^l (2l + 1) j_l(kd) h_i^{(1)}(kD) P_l(\hat{d} \cdot \hat{D})
\]

\[
d \leq D
\]

- Plane-wave expansion

\[
4\pi i^l j_l(kd) P_l(\hat{d} \cdot \hat{D}) = \int d^2 \hat{k} \exp(i k \cdot d) P_l(\hat{k} \cdot \hat{D})
\]

- Diagonalization

\[
\exp(ik|D + d|) \approx \frac{1}{4\pi} \int d^2 \hat{k} \exp(i k \cdot d) T_L(k, \hat{k}, D)
\]

- Translation function

\[
T_L(k, \hat{k}, D) = \frac{ik}{4\pi} \sum_{l=0}^{L} i^l (2l + 1) h_i^{(1)}(kD) P_l(\hat{k} \cdot \hat{D})
\]
FMM: Evaluating the Interactions

\[ \mathbf{\overline{G}}(r, r') \]

\[ \approx \frac{1}{4\pi} \int d^2 \hat{k} (\mathbf{I} - \hat{k}\hat{k}) e^{i\hat{k}\cdot(r_{fm} + r_{fmc} + r_{fnc} + r_{fnc}')} T_L(k, |r_{cc'}|, \hat{r}_{cc'} \cdot \mathbf{k}) \]

- **EFIE**

\[ Z_{mn}^E = \frac{1}{4\pi} \int d^2 \hat{k} F_{fmc}^E(\hat{k}) T_L(k, |r_{cc'}|, \hat{r}_{cc'} \cdot \mathbf{k}) \cdot F_{fnc'}^E(\hat{k}) \]

- **MFIE**

\[ Z_{mn}^M = \frac{k}{4\pi i} \int d^2 \hat{k} F_{fmc}^M(\hat{k}) T_L(k, |r_{cc'}|, \hat{r}_{cc'} \cdot \mathbf{k}) \cdot F_{fnc'}^M(\hat{k}) \]

**EFIE**

\[ F_{fmc}^E(\hat{k}) = e^{ik\cdot(r_{fm} - r_e)} \int_{S_m} dr e^{ik\cdot(r - r_{fm})} (\mathbf{I} - \hat{k}\hat{k}) \cdot \mathbf{t}_m(r) \]

\[ F_{fnc'}^E(\hat{k}) = e^{ik\cdot(r_{c'} - r_{fn})} \int_{S_n} dr' e^{-ik\cdot(r' - r_{fn})} (\mathbf{I} - \hat{k}\hat{k}) \cdot \mathbf{b}_n(r') \]

**MFIE**

\[ F_{fmc}^M(\hat{k}) = -\hat{k} \times e^{ik\cdot(r_{fm} - r_e)} \int_{S_m} dr e^{ik\cdot(r - r_{fm})} \mathbf{t}_m(r) \times \hat{n} \]

\[ F_{fnc'}^M(\hat{k}) = e^{ik\cdot(r_{c'} - r_{fn})} \int_{S_n} dr' e^{-ik\cdot(r' - r_{fn})} \mathbf{b}_n(r') \]

Prof. Levent Gürel

http://abakus.computing.technology/
FMM Considerations

1) Truncation of infinite summation:

\[ T_L(k, D, \theta) = \frac{i k}{4\pi} \sum_{l=0}^{L} i^l (2l + 1) h_l^{(1)}(kD) P_l(\cos\theta) \]

\[ L \approx kd + 1.8 d_0^{2/3} (kd)^{1/3} \]

2) Angular integration:

\[ Z_{mn}^M = \frac{k}{4\pi i} \int d^2 \hat{k} F_{fmc}^M(\hat{k}) T_L(k, |\mathbf{r}_{cc'}|, \hat{r}_{cc'} \cdot \hat{k}) \cdot F_{fnc'}^M(\hat{k}) \]

\[ K=2L^2 \]

Angular Integration

Angular integration points for \( L = 5 \)
Fast Multipole Method

\[
\sum_{n=1}^{N} Z_{mn} a_n = \sum_{G' \in N(G)} \sum_{n \in G} Z_{mn} a_n + \frac{1}{4\pi} \int d^2 \hat{k} \ F_{mG}(\hat{k}) \cdot \sum_{G' \notin N(G)} T_L(k, r_{GG'}) \sum_{n \in G'} F_{nG'}(\hat{k}) a_n
\]

Near-field interactions

\[m = 1, 2, \ldots, N\]

Far-field interactions

Aggregation

Translation

Disaggregation

Cluster of basis functions

Translation

Cluster of testing functions

Aggregation

Disaggregation

\[
\sum_{n=1}^{N} Z_{mn} a_n = \frac{1}{4\pi} \int d^2 \hat{k} \ F_{mG}(\hat{k}) \cdot F_T^{*}(\hat{k})
\]
Multilevel Fast Multipole Algorithm

- Complexity: $O(N \log N)$

Recursive clustering

Tree structure

Radiated or incoming fields
Integral Equations, Fast Algorithms, and Parallelization Strategies for the Solution of Extremely Large Problems in Computational Electromagnetics - Part 1

MLFMA Tree Structure

Level: 3
Level: 2
Unknown Level

Recursive Clustering
Multilevel Fast Multipole Algorithm

- **Tree structure:**

  - Radiated or incoming fields

  - **Complexity:** $O(N \log N)$

- Consider aggregation on the entire tree-structure.
• Consider translation on the entire tree-structure.

Far-field sub-clusters

Incoming field to sub-cluster

Radiation of sub-cluster

Touching (close) parent clusters
**Acceleration with MLFMA**

• Solve \( \sum_{n=1}^{N} Z_{mn} a_n = v_m \)  
  
  Matrix-Vector Multiplication  
  \( \mathbf{Z} \cdot \mathbf{x} = \mathbf{y} \)  

• Processing time for a matrix-vector product: 
  
  MOM [O(N^2)] \rightarrow FMM [O(N^{3/2})] \rightarrow MLFMA [O(N \log N)]

• Memory requirement: 
  
  MOM [O(N^2)] \rightarrow FMM [O(N^{3/2})] \rightarrow MLFMA [O(N \log N)]

---

**Multilevel FMM**

• Apply the FMM concept in multi-level scheme: 
  
  Group the groups!

• Form tree structure:

---

Prof. Levent Gürel  
http://abakus.computing.technology/
Multilevel FMM

Aggregation between the levels

Parent Clusters

Level $i$

Clusters

Radiation of the parent clusters

Level $i+1$

Clusters

Radiation of the clusters

Interpolation

Prof. Levent Gürel
http://abakus.computing.technology/

Integral Equations, Fast Algorithms, and Parallelization Strategies for the Solution of Extremely Large Problems in Computational Electromagnetics - Part 1

Prof. Levent Gürel
http://abakus.computing.technology/
Multilevel FMM

Translations: Required in each level.

\[
F_T^C(\hat{k}) = \sum_{P(C'_i) \in N(P(C_i))} \sum_{C'_i \in F(C_i)} T_L(k, |r_{cc'}, \hat{r}_{cc'} \cdot \hat{k}) F_{C'_i}^C(\hat{k})
\]

\[\tau(l) \approx 1.73k\alpha_l + 2.16(d_0)^{2/3}(k\alpha_l)^{1/3}\]

Use symmetry for efficiency:

Disaggregation between the levels

Parent Clusters

Incoming wave for the parent clusters

Anterpolation

Clusters

Incoming wave for the clusters
Disaggregation:

\[ F_{TD}^{C_{i+1}}(\hat{k}) = F_{T}^{C_{i+1}}(\hat{k}) + \overline{I}_{i+1,i} \cdot \beta_{C_{i+1}}^{C_i} F_{TD}^{P(C_{i+1})}(\hat{k}) \]

Translation \quad Disaggregation
Interpolation and Anterpolation

Integral Equations, Fast Algorithms, and Parallelization Strategies for the Solution of Extremely Large Problems in Computational Electromagnetics - Part 1

Interpolation
Lagrange interpolation employing 4×4 points (shaded circles) located in the coarse grid to evaluate the function at a point (star) located in the fine grid. Sampling values of $\theta$ and $\phi$ are specified in radians and selected from a practical case.
Multilevel FMM

Between the levels interpolation and anterpolation algorithms are required:

Error Source: Interpolation between levels

- Processing time for a matrix-vector product:

\[
\text{MOM } [O(N^2)] \rightarrow \text{FMM } [O(N^{3/2})] \rightarrow \text{MLFMA } [O(N\log N)]
\]

- Memory requirement:

\[
\text{MOM } [O(N^2)] \rightarrow \text{FMM } [O(N^{3/2})] \rightarrow \text{MLFMA } [O(N\log N)]
\]

Multilevel Fast Multipole Algorithm

Aggregation:
- Performed from bottom to top
- Local interpolations are used
- \(O(N)\) operations per level

Translation:
- \(O(1)\) testing clusters for each basis cluster
- \(O(N)\) operations per level
- \(O(1)\) different translation operators per level
(for cubic clusters)
## Complexity of MLFMA

Major Parts of MLFMA and Their Computational Requirements

<table>
<thead>
<tr>
<th>Part</th>
<th>Memory</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>MVM</td>
<td>$\sum_{l=1}^{L} N_l \tau(l) + 1^2$</td>
<td>$O(N \log N)$</td>
</tr>
<tr>
<td>Radiation and Receiving Patterns</td>
<td>$N\tau(1) + 1^2$</td>
<td>$O(N)$</td>
</tr>
<tr>
<td>Translation Operators</td>
<td>$\sum_{l=1}^{L} d_l \tau(l) + 1^2$</td>
<td>$O(N)$</td>
</tr>
<tr>
<td>Near-Field Interactions</td>
<td>$N^2 / N_1$</td>
<td>$O(N)$</td>
</tr>
</tbody>
</table>

### Processing Time

<table>
<thead>
<tr>
<th>Part</th>
<th>Processing Time</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>MVM</td>
<td>$\sum_{l=1}^{L} c_l N_l \tau(l) + 1^2$</td>
<td>$O(N \log N)$</td>
</tr>
<tr>
<td>Radiation and Receiving Patterns</td>
<td>$N\tau(1) + 1^2$</td>
<td>$O(N)$</td>
</tr>
<tr>
<td>Translation Operators</td>
<td>$\sum_{l=1}^{L} d_l \tau(l) + 1^2$</td>
<td>$O(N)$</td>
</tr>
<tr>
<td>Near-Field Interactions</td>
<td>$N^2 / N_1$</td>
<td>$O(N)$</td>
</tr>
</tbody>
</table>

Note: $c_l$ and $d_l$ represent relative weights for levels $l = 1, 2, \ldots, L$.

### Fast Multipole Methods

Reduced-Complexity Solutions

- **Electrodynamics** (Helmholtz’s Equation)
- **Electrostatics** (Laplace’s Equation)
- **Astrophysics** (Gravitational Force)
- **Acoustics**
- **Structural Mechanics**
- **Fluid Dynamics**
- **Quantum Mechanics** (Schroedinger’s Equation)
- **Molecular Dynamics**
Fast and Efficient Algorithms in Computational Electromagnetics
Integral Equations, Fast Algorithms, and Parallelization Strategies for the Solution of Extremely Large Problems in Computational Electromagnetics

Prof. Levent Gürel
CEO, ABAKUS Computing Technologies
Adjunct Professor, ECE, Univ. of Illinois at Urbana-Champaign

May 2015

Parallel MLFMA
(Multilevel Fast Multipole Algorithm)

Prof. Levent Gürel
CEO, ABAKUS Computing Technologies
Adjunct Professor, ECE, Univ. of Illinois at Urbana-Champaign

May 2015
What is the Main Source of Efficiency?

<table>
<thead>
<tr>
<th>$N$ Unknowns</th>
<th>$O(N^3)$ Gaussian Elimination</th>
<th>$O(N^2)$ Iterative MOM (MVM)</th>
<th>$O(N^{3/2})$ Single-Level FMM</th>
<th>$O(N \log N)$ Multi-Level FMM</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000</td>
<td>1 s</td>
<td>2 s</td>
<td>4 s</td>
<td>8 s</td>
</tr>
<tr>
<td>$10^6$</td>
<td>32 years</td>
<td>23 days</td>
<td>35 h</td>
<td>7 h</td>
</tr>
<tr>
<td>$10^7$</td>
<td>32 K years</td>
<td>6.3 years</td>
<td>46 days</td>
<td>89 h</td>
</tr>
<tr>
<td>$10^8$</td>
<td>32 M years</td>
<td>630 years</td>
<td>4 years</td>
<td>46 days</td>
</tr>
<tr>
<td>$10^9$</td>
<td>32 G years</td>
<td>63 K years</td>
<td>127 years</td>
<td>1.5 years (555 days)</td>
</tr>
</tbody>
</table>

Parallelization of MLFMA

Parallelization is required

- For the solution of realistic problems discretized with tens of millions of unknowns
- On relatively inexpensive computing platforms
  - 64-128 cores
  - Distributed memory
  - Fast networks such as Infiniband

Unfortunately, parallelization of MLFMA is not trivial!
Parallel Computers

Integral Equations, Fast Algorithms, and Parallelization Strategies for the Solution of Extremely Large Problems in Computational Electromagnetics - Part 2

Multilevel Fast Multipole Algorithm

**Setup**
- Calculation of Nearfield Interactions
- Calculation of Radiation & Receiving Patterns
- Calculation of Translation Operators

**Iterative Solution**
- Matrix-vector multiplications
- Using Nearfield Interactions
- Aggregation & Disaggregation
- Translation
Multilevel Fast Multipole Algorithm

• Tree structure:

\[ \text{Radiated or incoming fields} \]

\[ \text{Complexity: } O(N \log N) \]

Multilevel Tree Structure

- Higher levels:
  - Large numbers of samples
  - Small numbers of clusters

- Lower levels:
  - Small numbers of samples
  - Large numbers of clusters
(Simple) Partitioning of Multilevel Tree

Main work: Partitioning the tree structure

- Simple partitioning
  - Clusters are distributed in all levels
  - Inefficient for high levels (poor load-balance)

Tree structure

Field samples

Clusters

Multilevel Fast Multipole Algorithm

• Tree structure:
**Tree structure:**

- Poor load balancing!

**Hybrid partitioning**
(improves load-balance for higher levels)

- **Shared levels:**
  - Clusters are shared
  - Fields are distributed

- **Distributed levels:**
  - Clusters are distributed

Problem: There are some levels, where neither distributing fields nor clusters is efficient.
(Hierarchical) Partitioning of Multilevel Tree

- **Hierarchical partitioning**

  Fields and clusters are “perfectly” distributed

(Simple) Partitioning of Multilevel Tree

**Main work: Partitioning the tree structure**

- **Simple partitioning**
  - Clusters are distributed in all levels
  - Inefficient for high levels (poor load-balance)
Simple Parallelization

Distributing Clusters Among Processors

Aggregations

Translations

Near-Field Interactions

- Near-field partitioning
- Sphere problem (829,881 unknowns)

Rows equally distributed

Single iteration

Waits

Radiation and Receiving Patterns

Near-field interactions

Load imbalance

Prof. Levent Gürel

http://abakus.computing.technology/
Simple Parallelization

Processing Time (Seconds) for Sphere Problem (829,881 unknowns)
Matrix-Vector Multiplications

Translations

Inter-Processor Translations
Self-Processor Translations
Aggregation
Disaggregation
Wait

Prof. Levent Gürel
http://abakus.computing.technology/
Speed-up and Gain

Sphere Problem (132,003 unknowns)

![Graph showing speed-up and gain]

Prof. Levent Gürel                      http://abakus.computing.technology/

Partitioning

Simple parallelization  
(All levels are distributed)

Hybrid parallelization  
(Shared & distributed levels)

Problems arise in distributing fields
Hybrid Partitioning of Multilevel Tree

**Hybrid partitioning**
(improves load-balance for higher levels)

- **Shared levels:**
  - Clusters are shared
  - Fields are distributed

- **Distributed levels:**
  - Clusters are distributed

Problem: There are some levels, where neither distributing fields nor clusters is efficient

Far-Field Interactions

**HIGH LEVELS**
Small numbers of clusters
Large numbers of samples

**LOW LEVELS**
Large numbers of clusters
Small numbers of samples

Level of Distribution (LoD)
between shared and distributed levels

- Distribute fields among the processors
- Apply load-balancing in this level by considering bunch of subclusters below
- Distribute clusters among the processors

Hybrid Parallelization

- For Low Levels of MLFMA: Distribute Clusters Among Processors (Distributed Levels)
- For High Levels of MLMFA: Distribute Fields Among the Processors (Shared Levels)


**Aggregation / Disaggregation**

- Aggregation and disaggregation in distributed levels
Aggregation / Disaggregation

- Aggregation and disaggregation in distributed levels

- Aggregation and disaggregation in shared levels

Hybrid Parallelization

Level of Distribution (LoD):
Connection Between Distributed and Shared Levels

Radiated or Incoming Fields of a Cluster Stored at Single Processor (P1)
Radiated or Incoming Fields of a Cluster Distributed Among Processors
Aggregation / Disaggregation

- All-to-All communications in LoD

- Aggregation and disaggregation in the distributed levels are communication-free (no communications).

- In the shared levels, fields are partitioned along field samples.

- Aggregation and disaggregation in the shared levels require one-to-one communications

Prof. Levent Gürel
http://abakus.computing.technology/

Hybrid Parallelization

Aggregation in Shared Levels

Lower Level

P1
P2
P3
P4

Upper Level

P1
P2
P3
P4

Interpolation

One-to-One Communications Are Performed to Obtain the Required Data for the Interpolation

Prof. Levent Gürel
http://abakus.computing.technology/
Aggregation / Disaggregation

LEVEL i

\[ p = 1 \]
\[ p = 2 \]
\[ p = 3 \]
\[ p = 4 \]

LEVEL i + 1

Fields

Inflation

Interpolation & Shift

From “left”

From “right”

• Data to \( p = 2 \)

• Reduce communications by load-balancing

Translation

• Distributed levels
  One-to-one communications are required

• Processor pairing

• Shared Levels (Communication-free)

Prof. Levent Gürel
http://abakus.computing.technology/
Integral Equations, Fast Algorithms, and Parallelization Strategies for the Solution of Extremely Large Problems in Computational Electromagnetics - Part 2

Communications

**USING NEARFIELD INTERACTIONS**
- All-to-One & One-to-All

**DISTRIBUTED AGGREGATION**
- All-to-All

**SHARED AGGREGATION**
- One-to-One

**DISTRIBUTED TRANSLATION**
- One-to-One

**SHARED TRANSLATION**

**SHARED DISAGGREGATION**
- All-to-One

**DISTRIBUTED DISAGGREGATION**
- One-to-One
- All-to-All

![Full input vector](Image)

\[ y = Z^{NF} \cdot x \]

- **Output vector with near-field partitioning**
  - All-to-One

- **Output vector with far-field partitioning**
  - One-to-All

**Efficiency Results**

- **Setup Efficiency**
- **Solution Efficiency**
- **Overall Efficiency**

**Intel Xeon 5355 processors**
**Infiniband network**

- 1,462,854 unknowns
- 20\(\lambda\)

Prof. Levent Gürel  
http://abakus.computing.technology/
Integral Equations, Fast Algorithms, and Parallelization Strategies for the Solution of Extremely Large Problems in Computational Electromagnetics - Part 2

Efficiency Results

Setup Efficiency

Solution Efficiency

Overall Efficiency

Intel Xeon 5355 processors
Infiniband network

Prof. Levent Gürel

5,851,416 unknowns

74 %

53 %

13,278,096 unknowns

74 %

58 %

http://abakus.computing.technology/
Hybrid Parallelization

Efficiency for the Overall Processing Time

- Helicopter Problem
- Rectangular Sheet Problem

Efficiency Results

13,278,096 unknowns

- Translations with communications (some of those in distributed levels) are problematic.
- Increase number of shared layers?
- Aggregations/disaggregations in the shared levels are also problematic, when the number of processors is large and the number of samples is small.
Hybrid Parallelization

Processing Time (Seconds) for Helicopter Problem (117,366 unknowns)
Matrix-Vector Multiplications

Translations for Shared Levels (Communication-Free)
Translations for Distributed Levels
Load-Balanced Aggregation for Shared Levels
Load-Balanced Aggregation for Distributed Levels (Communication-Free)

Near-field Interactions

Steps of Efficient Parallelization for MLFMA

- Apply a Load-Balancing Algorithm for Near-Field Setup
- Divide the Levels into Distributed and Shared Layers
- For Distributed Levels, Apply a Load-Balancing Algorithm to Distribute Clusters Among Processors
- Perform All-to-All Communications to Pass from Distributed Levels to Shared Levels
- For Shared Levels, Apply a Load-Balancing Algorithm to Distribute Fields Among Processors
- For Translations in Distributed Levels, Apply a Communication Algorithm to Control the Data Traffic
Partitioning

**Simple parallelization**  
(All levels are distributed)

- Difficult to distribute clusters

**Hybrid parallelization**  
(Shared & distributed levels)

- Easy to distribute fields

- Problems arise in distributing fields

---

**Partitioning of the Tree Structure**

*Hierarchical Partitioning*

- Fields and clusters are “perfectly” distributed
Hierarchical Parallelization

• Hierarchical Partitioning: Fields and clusters are distributed simultaneously

![Multilevel tree structure]

Level 4
Level 3
Level 2
Lowest level (Level 1)

Field Samples (theta)
Field Samples (phi)
Clusters

(Hierarchical) Partitioning of Multilevel Tree

• Hierarchical partitioning

Fields and clusters are “perfectly” distributed

• Define intermediate levels

Level 3
Modify partitioning
Level 2.5
Aggregation/Disaggregation
Level 2

Prof. Levent Gürel
http://abakus.computing.technology/
Communications

Consider:
- Level 2
- Process(or) 4

Communications for translations

Communications for data exchange

Communications for aggregation/disaggregation

Level 2.5
Modify partitioning
Level 3

Aggregation

- Aggregation

Level 2

2 4 6 8
1 3 5 7

Interpolation & Shift

One-to-one communications are required (for interpolations)

Inflation
Data Exchange

- **Data exchange to modify the partitioning**

```
Level 2.5  2 4 6 8  1 3 5 7
         2 4 6 8
         1 3 5 7
```

```
Level 3  4 8  2 6  3 7  1 5
```

Translation

- **Translations**

```
Level 2  2 4 6 8  1 3 5 7
```

```
Intra-processor translations
```

```
Inter-processor translations
```

- **Possible communications**
  - Among \{1,3,5,7\}
  - Among \{2,4,6,8\}

- **Processor pairing**
Inter-Processor Translations

Perform MVM for near-field interactions

Intra-Processor Translations

For each level
One-to-one communication
Translate

Inter-Processor Translations

For each level
One-to-one communication
Interpolate & Shift
Data exchange (One-to-one)

Disaggregation (Reverse of Aggregation)

Advantages of the Hierarchical Strategy

- Improved load-balancing and reduced communications
- The amount of communications is decreased
- The number of communication events is reduced

<table>
<thead>
<tr>
<th></th>
<th>HYBRID*</th>
<th></th>
<th>HIERARCHICAL</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Events</td>
<td>Amount</td>
<td>Events</td>
</tr>
<tr>
<td>Interpolations</td>
<td>8320</td>
<td>1,676,352</td>
<td>2188</td>
</tr>
<tr>
<td>Data Exchanges</td>
<td>0</td>
<td>0</td>
<td>10</td>
</tr>
<tr>
<td>Switch</td>
<td>1160</td>
<td>171,680</td>
<td>0</td>
</tr>
<tr>
<td>Total Aggregation</td>
<td>9480</td>
<td>1,848,032</td>
<td>2198</td>
</tr>
<tr>
<td>Translation</td>
<td>6375</td>
<td>2,416,780</td>
<td>7215</td>
</tr>
<tr>
<td>TOTAL</td>
<td>15855</td>
<td>4,264,812</td>
<td>9413 (59%)</td>
</tr>
<tr>
<td>Average Package Size</td>
<td>269 Bytes</td>
<td></td>
<td>332 Bytes (123%)</td>
</tr>
</tbody>
</table>
Parallel Computers

Results (Efficiency)

Solution Efficiency (%)

Overall Efficiency (%)

System:
2.66 GHz Intel Xeon 5355 processors
Infiniband network

Parameters:
7-level MLFMA (Bottom-up clustering)
3-digits of accuracy
1e-6 residual error (27 iterations)

1,462,854 unknowns
Results (Efficiency)

**Solution Efficiency (%)**
- 59% with 5,851,416 unknowns

**System:**
- 2.66 GHz Intel Xeon 5355 processors
- Infiniband network

**Parameters:**
- 8-level MLFMA (Bottom-up clustering)
- 3-digits of accuracy
- 1e-6 residual error (30 iterations)

**的整体效率 (%)**
- 68%

**System:**
- 2.66 GHz Intel Xeon 5355 processors
- Infiniband network

**Parameters:**
- 8-level MLFMA (Bottom-up clustering)
- 3-digits of accuracy
- 1e-6 residual error (43 iterations)

---

Results (Efficiency)

**Solution Efficiency (%)**
- 86% with 13,278,096 unknowns

**System:**
- 2.66 GHz Intel Xeon 5355 processors
- Infiniband network

**Parameters:**
- 8-level MLFMA (Bottom-up clustering)
- 3-digits of accuracy
- 1e-6 residual error (30 iterations)
### Results (Large Problems)

#### 23,405,664 unknowns, 16 processors, BDP

<table>
<thead>
<tr>
<th>Number of Levels</th>
<th>9</th>
<th>Setup Time</th>
<th>104 minutes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Smallest Box Size</td>
<td>0.156(\lambda)</td>
<td>Solution Time</td>
<td>178 minutes</td>
</tr>
<tr>
<td>Truncation Number</td>
<td>5 - 457</td>
<td>Iterations</td>
<td>17 BiCGStab</td>
</tr>
<tr>
<td>FMM Accuracy</td>
<td>2 digits</td>
<td>MVM Time</td>
<td>307 seconds</td>
</tr>
<tr>
<td>Residual</td>
<td>0.001</td>
<td>Total Memory</td>
<td>111 GB</td>
</tr>
</tbody>
</table>

![Graph showing processing and MVM time for 16 processors](image1)

---

### Results (Large Problems)

#### 33,791,232 unknowns, 16 processors, BDP

<table>
<thead>
<tr>
<th>Number of Levels</th>
<th>9</th>
<th>Setup Time</th>
<th>205 minutes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Smallest Box Size</td>
<td>0.188(\lambda)</td>
<td>Solution Time</td>
<td>289 minutes</td>
</tr>
<tr>
<td>Truncation Number</td>
<td>6 - 546</td>
<td>Iterations</td>
<td>21 BiCGStab</td>
</tr>
<tr>
<td>FMM Accuracy</td>
<td>2 digits</td>
<td>MVM Time</td>
<td>406 seconds</td>
</tr>
<tr>
<td>Residual</td>
<td>0.001</td>
<td>Total Memory</td>
<td>179 GB</td>
</tr>
</tbody>
</table>

![Graph showing processing and MVM time for 16 processors](image2)
Results (Large Problems)

41,883,648 unknowns, 16 processors, BDP

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Levels</td>
<td>9</td>
</tr>
<tr>
<td>Smallest Box Size</td>
<td>0.215λ</td>
</tr>
<tr>
<td>Truncation Number</td>
<td>6 - 623</td>
</tr>
<tr>
<td>FMM Accuracy</td>
<td>2 digits</td>
</tr>
<tr>
<td>Residual</td>
<td>0.001</td>
</tr>
<tr>
<td>Setup Time</td>
<td>313 minutes</td>
</tr>
<tr>
<td>Solution Time</td>
<td>314 minutes</td>
</tr>
<tr>
<td>Iterations</td>
<td>19 BiCGStab</td>
</tr>
<tr>
<td>MVM Time</td>
<td>467 seconds</td>
</tr>
<tr>
<td>Total Memory</td>
<td>223 GB</td>
</tr>
<tr>
<td>Number of Levels</td>
<td>9</td>
</tr>
<tr>
<td>Smallest Box Size</td>
<td>0.215λ</td>
</tr>
<tr>
<td>Truncation Number</td>
<td>6 - 623</td>
</tr>
<tr>
<td>FMM Accuracy</td>
<td>2 digits</td>
</tr>
<tr>
<td>Residual</td>
<td>0.001</td>
</tr>
</tbody>
</table>

Prof. Levent Gürel
http://abakus.computing.technology/

53 Million Unknowns

Sphere with radius of 120λ and diameter of 240λ

November 2007

53,112,384 Unknowns

Prof. Levent Gürel
http://abakus.computing.technology/
85 Million Unknowns
January 2008

Sphere with radius of $150\lambda$ and diameter of $300\lambda$

Scattering results are compared to analytical values to demonstrate the accuracy of the numerical solution.

85,148,160 Unknowns

Prof. Levent Gürel  
http://abakus.computing.technology/
135 Million Unknowns
August 2008

Sphere with radius of $180\lambda$ and diameter of $360\lambda$
Scattering results are compared to analytical values to demonstrate the accuracy of the numerical solution.

135,164,928 Unknowns

Scattering results are compared to analytical values to demonstrate the accuracy of the numerical solution.

167 Million Unknowns
August 2008

Sphere with radius of $190\lambda$ and diameter of $380\lambda$
Scattering results are compared to analytical values to demonstrate the accuracy of the numerical solution.

167,534,592 Unknowns
Scattering results are compared to analytical values to demonstrate the accuracy of the numerical solution.

205 Million Unknowns

Sphere with radius of $210\lambda$ and diameter of $420\lambda$

September 2008

204,823,296 Unknowns

Scattering results are compared to analytical values to demonstrate the accuracy of the numerical solution.

307 Million Unknowns

Sphere with radius of $260\lambda$ and diameter of $520\lambda$

December 2009

307,531,008 Unknowns

Scattering results are compared to analytical values to demonstrate the accuracy of the numerical solution.
375 Million Unknowns
December 2009

Sphere with radius of $280\lambda$ and diameter of $560\lambda$

Scattering results are compared to analytical values to demonstrate the accuracy of the numerical solution.

374,490,624 Unknowns

540 Million Unknowns
September 2010

Sphere with radius of $340\lambda$ and diameter of $680\lambda$

Scattering results are compared to analytical values to demonstrate the accuracy of the numerical solution.

540,659,712 Unknowns
670 Million Unknowns

Sphere with radius of $340\lambda$ and diameter of $680\lambda$

Scattering results are compared to analytical values to demonstrate the accuracy of the numerical solution.

670 Million Unknowns

RCS of a Sphere with 670M Unknowns

850 Million Unknowns

Sphere with radius of $440\lambda$ and diameter of $880\lambda$

Scattering results are compared to analytical values to demonstrate the accuracy of the numerical solution.

850 Million Unknowns
1.1 Billion Unknowns

Sphere with radius of $500\lambda$ and diameter of $1000\lambda$

Scattering results are compared to analytical values to demonstrate the accuracy of the numerical solution.

BENCHMARKING

Available from
www.abakus.computing.technology

Scattering from sphere (radius: $20\lambda - 340\lambda$)
Web-based application: Upload the computational results and get the error with respect to analytical Mie-series solutions.

Scattering from NASA Almond (size: $94\lambda - 1514\lambda$)
Web-based application: Upload the computational results and get the error with respect to our results.
Integral Equations, Fast Algorithms, and Parallelization Strategies for the Solution of Extremely Large Problems in Computational Electromagnetics - Part 2

WEB-BASED BENCHMARKING TOOL

**Metamaterials**

**Split-Ring Resonators**
Meta Property: Stop band in frequency!

Special Configurations

Closed-Ring Resonators (CRRs)

Thin Wire Array (TWA)

All pass!  No pass!
Composite Metamaterial (CMM)

Split-Ring Resonators (SRRs)

Thin Wire Array (TWA)

Meta Property: Pass band in frequency!
Large MM Problem

20 x 51 x 29 SRR Block

2,425,560 Unknowns
9381 Seconds

MLFMA parallelized into 64 process on a cluster of Intel-Xeon processors connected via Infiniband network

Scattering from and Imaging of Red Blood Cells
Red Blood Cells (RBCs)

Ordinary (Healthy)
Red Blood Cell

Macrocyte
(Macrocytosis)

Microcyte
(Microcytosis)

Sickle Cell
(Sickle Cell Anemia)
Flow Cytometry

Blood sample

Forward scattering

Side scattering

Backscattering

Source: http://nirfriedmanlab.blogspot.in/2010_04_01_archive.html

Forward Scattering

"Detection" of an Extra-Ordinary Cell

"No Detection"

"Detection" of an Extra-Ordinary Cell
An extra-ordinary cell can be detected for all orientations

"Detection" of an Extra-Ordinary Cell

"No Detection"

"Detection" of an Extra-Ordinary Cell

An extra-ordinary cell may have "normal" SCS values

An extra-ordinary cell can be detected for some orientations

Decision Chart

FORWARD SCATTERING

<43 dB\textmu m^2

Detected

No Detection

>44 dB\textmu m^2

Detected

BACK SCATTERING

<25 dB\textmu m^2

Detected

No Detection

<24 dB\textmu m^2

Detected

SIDE SCATTERING

<15 dB\textmu m^2

Detected

No Detection

>15 dB\textmu m^2

Detected

Macrocytosis

Microcytosis

Healthy

Spherocytosis

Sickle Cell Anemia
Integral Equations, Fast Algorithms, and Parallelization Strategies for the Solution of Extremely Large Problems in Computational Electromagnetics - Part 2

Multi-Disciplinary Approach

- **Computer Engineering**
  - Parallel Architectures
  - Multicore Processors

- **Mathematics**
  - Integral Equations
  - Numerical Analysis

- **Geometry Modelling**
  - 3-D CAD Modelling
  - Meshing

- **Physics**
  - Electromagnetic Theory
  - Radiation and Scattering

- **Linear Algebra**
  - Iterative Solvers
  - Preconditioners

- **Computer Science**
  - Fast Algorithms
  - Parallel Programming

---

Prof. Levent Gürel

http://abakus.computing.technology/