Efficient large electromagnetic simulation based on hybrid TLM and modal approach on grid computing and supercomputer

Mihai Alexandru

Université de Toulouse

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par :
Institut National Polytechnique de Toulouse (INP Toulouse)

Discipline ou spécialité :

Micro-ondes, Electromagnétisme et Optoélectronique

Présentée et soutenue par :
Mihai ALEXANDRU

le : vendredi 14 décembre 2012

Titre :

Efficient large electromagnetic simulation based on
hybrid TLM and modal approach
on grid computing and supercomputer

Ecole doctorale :
Génie Electrique, Electronique et Télécommunications (GEET)

Unité de recherche :
LAAS-CNRS

Directeur(s) de Thèse :
PR Hervé AUBERT
MCU Thierry MONTEIL

Rapporteurs :

PR Serge VERDEYME
DR Christian PEREZ

Membre(s) du jury :

PR Roberto SORRENTINO, PR Serge VERDEYME, DR Christian PEREZ, PR Hervé AUBERT,
MCU Thierry MONTEIL, Dr Fabio Coccetti, Dr Petr LORENZ

THÈSE

en vue de l'obtention du

Doctorat de l'Université de Toulouse

délivré par
l'**Institut National Polytechnique de Toulouse (INP Toulouse)**

Ecole Doctorale : Génie Electrique, Electronique et Télécommunications (GEET)

Discipline : Micro-ondes, Electromagnétisme et Optoélectronique

présentée et soutenue par

Mihai ALEXANDRU

le 14 décembre 2012

Efficient large electromagnetic simulation based on hybrid TLM and modal approach on grid computing and supercomputer

Jury

M. Serge VERDEYME	Rapporteur
M. Christian PEREZ	Rapporteur
M. Hervé AUBERT	Directeur de thèse
M. Thierry MONTEIL	Co-Directeur de thèse
M. Roberto SORRENTINO	Examinateur
M. Fabio COCCETTI	Examinateur
M. Petr LORENZ	Invité

I would like to dedicate this thesis to my loving parents ...

Remerciements

Ce travail a été réalisé au Laboratoire d'Analyse et d'Architecture des Systèmes (LAAS) à Toulouse, au sein des groupes de recherche MINC et MRS (devenu par la suite SARA).

J'exprime ma profonde gratitude à l'égard de Messieurs Hervé Aubert, Professeur à l'ENSEEIHT et Thierry Monteil, Maître des Conférences à l'INSA de Toulouse, pour m'avoir proposé ce sujet de recherche et pour avoir dirigé mon parcours. Leur dynamisme et leurs compétences scientifiques ont constitué le support qui m'a permis d'aboutir aux résultats de cette thèse.

J'adresse toute l'expression de ma reconnaissance aux Messieurs Serge Verdeyme, Professeur à l'Université de Limoges, et Christian Perez, Directeur de recherche à l'INRIA de Lyon, pour avoir accepté d'être rapporteurs de ma thèse. Je les remercie notamment pour l'intérêt manifesté pour ce travail.

Je suis très sensible à la présence dans mon jury de thèse de Monsieur Roberto Sorrentino, Professeur à l'Université de Perugia, Italie, qui m'a fait l'honneur de présider le jury. Je le remercie pour l'examen attentif porté à ma thèse.

Je remercie vivement Monsieur Fabio Coccetti, chargé de recherche au LAAS-CNRS, pour l'aide apportée à mon travail tout au long de la thèse et pour avoir pris part à ce jury.

J'exprime également ma profonde gratitude à Monsieur Petr Lorenz pour le support dans la validation du code de calcul.

Toute mon amitié à Cristian Ruiz et Tom Guérout pour les collaborations fructueuses que nous avons menées ensemble.

Je remercie également Alexandru Takacs et Daniela Dragomirescu pour les moments passés ensemble, pour leurs conseils et leur gentillesse.

J'exprime toute mon amitié à Nuria Torres Matabosch et Mariano Ercoli avec qui j'ai partagé le bureau pendant présque trois ans. Je leur souhaite beaucoup de bien.

Un GRAND merci à tous mes collègues de laboratoire, pour les discussions très variées tout au long de ces trois années: Dina Medhat, Giancarlo Vincenzi, Hong Liu (Patricia), Franck Chebila, Tonio Idda, Farook Ahmad Tahir, Badreddine

Ougague, Florian Perget, Rosa De Paolis, Sofiene Bouaziz, Khaldoun Saleh, Serge Karboyan, Aurélien Gonzalez, Rémi Sharrock, Ahmad Al Sheikh, Jean-Marie Codol, Robert Guduvan, Miruna Stoicescu, Georgia Deaconu et Roxana Albu.

Enfin, je tiens à remercier le LAAS, notamment le service du personnel, le service Gestion financière, le magasin, le service Logistique et infrastructure, le service Information Scientifique et Technique-Edition.

Abstract

In the context of Information Communications Technology (ICT), the major challenge is to create systems increasingly small, boarding more and more intelligence, hardware and software, including complex communicating architectures. This requires robust design methodologies to reduce the development cycle and prototyping phase. Thus, the design and optimization of physical layer communication is paramount. The complexity of these systems makes them difficult to optimize, because of the explosion in the number of unknown parameters. The methods and tools developed in past years will be eventually inadequate to address problems that lie ahead.

Communicating objects will be very often integrated into cluttered environments with all kinds of metal structures and dielectric larger or smaller sizes compared to the wavelength. The designer must anticipate the presence of such barriers in the propagation channel to establish properly link budgets and an optimal design of the communicating object. For example, the wave propagation in an airplane cabin from sensors or even an antenna, towards the cockpit is greatly affected by the presence of the metal structure of the seats inside the cabin or even the passengers. So, we must absolutely take into account this perturbation to predict correctly the power balance between the antenna and a possible receiver.

More generally, this topic will address the theoretical and computational electromagnetics in order to propose an implementation of informatics tools for the rigorous calculation of electromagnetic scattering inside very large structures or radiation antenna placed near oversized objects. This calculation involves the numerical solution of very large systems inaccessible by traditional resources. The solution will be based on grid computing and supercomputers.

Electromagnetic modeling of oversized structures by means of different numerical methods, using new resources (hardware and software) to realize yet more performant calculations, is the aim of this work.

The numerical modeling is based on a hybrid approach which combines Transmission-Line Matrix (TLM) and the mode matching methods. The former is applied to homogeneous volumes while the latter is used to describe complex planar structures.

In order to accelerate the simulation, a parallel implementation of the TLM algorithm in the context of distributed computing paradigm is proposed. The subdomain of the structure which is discretized upon TLM is divided into several parts called tasks, each one being computed in parallel by different processors. To achieve this, the tasks communicate between them during the simulation by a message passing library.

An extension of the modal approach to various modes has been developped by increasing the complexity of the planar structures.

The results prove the benefits of the combined grid computing and hybrid approach to solve electrically large structures, by matching the size of the problem with the number of computing resources used. The study highlights the role of parallelization scheme, cluster versus grid, with respect to the size of the problem and its repartition.

Moreover, a prediction model for the computing performances on grid, based on a hybrid approach that combines a historic-based prediction and an application profile-based prediction, has been developped. The predicted values are in good agreement with the measured values. The analysis of the simulation performances has allowed to extract practical rules for the estimation of the required resources for a given problem.

Using all these tools, the propagation of the electromagnetic field inside a complex oversized structure such an airplane cabin, has been performed on grid and also on a supercomputer. The advantages and disadvantages of the two environments are discussed.

Contents

Contents vii

List of Figures ix

List of Tables xiii

Abbreviations xv

1 Introduction 1
- 1.1 Background and motivation . . . 1
- 1.2 Objectifs and contributions . . . 1
- 1.3 Structure of the thesis . . . 2

2 Computational Electromagnetics 5
- 2.1 Numerical Modeling . . . 5
- 2.2 Computational Electromagnetics (CEM) . . . 6
- 2.3 Classification of Numerical Methods in CEM . . . 8
- 2.4 Analytical and Numerical Issues in CEM . . . 9
- 2.5 Transmission-Line Matrix vs Finite-Difference Time-Domain . . . 11
- 2.6 Conclusions . . . 11

3 Parallel Computing 13
- 3.1 Parallel and distributed architectures . . . 14
  - 3.1.1 Grid computing . . . 15
    - 3.1.1.1 Grid'5000 platform . . . 18
  - 3.1.2 Hyperion-Calmip . . . 22
- 3.2 Performance assessment on parallel architectures . . . 22
- 3.3 Application execution-time prediction . . . 23
- 3.4 Conclusions . . . 24

4 Transmission-Line Matrix Modal Hybrid Approach 27
- 4.1 Transmission-Line Matrix Method . . . 27
  - 4.1.1 Overview . . . 27

4.1.2 Huygens principle . . . 29
4.1.3 TLM basics . . . 31
4.1.4 Scattering matrix . . . 33
4.1.5 Stub loaded node . . . 34
4.1.6 Boundary Conditions . . . 34
4.1.7 Outputs . . . 35
4.1.8 TLM 3D - SCN . . . 35
4.1.8.1 Symmetrical Condensed Node (SCN) . . . 36
4.1.9 Error sources and corrections . . . 40
4.2 Modal Approach . . . 41
4.3 TLM/Modal Hybrid Approach . . . 42
4.3.1 State of the art . . . 42
4.3.2 TLM/Modal coupling matrix . . . 43
4.3.3 Generation of the analytical modal time-domain Green's functions . . . 47
4.3.4 TLM convolution product . . . 47
4.3.5 Numerical examples . . . 48
4.4 Conclusions . . . 57

5 Parallel TLM/Modal approach 61
5.1 Context . . . 61
5.2 Parallel TLM/modal hybrid approach . . . 62
5.2.1 Parallel algorithm . . . 62
5.2.2 Experiments . . . 63
5.3 Prediction model for an application execution time . . . 71
5.3.1 Prediction model based on least squares method . . . 71
5.3.1.1 Single precision vs double precision . . . 71
5.3.2 Application profile based prediction model . . . 76
5.3.2.1 "Four coefficients" model . . . 76
5.3.2.2 "Seven coefficients" model . . . 77
5.3.2.3 "Cache-misses" model . . . 78
5.4 Conclusions . . . 85

6 Supersized and Complex Structures 87
6.1 State of the art . . . 87
6.2 Modeling Complex Structures . . . 88
6.3 Modeling Supersized and Complex Structures . . . 93
6.4 Conclusions . . . 99

7 Conclusions and Perspectives 101

References 105

List of Figures

3.1 Grid'5000's architecture. 19
3.2 Grid'5000 access. 20
3.3 Drawgantt charts displaying past, current and scheduled OAR jobs. 21

4.1 Huygens principle [2]. 30
4.2 Wave propagation in a two-dimensional TLM network [2]. 31
4.3 Dispersion of the velocity of waves in a two-dimensional TLM network [64]. . 33
4.4 Boundaries of a two-dimensional TLM network [64]. 35
4.5 Symmetrical Condensed Node [63]. 36
4.6 Scattering matrix modeling homogeneous media. 37
4.7 Scattering matrix modeling non-homogeneous media. 39
4.8 Schematic view of the domain decomposition. 44
4.9 SCN nodes on reference plane. 45
4.10 TLM/modal connection network. 46
4.11 Hollow waveguide. 48
4.12 General view over the structure modeled upon the hybrid approach. 48
4.13 TE_{10} mode characteristic impedance. 49
4.14 Frequency-domain representation of the Scattering Coefficient Γ. 49
4.15 Frequency-domain representation of the Scattering Coefficient Γ prepared for IFFT. 50
4.16 Time-domain impulse response of the Scattering Coefficient Γ. 50
4.17 Results for the TE_{10} mode propagation in a matched waveguide. 51
4.18 Results for the TE_{10} mode propagation in a short-circuited waveguide. . . . 52
4.19 Input impedance (imaginary part) of a short-circuited lossless waveguide with dielectric $\epsilon_r = 2.54$. 53
4.20 Results for the TE_{10} mode propagation in an open-circuited waveguide. . . . 54
4.21 Hollow waveguide terminated by a metallic strip printed on a perfectly magnetic wall. 55
4.22 General view over the structure modeled upon the hybrid approach. 57

4.23 Input scattering matrix elements (absolute value) of a lossless waveguide, terminated by a metallic strip printed on a perfectly magnetic wall (1 denotes the TE_{10} mode while 2 denotes the TE_{30} mode); hybrid model computation (blue curve) vs. analytic computation (red dashed curve). 58
4.24 Input impedance matrix elements (imaginary part) of a lossless waveguide, terminated by a metallic strip printed on a perfectly magnetic wall (1 denotes the TE_{10} mode while 2 denotes the TE_{30} mode); hybrid model computation (blue curve) vs. analytic computation (red curve). 58

5.1 Schematic view of the modeling approach. 62
5.2 Schematic view of the parallel hybrid approach implementation. 63
5.3 Simulation times on cluster for different TLM cell numbers. 65
5.4 Parallel computation Speedup on shared memory multiprocessors - measured vs optimal values. 66
5.5 Simulation times on grid for different TLM cell numbers. 67
5.6 Simulation times on grid for different number of modes. 68
5.7 Parallel computation Speedup for a waveguide modelled with 145 million of TLM cells, obtained on grid computing and supercomputer. 70
5.8 Computation time and communication time at each time step during a parallelized simulation over 32 processes. 72
5.9 Simulation time when no computation is performed. 73
5.10 Computation time of the *process 0* increases once the excitation begins. . . . 74
5.11 Simulation time using the application with double-precision. 75
5.12 Parallel computation Speedup: single vs double prediction. 75
5.13 Cache misses evolution when increasing the number of TLM cells. 79
5.14 3D view of the structure (X, Y, Z - TLM cell number on the three cartesian directions). 79
5.15 Memory structure representation. 81
5.16 Computation time on grid for different TLM cell numbers: measured values versus predicted values. 83
5.17 Total simulation time on grid: measured values versus predicted values. . . . 83
5.18 Processes optimal number. 84
5.19 Computation time prediction for various simulations when TLM cells model dielectric media - the memory space required for storing *2*X*Y*18*8* bytes is smaller than the processor cache memory size. 84
5.20 Computation time prediction for various simulations when TLM cells model dielectric media - the memory space required for storing *2*X*Y*18*8* bytes is larger than the processor cache memory size. 85

6.1 Perspective view over a small airplane model with seats. 88
6.2 Longitudinal section of the small airplane model with seats. 89
6.3 Schematic view of the whole structure. 89
6.4 TLM seats model with $\epsilon_r = 4.3$ for a small airplane model. 89
6.5 Field propagation inside the small airplane structure at (a) excitation time, $t = 103$, and at time steps (b) $t = 350$ and (c) $t = 700$. 90
6.6 Sight over the field propagation inside the small airplane structure at time step $t = 700$. 90
6.7 Field propagation inside the small airplane structure modelled with $\tan\delta = 0.01$ of the dielectric, at (a) excitation time, $t = 103$, and at time steps (b) $t = 350$ and (c) $t = 700$. 91
6.8 Sight over the field propagation inside the structure modelled with $\tan\delta = 0.01$ of the dielectric, at time step $t = 700$. 91
6.9 A subdomain of the parallelized small airplane structure corresponding to the process 0. 92
6.10 A subdomain of the parallelized small airplane structure corresponding to the process 1. 92
6.11 TLM seats model with $\epsilon_r = 4.3$ for the airplane cabin model. 94
6.12 Schematic cross-sectional view of the cabin model. 94
6.13 Schematic view of the whole structure. 95
6.14 Parallel implementation of the aircraft cabin model with seats (longitudinal view). 95
6.15 Field propagation inside eight consecutive subdomains corresponding to the processes 30 through 37, at time steps: (a) $t = 1549$, (b) $t = 1716$, (c) $t = 1876$ and (d) $t = 29364$. 96

List of Tables

5.1 Summary of Computed Structures 63
5.2 Blocking vs Non-Blocking MPI Communications on Cluster 64
5.3 Speedup and Efficiency on Cluster 65
5.4 Speedup and Efficiency on Grid 67
5.5 Compressed vs Non-Compressed message on Cluster 69
5.6 Simulation time on Cluster: measured vs predicted values. 71
5.7 Simulation time on Cluster: single vs double precision 75
5.8 Computation time on grid: measured vs predicted values 77
5.9 Computation time on grid: measured vs predicted values. 77
5.10 Cache misses evolution when increasing the number of TLM cells. 78

6.1 Process 0 and Process 1 at different time steps during the simulation. 93
6.2 Summary of Simulation Time on Grid: measured and predicted values 97

Abbreviations

ABC	**A**bsorbing **B**oundary **C**ondition
CEM	**C**omputational **E**lectromagnetics
CPU	**C**entral **P**rocessing **U**nit
DE	**D**ifferential **E**quation
EMC	**E**lectromagnetic **C**ompatibility
EMI	**E**lectromagnetic **I**nterference
FDTD	**F**inite-**D**ifference **T**ime-**D**omain
GC	**G**rid **C**omputing
HFSS	**H**igh **F**requency **S**tructural **S**imulator
HPC	**H**igh **P**erformance **C**omputing
IE	**I**ntegral **E**quation
IFT	**I**nverse **F**ourier **T**ransform
MCN	**M**odal **C**onnection **N**etwork
MPI	**M**essage-**P**assing **I**nterface
MPP	**M**assively **P**arallel **P**rocessing
NFS	**N**etwork **F**ile System
PED	**P**ortable **E**lectronic **D**evice
PVM	**P**arallel **V**irtual **M**achine
SCN	**S**ymmetrical **C**ondensed **N**ode
SCT	**S**cale-**C**hanging **T**echnique
SIMD	**S**ingle **I**nstruction **M**ultiple **D**ata
SPMD	**S**ingle **P**rogram **M**ultiple **D**ata
TLM	**T**ransmission-**L**ine **M**atrix
WCET	**W**orst-**C**ase **E**xecution **T**ime
WLAN	**W**ireless **L**ocal **A**rea **N**etwork

Chapter 1

Introduction

1.1 Background and motivation

Nowadays, the advances in wireless and microwave communication systems have drastically increased the development of portable electronic devices (PEDs), such as cellular phones, laptops, etc. This implicitly increased the necessity of their usage more and more, in different environments, existing the risk of the electromagnetic interference (EMI) phenomena with other crucial communication systems.

In this scenario are also the PEDs inside the airplane. Their operation on board is still considered a potential EMI source for the communication and navigation airplane systems. However, the aircraft manufacturers are interested to offer more services to their passengers. Installing a wireless network on board, would permit to passengers to use their new portable devices over the internet. Also, a wireless network would replace the existing on-board cables reducing the weight of the aircraft and gaining space on board. Many research works related to the potential risk of wireless networks inside the airplane are carried out.

An electromagnetic simulation model is much cheaper and flexible than an experimental approach, but needs fast and accurate numerical simulation techniques. An airplane cabin is a complex environment for the electromagnetic field propagation prediction. Most of the full-wave computational electromagnetics techniques require discretization of the entire computational domain. When dealing with a complex and large structure as the airplane cabin, the computational resources of the most powerful computers are limited.

The limits imposed by the traditional computation architecture are the memory and time constraints.

1.2 Objectifs and contributions

The main objectif of this thesis is to give a solution to the rigorous calculation of the electromagnetic scattering inside very large and complex structures such as an airplane

cabin, process involving the numerical solution of a large number of unknowns unaffordable by traditional resources.

To achieve this, a hybrid modeling approach is implemented, since a conventional full-wave electromagnetic simulation would require enormous amount of computational resources. This numerical tool uses a rigorous method - Transmission-Line Matrix (TLM) - and a modal approach. The former is applied to homogeneous volumes while the latter is used to describe complex planar structures as a multiport surface impedance, according to diakotics procedure. Also, a multimodal approach has been developped in order to increase the complexity of the planar structures.

This hybrid approach saves large computational resources, because a simulation using a rigorous method for the whole structure would be crucial, due to the large electrical size of the planar structures. This numerical hybrid tool is validated by analytical curves and also by HFSS results.

The second very important research contribution is the implementation of the parallel TLM algorithm in order to take full advantage of the parallel and distributed architectures, as the cluster, computational grid and supercomputer. The TLM/modal hybrid tool and the TLM parallel approach permit to realize fast and full-wave simulations of large and complex structures that are not possible even with the most powerful personal computers.

Thus, subdomains of the structure discretized upon TLM are divided into several parts each one being computed in parallel by different processors that exchange between them messages to fulfill the job, based on message passing libraries. The experiments show the benefits of the parallelization scheme on cluster, grid and supercomputer with respect to the size of the problem and its repartition.

Also, another contribution is represented by the development of a time prediction model, based on the profile of the application, that allows a safe reservation of the computational resources on grid environment. Rules for the estimation of the required resources to compute a given structure with a certain efficiency have been shown.

Using all these results, the electromagnetic computation of an airplane cabin with seats has been possible. In brief, this research work deals with the electromagnetic computation of large and complex structures, such as an airplane cabin, by means of a parallel TLM/modal hybrid numerical approach efficiently deployed on parallel computing systems.

1.3 Structure of the thesis

This thesis is organized as follows. In Chapter 2, a synoptic overview of the numerical modeling applied in electromagnetism is presented, focusing on the main analytical and numerical issues in computational electromagnetics.

Chapter 3 gives a short presentation of the parallel and distributed computing systems

and of the computing platforms used to launch experiments during this research work. This chapter highlights grid technologies and their multidisciplinary nature providing an extremely powerful tool for scientists motivating them to migrate their applications on a such large and complex scale system. Also, the historic and profile based time prediction model for the computation time required by the experiments on grid, is given.

Chapter 4 describes the theory behind the TLM method and the possible error sources and corrections. Symmetrical condensed node structure with or without derivations is presented, focusing on the scattering matrix. Also, a short presentation on Mode Matching method is given.

The time-domain hybrid based on computing domain decomposition according to diakotic procedure is outlined in this chapter. The volume subdomains are discretized upon TLM and the planar structures subdomains are modelled by modal approach. The two subdomains are joined together by relating the tangential electromagnetic fields at the interface between the two subdomains, based on an integral equation (IE) formulation: the field on the volume subdomain surface to the active modes of the planar subdomains.

Chapter 5 presents the parallel approach which combines the numerical hybrid method based on TLM and modal approach, implemented in the previous chapter, with computational grid and supercomputer, to run fast and full-wave electromagnetic simulations of large and complex structures. This chapter describes numerous experiments performed on cluster, grid and supercomputer and focuses on the performance gained by the parallel computation, with respect to the size of the problem and its repartition.

Also, some aspects related to the application implementation and computing platform which directly influence the computing performance, such as data precision, communication modes between processes, memory contention, cache misses are outlined. Chapter 5 presents also tools for the estimation of the simulation time and for the required number of resources, that have been developed based on the TLM algorithm to efficiently use the computational grid.

In Chapter 6, the TLM/modal hybrid approach described in Chapter 4 and the parallel approach and prediction model given in Chapter 5 are used to simulate, on grid and supercomputer, the propagation of the electromagnetic field inside a real complex and supersized structure - an airplane cabin with seats.

Chapter 2

Computational Electromagnetics

In this chapter, a short introduction to numerical modeling and computational electromagnetics (CEM) is outlined. A not exhaustive classification of numerical methods in CEM is exposed with a brief overview of the main analytical and numerical issues. Finally, the chapter is concluded comparing two well-known differential equations methods, as Transmission-Line Matrix and Finite-Difference Time-Domain.

2.1 Numerical Modeling

Numerical modeling is a very useful tool for the computation of the physical phenomena by means of computers. A model helps us to calculate and test different scenarios mathematically, in order to predict what will happen in a given situation in reality. Modeling is a process aimed for the developping of a simplified mathematical model of a complex reality.

The challenge in modeling is to develop an understanding of the complex physical reality as a conceptual model [61]. Then, the defined concept is put in a mathematical form in order to get a numerical result and to understand the risks of the model.

Modeling is a process that can uses numerical methods to explain more complex things. According to the nature of phenomena to describe, the model can have different complexity degrees. So, the computational capacity of a machine to deal with the corresponding model can be quickly reached. At this level, the informatic resources play an important role in perfecting the process of modeling.

A modeling methodology applicable in a very large area can be summarized in the following steps, [77]:

- conceptualization means the observation and the analysis of the physical principles;
- formulation - physical principles are put in a mathematical form;
- numerical implementation - the matematical form is put in an algorithm form for a digital computer;

- computation - coding the algorithm in a computer programming language;

- validation - the results are checked for numerical and physical credibility.

In more detail, conceptualize means defining the purpose of the model, collecting the existing information in the corresponding domain and defining a conceptual model of the phenomena in study; in initial phase, the model should be as simple as possible; the complexity can be increased step by step, as the model is validated. In the formulation step, starting from the concept and using mathematical tools, the model is given in a mathematical form based on fundamental theory.

The main purpose of the numerical implementation of the model is determining a solution. In order to obtain an analytical solution, the mathematical formulation established the step before is applied to a geometrical domain defined in the context of a set of boundary conditions and a collection of material properties. For a numerical solution, the model has to be discretized and implemented using a programming language interpretable by a digital computer. The analytical solution and the numerical one have to be verified with other sources from the literature or measures.

Analysing the numerical results, there are a lot of problems which can alterate the solution: roundoff, convergence, numerical oscillation, numerical dispersion, space and time discretization. Finally, the results are checked against the physical reality.

Simple models used together with computer modeling tools enhance the engineering experience to understanding complex phenomena. In order to realize all these steps, in numerical modeling are required many skills as physicist and engineer skills, mathematical skills, numerical analysis skills, computer scientis skills for the computation and also good interpretation skills for the results.

2.2 Computational Electromagnetics (CEM)

Computational Electromagnetics is a branch of electromagnetism which is based on computing resources to obtain numerical solutions to Maxwell's equations for a specific electromagnetic problem.

Electromagnetism is a discipline which deals with electric and magnetic sources and the fields that they produce in different environments. The analytical model of electromagnetic phenomena is presented by James Clerk Maxwell (1865) ("A dynamical Theory of the Electromganetic Field") in the four equations that bear his name. The general form of

time-varying Maxwell's equations can be written in differential form as:

$$\nabla \times \overrightarrow{\mathcal{E}} = \frac{-\partial \overrightarrow{\mathcal{B}}}{\partial t} - \overrightarrow{\mathcal{M}}, \tag{2.1a}$$

$$\nabla \times \overrightarrow{\mathcal{H}} = \frac{\partial \overrightarrow{\mathcal{D}}}{\partial t} - \overrightarrow{\mathcal{J}}, \tag{2.1b}$$

$$\nabla \cdot \overrightarrow{\mathcal{D}} = \rho, \tag{2.1c}$$

$$\nabla \cdot \overrightarrow{\mathcal{B}} = 0, \tag{2.1d}$$

where: $\overrightarrow{\mathcal{E}}$ is the electric field intensity, V/m;
$\overrightarrow{\mathcal{H}}$ is the magnetic field intensity, A/m;
$\overrightarrow{\mathcal{D}}$ is the electric flux density, C/m^2;
$\overrightarrow{\mathcal{B}}$ is the magnetic flux density, Wb/m^2;
$\overrightarrow{\mathcal{M}}$ is the magnetic current density, V/m^2;
$\overrightarrow{\mathcal{J}}$ is the electric current density, A/m^2;
ρ is the electric charge density, C/m^3.

The script quantities presented above represent time-varying vector fields and are real functions of spatial coordinates *x,y,z* and the time variable *t*. Any electromagnetic problem can be solved using Maxwell's equations. Analytical solutions to Maxwell's equations can be found only for a small number of problems.

Electromagnetic problems today are complex and it is impossible to solve them by finding analytical solutions to Maxwell's equations. Numerical methods in electromagnetism based on approximations, discrete the analytical model of the problem to be solved, thus creating a model that can be calculated by digital computer.

From a systemic perspective [77], the electromagnetic problem can be treated following the model: cause - effect. The problem to solve is transposed into mathematical form using Maxwell's equations, considering the boundary conditions and the materials. Thus, the transfer function or field propagator is built. The excitation of the structure to be modeled is the system input. The output, represented by the result provided by the model is obtained considering the excitation and the field propagator.

A digital model is a tool increasingly used by engineers, complementary with the measurements and the analytical approach, that meet certain requirements of accuracy, efficiency and applicability, [52].

The accuracy of the model is represented by the distance between the real values and the estimated values by the model. The accuracy of the estimated values is given by the nature of the application where the model is used. Model accuracy is proven by comparing estimated values with values obtained through analytical approach or from measurements.

The efficiency of the model is related to the number of computing resources required to perform an estimation and the user effort consumed in using the model. The efficiency depends on how the model is implemented in the computing machine, and on the application that uses it.

The applicability of the model represents the diversity of the applications that the model can be used efficiently and accurately, depending on the size and materials of the simulated structure, the excitation and the required results, the frequency domain of interest.

2.3 Classification of Numerical Methods in CEM

The numerical modeling techniques used in electromagnetism can be classified according to various criteria. According to the equations underlying the field propagator, there are methods based on Maxwell's curl equations, Green's function, modal (spectral) expansions and optical description.

The numerical methods that build models based on global IE operators as Method of Moments [47] or Integral Equation Technique [55], define the field propagator using the Green's function which is closest to the analyzed problem. By analytical manipulation, Green's functions incorporate specific boundary conditions. The model solution is obtained using Method of Moments by discretizing the analytical form in a matrix. The matrices can be dense depending on the size of the problem. The effort required to calculate the matrix depends on the size of the problem.

The methods based on local differential equation (DE) operators as Finite-Difference Time-Domain (FDTD) [115] or TLM [28] define a field propagator with Maxwell's curl equations. The problem to be analyzed is discretized by all sides to highlight the spatial variation of the electromagnetic field. The analytic manipulation of the model is reduced. The DE model is suitable for modeling issues that are nonlinear, time variating properties, non-homogeneous materials. The disadvantages of this approach are: the entire space of the problem is discretized into points which implies a large computing effort, the difficulty of modeling curved objects, the dispersion errors, the difficulty of modeling the radiation problems by open field boundary conditions. The IE model has the radiation condition incorporated and there is no need for the simulation of open field boundary conditions.

The model based on modal expansion as Mode-Matching Method [55] or Scale-Changing Technique (SCT) [14] is used to formulate boundary-value problems, the discontinuities located inside a structure. The two spaces, located on both sides of the junction, have solutions defined by Maxwell's equations for specific boundary conditions which do not include the junction. The junction plane fields in the two regions are decomposed into an infinite number of modes, which involve infinite set of linear equations for unknown coefficients of the modes considered. The limitation of the number of modes has to be done

carefully to preserve the convergence problem.

Methods based on optical description are asymptotic methods for the solution of the Maxwell equations. These models are built by means of ray tracing and diffraction coefficients. The ray tracing consists in drawing all the propagation paths between a fixed source and a receiver. The signal reaching a receiver is superposed from a finite number of different propagation paths, that can be determined independently. In Geometrical Theory of Diffraction (GTD) [67, 68], the considered wave propagation phenomena are the incident direct illumination, reflection, diffraction by edges or tips, and surface diffraction waves also known as creeping waves. These phenomena are called effects and together form the ray tracer.

These models have the ability to predict the electromagnetic field asymptotically in the limit of vanishing wavelength with a low computational cost because the ray tracing depends on the geometric aspects of the structure and not on its electrical size. The disadvantages of this method are: the low accuracy of the calculated field the theory will only yield the leading terms in the asymptotic high frequency solution of the Maxwell's equations, the inability to model small details on board of large structures, the non-physicall zero field values inside the transition regions between the illuminated regions and shadow zones.

2.4 Analytical and Numerical Issues in CEM

The main analytical nature problems which arise in the development of a model are: the selection of the model solution (time or frequency) and the selection of the field propagator.

The numerical nature problems are the approximations. The quantities are often approximated in terms of polynomials sampling functions, that are then substituted for these quantities in various analytical operations. Thus, integral operators are replaced by finite sums and differential operators by finite differences.

In [77], Miller presents the computer-time dependence on the number of unknowns included in the model. For the two different numerical modeling approaches: integral equation modeling and differential equation modeling, it is proved the increasing of the computer time with the increasing of the problem size. These computer-time estimates are realized assuming that solutions are obtained by method of moments.

Depending on the requirements of the application, a time-domain model can be used to obtain a spectral response from one single computation, while for a frequency-domain model the computation has to be launched for every required frequency.

Considering the application, the main objective of a model is to obtain the results as quick as possible, with the required accuracy. There are different ways of reducing the computation time. Concerning the analytical part of the model, the possibilities to optimize the computation time are: using specialiazed Green's functions which satisfy additional boundary con-

ditions in case of the integral equation model, using different numerical methods in order to exploit their complementary advantages to create a better model in a hybrid approach, using other methods for simplifying the formulation and the computation as impedance boundary condition (IBC) [106, 107], physical optics (PO) approximation presented in Chapter 4 of [133], reflection-coefficient approximation (RCA) [20].

There are a number of numerical aspects which can lead to the reduction of the cost for a moment-method model such as: iterative technique to solve direct matrix, exploiting problem symmetries, near-neighbor approximations, adaptive modeling, model-based parameter estimation, numerical Green's function.

Solving the direct matrix by direct solution techniques is proportional to the number of spatial samples, and for oversized structures the computation becomes heavy. If an RHS-dependent solution is acceptable, then iterative techniques which have a lower order dependence on the number of spatial samples can be used, such as: Jacobi, Gauss-Seidel, Gauss-Seidel with simultaneous over relaxation, conjugate-gradient technique (CGT) [104, 109].

The reduction of the cost in terms of time and storage during the computation of a model can be achieved also by exploiting the application symmetries. There are three types of symmetries: by reflection, rotation and translation. Applying the symmetries during the computation creates repetitive elements in the solution matrix, which means a reduction of the number of operations during the computation.

Near-neighbor approximations (NNA), outlined in the first chapter of [95], is based on ignoring small interactions due to the geometric attenuation of the fields with increasing distance from the source. Concerning the modal approach, the coupling between the modes can be ignored as the difference between their mode numbers increases [74]. Adaptive modeling variates the sampling to the requirements of a given problem. The static adaptation establishes for different problem classes different sampling methods which will be used depending on the problem to deal with. The dynamic adaptation adjusts the sampling during the computation, regarding also the accuracy requirements.

Validation:

The process of constructing a model starting from a physical problem implies also a lot of approximations which lead to important uncertainties. Modeling uncertainties can be assigned to two basic error categories: a physical modeling error and a numerical modeling error. The physical error is due to the mathematical representation of the physical reality. The numerical error is due to the finite number of unknowns able to treat and to the roundoff caused by the finite precision computation or the limited solution convergence for the iterative methods. Validation step in the modeling process checks the obtained results. The results are compared with other external sources such as analytical, experimental or numerical.

2.5 Transmission-Line Matrix vs Finite-Difference Time-Domain

The method adopted for this work is TLM, which can be classed as a differential time domain method; although a frequency domain formulation has been proposed. TLM has a number of features which make it ideally suited to EMC problems. The method, based on a physical model described by Huygen's principle, is expressed in terms of circuit concepts which are familiar to the engineer. In the absence of any active components, stability problems do not arise. Increased resolution can be applied only in areas where it is required. A single calculation will give information over a wide range of frequencies. Both the internal and external environments can be modelled simultaneously. Complex shapes can be included. Inhomogeneous materials can be described. Source regions with non-linearities can be represented. There are several features which make it less suitable, although these are areas of current research. There are problems in representing features which are smaller than a node, e.g. wire-like structures. Completely general open boundaries, or radiation boundaries, have yet to be implemented.

Yee's Finite-Difference Time-Domain is formulated by a mathematical model based on Maxwell's equations. Chen et al. [27] present a new FDTD equivalent with the symmetrical condensed node scheme in TLM, avoiding the displacement between the electric field and the magnetic field existent in the Yee's version. Thus, the TLM algorithm can be implemented in the FDTD form or vice versa.

In the Johns's view [62], "the TLM method and the finite difference method complement each other rather than compete with each other. Each leads to a better understanding of the other."

Concerning the computational expenditure, TLM compares favorably with FDTD. Its accuracy is even slightly better by virtue of the Fourier transform, which ensures that the field function between nodes is automatically circular rather than linear as in FDTD. In Radar Cross-Section (RCS) calculations and antenna modelling [138] TLM needs more computation time and memory, but seems less sensitive to numerical dispersion.

The large amount of information obtained by one computation with TLM, as the impulse response of the structure, the characteristics of the propagating modes accessible in frequency domain by Fourier transform, is another advantage.

2.6 Conclusions

In this chapter, a short presentation of the numerical modeling applied in electromagnetism is exposed. TLM and FDTD numerical methods are compared and have a lot of similarities. But, the main advantage of the TLM is the great flexibility and versatility, incorporating

all the properties of the electromagnetic field and his interaction with the materials and the boundaries. Thus, it is not more necessary to reformulate the electromagnetic problem for different structures, but only changing the parameters is required. Based on a physical model, TLM provides new insights on the behaviour of waves, having a great range of fields where can be applied as electromagnetism, thermodynamics, optics and acoustics. TLM is the method chosen to work with in this thesis.

Chapter 3

Parallel Computing

One of the biggest problem that science faces today is the huge amount of data to be processed very often in an entangled interdisciplinary context. Mathematics, physics, medecine, informatics or the other areas derived from them, are all dependent on usage of numerical resources. Therefore, the parallel computing plays a crucial role in this scenario, even though could be quite expensive depending on the parallel architecture, as supercomputers, massively parallel processors (MPP).

The huge spread of computers and the increasing development of the Internet and web tools favored the emergence of distributed-memory architectures, as clusters, grids, with a substantial costs reduction. Yet the grid offers much more than that. Grid Computing (GC) makes possible coupling and using as single unifying resource, a wide variety of resources distributed world wide as supercomputers, local area networks (LANs), storage systems, different devices, in several interesting scenarios: collaborative engineering, data exploration, high-throughput computing (HTC), meta application and high-performance computing (HPC). Most of the scientific simulations in weather forecasting, astrophysics, financial modeling or communication research are HPC applications and require large amounts of computational power that even supercomputers cannot fulfill.

In this chapter, a synoptic overview is given on parallel and distributed computing. Large scale parallel systems, parallel and distributed architectures, parallel and distributed programming approaches and parallel computing performance assessment indicators are outlined. Execution time prediction methods for masivelly parallel computation applications performed on distributed computing systems such as clusters and grids, are summarized. A hybrid approach time prediction model for the computation time on the parallel TLM application, based on grid historic experiments and on application profile, is proposed.

Grid'5000, a research infrastructure for large scale parallel and distributed computing, and Hyperion supercomputer, the two platforms used for the research experiments performed in this work, are presented.

3.1 Parallel and distributed architectures

According to Flynn's taxonomy [37], there are four main computing architectures: Single Instruction Single Data (SISD), Single Instruction Multiple Data (SIMD), Multiple Instruction Single Data (MISD) and Multiple Instruction Multiple Data (MIMD). The parallelism is introduced by the last three architectures. Today, most computers are based on MIMD, including the Symmetric Multiprocessor (SMP), the Massively Parallel Processor (MPP), the Distributed Shared Memory (DSM) and the Cluster of Workstation (COW).

The development of parallel computing machines starts in 1960s with ILLIAC IV, a pioneer in massively parallel computing and continues with Parallel Vector Processor (PVP) machines as Cray YMP-90, NEC SX-3, Fujitsu VP-2000 in 1970s, shared memory systems as SGI Challenge and Sun Sparc Center 2000, MPP machines as Intel Paragon, CM-5E, Cray T3D, IBM SP2 in 1980s.

Even if the recent technological developments led to the creation of more powerful computing machines, they still fail to provide enough power in order to perform calculations of high complexity or dealing with too many data. The parallel or distributed computing is an answer to this issue. The idea is to start the computation on a set of machines, linked together by fast networks to increase the performance.

So, the concept of cluster computing has emerged in 1990s, while the number of personal computers and the availability of networking equipement have increased due to the lower prices. The cluster groups several computers together, called as computing nodes, and connect them in order to share their resources and to enable a global management. A cluster is generally used for high performance computing, being composed of homogeneous machines, in terms of architecture and operating system, geographically close and linked by very fast networks, with throughput up to 10 Gb / sec [137].

So, the computation effort is reduced by spliting the problem to be solved into several subproblems that will be computed in parallel on multiple CPUs that communicate between them to complete the entire job. The communication between the CPUs can be realized by two different techniques: message passing and shared memory. Depending on the architecture, the application has to be parallelized using one of the two programming paradigms. Architectures with shared memory allow communication between CPUs by variables stored in a shared space memory, while the architectures with distributed memory make possible the connection between CPUs by a communication network.

For the first type of architecture an important issue is to ensure the integrity of shared data. In the case of distributed architectures, one difficulty is given by the distribution of the calculation task on multiple processors with individual memory and the reconstruction of the final solution based on the results obtained by each CPU. A parallel programming paradigm that fits these characteristics is *Message Passing* - the interaction between processes is achieved by an exchange of messages. For *Shared memory* systems, all processes

access data from shared memory and data can be transferred between processes at any time. In general, the shared-memory systems have a poor scalability because of the memory bus saturation and the memory access time slowing down. The application to deal with determines the choice of a certain architecture and the programming method.

According to the parallel programming technique, there are different standards: Parallel Virtual Machine (PVM) [31] and Message-Passing Interface (MPI) [79] for message passing and OpenMP [88] for shared memory. According to the architectures used to launch experiments in this research work, the parallel TLM application is based on MPI. MPI is a standard programming based on the exchange of messages, in order to exploit remote computers or multiprocessors in heavy calculations by exchange of messages. Several MPI implementations can be mentioned: MPICH2 [80], OpenMPI [89]. The parallel TLM application is based on OpenMPI v1.4, under Squeeze-x64-base-1.1, a Debian Linux distribution (Kernel version 2.6.32-5-amd64).

Since the long-distance broadband networks have been implemented, including the Internet, the explosion of the computing power on low-cost personal computers, allowed the emergence of a new concept: the interconnection of clusters leading to computational grids.

3.1.1 Grid computing

A computational grid is a hardware and software infrastructure that provides dependable, consistent, pervasive, and inexpensive access to high-end computational capabilities [38]. The purpose is to coagulate resources from various organizations wishing to collaborate in order to provide to users computing capacities and storage that a single machine cannot provide. However, any distributed computing system cannot possess the name of grid computing. Indeed, a grid is a system that coordinates resources that are not submitted to a centralized control, using standard protocols and interfaces in order to deliver a certain quality of service.

There is a strong analogy between the development of computational grids and the power grids, at the beginning of the twentieth century. At the time, the revolution did not resided in the electricity itself, but rather in the design of a network providing to people a reliable access to electricity with a low cost, through a standard interface, as a socket. The components forming the electrical network are heterogeneous, and the induced complexity is completely masked to the end user. Also, GC possesses the same properties of heterogeneity of resources and transparency to the end user.

GC concentrates computing and storage resources, geographically distributed, to enable transparently their use for any grid client. The main features of a grid computing and its resources, are [116]:

1. **Heterogeneity.** Grid resources are heterogeneous: networks, platforms, operating systems, electronic devices and software tools from different providers and following

different architectures and paradigms are merged together, in a grid. This diversity has to be hidden to the end user, in order to provide a transparent access to all these resources. This involves the implementation of standardized communication protocols and imposes constraints on the code portability.

2. **Sharing resources.** Different organizations can share reciprocally their resources in order to better allocate the workload and to exploit the less used resources, promoting the efficiency and the reduction of operating costs.

3. **Scalability.** The number of resources that constitute a grid can vary from a dozen to several thousand. Grid performance must not be affected by this.

4. **Dynamism.** The grid should be seen as a single virtual machine. The complexity of the platform has to be masked to the user. So, the grid must manage possible resource status changes, and continuously adapt to the dynamic environment that constitutes it. The access to resources must be guaranteed to the user at any moment.

5. **Security.** Users must be recognizable and the access to resources must be monitored and controlled.

6. **Fault tolerance.** Due to the complexity of the grid environment, given by the heterogeneity of its resources, the failures are very probably. Robustness with respect to failure of network connections, machines, devices and software components, is a critical issue.

7. **Autonomy.** Each organization sharing resources on a grid, can implement different management and security policies, in terms of access to the network, authentication or confidentiality. Resources must nevertheless be accessible to all the grid users.

In addition, grid computing possesses the following objectives: provide an important capability of parallel computing, manage applications with close deadline, better distribute the use of resources, exploit less used resources, access to additional resources, provide fault tolerance for a lower cost.

Several grid platforms that are used in real-life applications can be mentioned. TeraGrid [117] is a project that operated for six years and finished in 2011. TeraGrid was an open scientific discovery infrastructure coordinated by the University of Chicago, including more than 1 petaflop of computing capability and more than 30 petabytes (quadrillions of bytes) of online and archival data storage with rapid access and retrieval over high-performance networks. Through the TeraGrid, researchers could access over 100 discipline-specific databases. Thus, TeraGrid was ranked the world's largest, most comprehensive distributed cyberinfrastructure for open scientific research. Today, it is replaced and expanded by The Extreme

Science and Engineering Digital Environment (XSEDE) [132], a single virtual system that scientists can use to interactively share computing resources, data, and expertise.

National e-Science Centre, developed by several departments from the Universities of Edinburgh and Glasgow, and the London e-Science Centre are supporting the development of e-Science project [34]. The European distributed computing infrastructure built by European Union in projects as DataGrid (2002-2004), EGEE-I, -II and -III (2004-2010) [32], is now supported by the European Grid Infrastructure (EGI) [33], project whose main purpose is to coagulate the European research community.

Grid3 infrastructure [13] has been operating since November 2003 with 27 sites, a peak of 2800 processors, workloads from 10 different applications exceeding 1300 simultaneous jobs, and data transfers among sites of greater than 2 TB/day. Grid3 has sustained for several months the production-level services required by physics experiments of the Large Hadron Collider at CERN (ATLAS and CMS), the Sloan Digital Sky Survey project, the gravitational wave search experiment LIGO, the BTeV experiment at Fermilab, as well as applications in molecular structure analysis and genome analysis, and computer science research projects in such areas as job and data scheduling.

Distributed European Infrastructure for Supercomputing Applications (DEISA) project [30] has operated for seven years and ended in April 2011. Now, it is continued by Partnership for Advanced Computing in Europe (PRACE) [93]. Also, National Research Grid Initiative (NAREGI) [82] provides scietists easy access to distributed computing resources. PlanetLab [92] is a global research network which provides since 2003, support for the development of new technologies for distributed storage, network mapping, peer-to-peer systems, distributed hash tables, and query processing.

Some examples of scientific applications using grids are given. GriPPS [124] is an application that serves the community of biologists to search for protein signature, realized at the Institute of Biology and Chemistry of Proteins, CNRS. This research is based on multiple access to data banks of huge volumes. In computational electromagnetics, the numerical resolution of systems of partial differential equations are performed on grid [26]. Another project, using a P2P grid is seti@home (Search for Extra Terrestrial Intelligence), for analyzing radio telescope data [108].

The computational grid joins together the web technologies and the distributed computing by means of a complex standardization effort. So, the power of distributed resources is opened to a large number of end users. The grid architecture is based on three layers:

1. **fabric level** - all the resources composing the grid, such as hardware (IT devices), software (applications, databases) and logical (clusters, distributed pools);

2. **middleware level** - grid software which makes possible to the end user, the easy usage of all the ressources;

3. **application level** - tools which permit the interaction between the end user and the grid resources.

A grid middleware is a software collection providing basic services necessary for applications to transparently interface with local systems and execute their processes on the nodes of the grid. The grid middleware is responsible for: resources allocation, computational economy, information management, security and data management.

Several grid middleware tools has been developped in order to fullfill the needs to create a grid, such as Legion [71], Unicore [122], BOINC [18], Vishwa P2P middleware [126], etc.. But, Globus Toolkit [43] still remains the grid middleware *de facto* standard for grids implementation. Created by Ian Foster and Carl Kesselman, widely considered as the "fathers of the grid" [39], Globus Toolkit is a project of several academic organizations such as, the University of Southern California, the Argonne National Lab and the University of Chicago, providing an open-source set of services addressing fundamental grid issues, such as security, information discovery, resource management, data management and communication. Globus toolkit has three major functions: to allocate grid resources to the consumer, to provide information about the available resources and to deal with the data access and management, all these based on Globus Security Infrastructure which ensures the authentication, confidentiality and integrity services. GRACE [22] is an economy-driven resource management tool that can coexists with the middleware systems, providing computational economy.

3.1.1.1 Grid'5000 platform

The research French project, Grid'5000 [25] provides a large scale and highly reconfigurable computational grid, as an experimental testbed for researchers. Seventeen laboratories and research units are involved in this project. Machines constituting the grid are distributed over ten sites in France, as: Bordeaux, Grenoble, Lille, Lyon, Nancy, Orsay, Reims, Rennes, Sophia Antipolis and Toulouse.

The main objective of Grid'5000 is to provide an experimental platform for all national research projects related to the field of GC. Thus, the latest innovations in terms of middleware, massively parallel applications or even parallel programming techniques can be tested on a real platform.

So, the grid implementation must therefore be fully configurable and controllable by users: the computing nodes can execute fully customized operating systems and environments and can even be restarted remotely. Tools for computing nodes reservation and monitoring are also available.

Grid'5000's resources can be represented by a hierarchical structure, as shown in Fig. 3.1. Thus, Grid'5000 platform is composed by multiple sites. Each site has one or more clusters; each cluster merges machines having the same physical characteristics, in terms of CPU, memory and storage capacity. Each machine can contain multiple processors (CPUs)

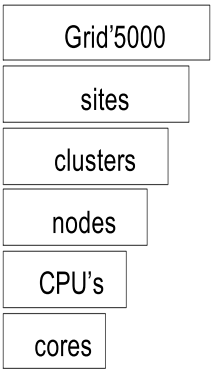


Figure 3.1: Grid'5000's architecture.

and each processor consists of several cores, the core being the smallest indivisible available unit of computation.

The computing nodes, that populate the sites, are mainly IBM eServer, IBM System, HP ProLiant, Sun Fire, Dell PowerEdge, Dell R720, Carri System, Altix or AppleXserve. They are generally equiped with at least two processors from two major families of processors: AMD Opteron and Intel Xeon. The different Grid'5000 sites are interconnected by the RENATER network, the National Network of Telecommunications for Technology, Education and Research [96]. Links connecting them support the flow rates of 10 Gb / sec. Some sites are equipped with a multi-gigabit internal network such as Myrinet 2G, Myrinet 10G or Infiniband; all the others are limited to gigabit Ethernet. For security reasons, the computing nodes are confined in a completely closed network and non-routable (private IP addressing), with no access to the Internet network. Only some specific machines are authorized to receive external connections.

In order to provide access to the grid resources, certain conventions have been agreed, related to the manner of operation of each site, to the machines name, or even the specific role of certain machines. The goal is to standardize the access to grid resources, and therefore hide the heterogeneity present between different sites.

The following machines are present on each site Grid'5000:

1. *Access machine* - Its role is to give access to the grid from the outside. Due to the confinement of the grid, it is the only possibility to access the grid from the outside. This machine allows only incoming connections authenticated and secured via ssh (secure shell, tcp port 22). No outgoing connection is possible, ensuring the tightness of the grid to the outside, as in Fig. 3.2 published on [44].

2. *Front-end machine* - This machine allows access between sites via the internal private network to connect from a site to another. This is not possible with the machine Access, the latter accepts only external connections. In addition, the Front-end machine provides tools for computing nodes reservation.

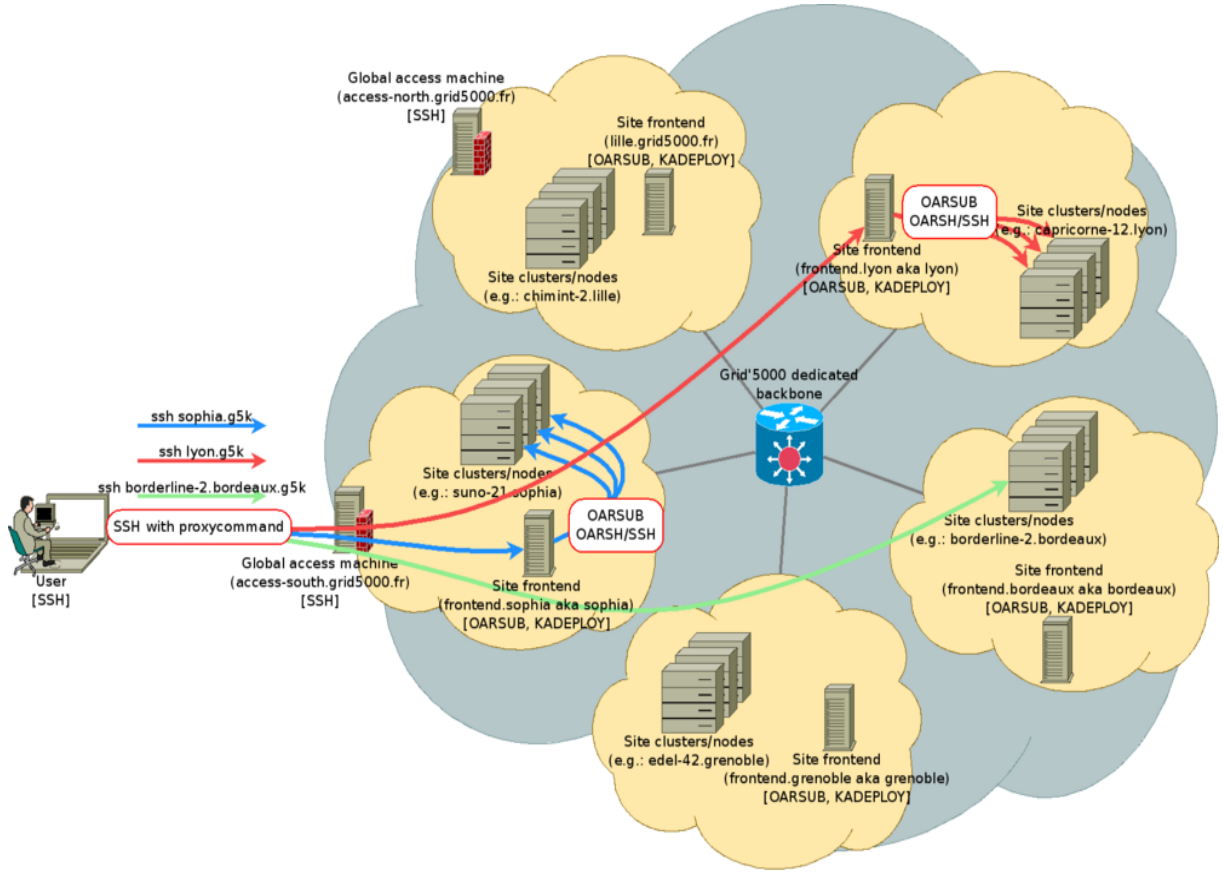


Figure 3.2: Grid'5000 access.

3. *Sync machine* - It is responsible for the data synchronization between different sites. Indeed, there is no automatic synchronization between sites where the local storage space is shared via Network File System (NFS). To synchronize the data on all the sites of the grid, Sync machines have to be used.

4. *Node machine* - These are the computing nodes, grouped into homogeneous clusters, performing the user applications. The nodes can be reserved by means of tools provided by the Front-end. These machines are fully configurable and controllable by the users.

Job scheduling and resource allocation and deallocation are performed by OAR [24], an open source batch scheduler at the cluster level. OARGRID is a batch scheduler at grid level, based on OAR, providing to users the possibility to allocate distributed resources. Graphical monitoring tools for the reservations and submissions are also available [85], such as: Monika and DrawOARGantt, as in Fig. 3.3. Ganglia [40] provides resources usage metrics. KADEPLOY [41] is a tool which lets the users to deploy their own operating systems and applications on nodes, at cluster and grid level.

As part of research conducted in this thesis, the efficiency of grid computing in the field of electromagnetic simulation of complex and large structures was demonstrated [5, 6, 8, 7], as a challenge of the HEMERA project [49]. The simulations were performed using

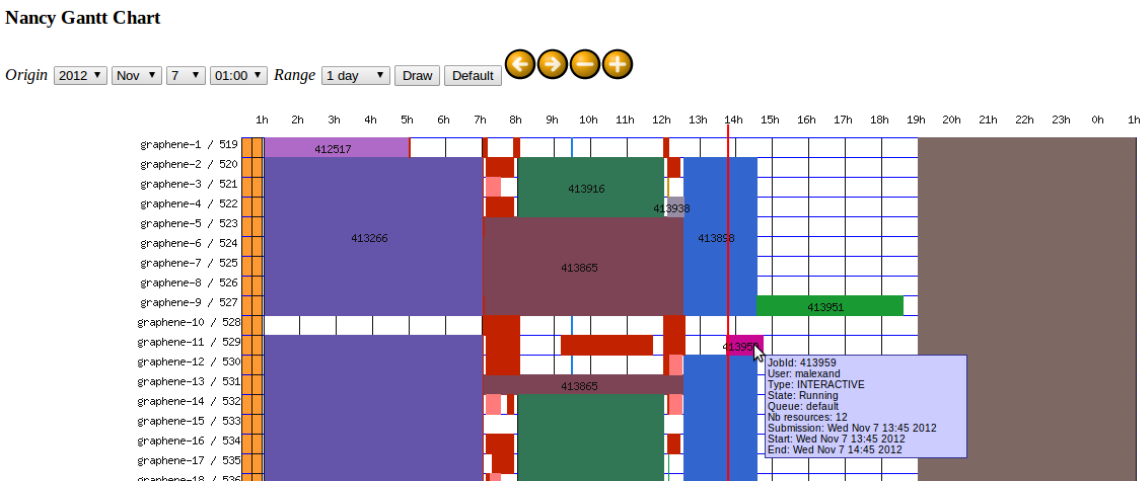


Figure 3.3: Drawgantt charts displaying past, current and scheduled OAR jobs.

computing nodes from clusters: Griffon (Nancy site), Chinqchint (Lille site), Paradent, Parapide (Rennes site) and Genepi (Grenoble site).

During the research work, an appropriate procedure for experiments performed on Grid'5000 platform was established. It is based into two steps: the development of experiments and then the execution.

1. Experiment development:

   (a) **Application development**: The application is developed to be executed on grid. In most cases, it is about massively parallel applications. A large number of tasks are executed in parallel, making full use of the grid capabilities.

   The grid can also deal with parametric applications. In this case, the same application is executed many times, with different input parameters. The grid can save time by putting these executions in parallel on a large number of computing nodes. Electromagnetic simulations of large and complex systems involve this type of applications, for example in a multimodal modeling approach.

   (b) **Initialization Phase**: This phase includes all operations to be performed before starting the experiment. It may be, for example, the generation of the input files for the application.

   (c) **Environment customization**: Once the application is ready to be distributed in a parallel computation, the user can create a customized work environment. This is a Linux operating system containing the application to be launched and various tools and libraries needed for the experiment. The data necessary for the execution of the experiments, such as the input files generated during the *Initialization Phase* are transferred to the environment. So far, the grid is only used occasionally, mostly for testing the environment.

2. Experiment execution:

   (a) **Resource reservation**: User reserves, for a given period of time, the resources he needs. These resources can be located on one cluster or they can even be distributed over several sites.

   (b) **Environment deployment**: The customized environment can then be automatically deployed on all the computing nodes used. These are then restarted to use this system.

   (c) **Execution**: The application can then be executed on each reserved node.

   (d) **Recovery results**: The data resulting from the execution of the application are collected and made available to the user on the NFS server of the concerned sites. The data presented locally on each node are erased during machines reboot.

3.1.2 Hyperion-Calmip

Experiments have been performed also on Hyperion, a supercomputer which belongs to the Calmip group [23]. This machine was ranked 223^{th} on the TOP 500 machines in November 2009, having a power of 33 Teraflop. The computing system consists of a cluster Altix ICE 8200 of 352 computing nodes, each node having two quad-core Nehalem EX clocked at 2.8 GHz with 8 MB cache per processor and 36 GB of RAM. The nodes are connected by Infiniband network. The computing resources are managed through PBS scripts.

The computing nodes used on Grid'5000 clusters are almost all based on Intel Penryn microarchitecture, while the Hyperion nodes provide the benefits of the Intel Nehalem microarchitecture [118]. Nehalem micro-architecture provides some state-of-the-art technologies which enable high computation rates for scientic workloads. Nehalem performance is based on the fact that DRAM controller, the L3 cache level and Quick path interconnect (QPI) ports are all housed within the same silicon die as the four cores. This means a high performance computing by low memory latency, high memory bandwidth and an efficient cache miss approach, apects that can be observed also with the TLM application used in this work.

3.2 Performance assessment on parallel architectures

The performance of a parallel system can be illustrated by several indicators [45]. The speedup is a parameter used to evaluate the gain in terms of simulation time with N parallel computating processors compared to the same calculation on a single processor. The speedup [131] is defined as the ratio between the time of the sequential and the parallel execution. The computational efficiency is the ratio between the speedup and the number of processes. The

total simulation time equals the summation of the communication time and the computation time of the longest task.

3.3 Application execution-time prediction

When dealing with massively parallel computations performed on distributed computing systems, the estimation of the application execution time is crucial, due to the time and memory constraints imposed by the computing platforms. The knowledge of the execution time of the applications is required by the batch schedulers, to manage the order of the execution of applications that are submitted to them [21]. The scheduling mechanisms provided by the clusters and grids systems [111, 112], need to know an estimation of the duration of the applications to perform. They use, for example, the Worst-Case Execution Time (WCET).

The users have to mention the execution time when they submit applications to the grid computing. This estimation has to be correct, because an underestimated time causes a premature stop of the application, while an overestimated time gives a late on the startup of the application if there are no enough free resources for the required time [42]. This is the case encoutered also on Grid'5000 platform, using OAR tool for resource allocation.

There are two main families of execution time prediction techniques presented in the literature:

1. historic-based prediction
2. profile-based prediction

The historic-based prediction technique is especially used to predict the time, sequential or parallel, of an application in order to perform simulations on cluster or grid computing. The historic-based prediction approach, estimates the execution time of an application by the execution time of the same application obtained during past experiments.

This approach considers that the execution time of an application depends on the context in which it is launched. Two executions carried out in contexts relatively close provide similar execution time [42]. The problem that arises is to define and to quantify the notion of proximity between contexts. The context of execution of an application is defined by the hardware architecture used and the input parameters.

Considering the experiments as points, it is possible to calculate a distance between them, according to the euclidian distance, Manhattan distance or Minkowski distance [129]. The distance is much smaller as the values of the experiments are similar.

The profile-based prediction technique is particularly used for real-time systems, in order to determine the execution time of an application in the worst case (WCET). The execution time in the worst case is the maximum time that the application will run, given a set of

inputs and a hardware architecture. It is therefore the time taken by the application to execute when its inputs determine a path that increases the execution time.

WCET must satisfy two main criteria [128, 94]: The first one is a secure criterion, because the WCET must absolutely be higher than the maximum execution time of the application. If this constraint is not fullfilled, the failure of the period of resource reservation is critical on real-time systems. The second criterion is a precision criterion, more flexible than the previous one. The precision of the WCET provides the gap between the predicted value and the real value. To be useful, the estimated execution time of an application should not be too pessimistic, otherwise cause oversizing the system.

WCET is achieved through a complex analysis based on dividing the program into basic blocks. A basic block is a maximal instructions sequence having a single entry point and a single exit point in the control flow graph of the program. A basic block contains only simple instructions, excluding any controle instruction or function calls [76]. Analysis uses a control flow graph to determine the set of possible execution paths between the blocks of the program. According to the hardware architecture, WCET is defined by the longest flow in time. Several techniques for the computation of WCET exists: path-based techniques [114], tree-based techniques [29] and implicit path enumeration techniques [19].

The latter, transforms the control flow graph into a set of constraints that must be accomplished in order to obtain a linear optimization problem formulation, based on integer variables. Two types of constraints are distinguished: the structural constraints - these constraints describe the graph structure, and functional constraints - these constraints express the complementary information on the control flow.

The predictive model for computation time proposed in this research work is a hybrid based on the two approaches described above. The model consists in decomposing the application in blocks where the CPU spends various periods of time during the execution, depending on the input parameters of the application. The model uses a series of previous experiences to determine, using a linear optimization formulation, the time allocated to each block, taking into account the material architecture of computing nodes where the simulation is performed. The model was designed also to consider the cache mechanisms found on the computing nodes, in oder to increase the accuracy of estimated values of computation time when dealing with TLM simulations for oversized structures.

3.4 Conclusions

This chapter gives an overview on parallel and distributed computing, starting with some historical mentions about the development of the parallel approach in the computing field, well-known parallel computing systems and programming approaches.

A short presentation highlights the computational grid technology, merging the parallel

and distributed computing with web and security technologies, in order to provide a complex and dynamic computing tool, in a multidisciplinary paradigm application, completely masking the system complexity to the scientists. The grid provided services are based on the grid middleware, which is the core of the system. Also, the most important grid platforms used in real-life applications are mentioned.

The two platforms that provide the experiment in this thesis are presented. Hyperion supercomputer, based on Intel Nehalem microarchitecture, is a very powerful machine conceived for high performance computing. Grid'5000 is a large scale experimental tool, with deep reconfiguration capability, a controlled level of heterogeneity and a strong control and monitoring infrastructure.

A protocol for the experiments launched in this thesis is proposed in order to easily and time-efficient place computational electromagnetics applications on the grid platform, Grid'5000. Several parallel computation performance assessment indicators are summarized. They are applied on parallel TLM/modal hybrid application and results are given in the next chapters. Given the importance of estimating the execution time of applications running on parallel and distributed systems, a hybrid historic and profile based time prediction model for the parallel TLM/modal hybrid application, is proposed.

Chapter 4

Transmission-Line Matrix Modal Hybrid Approach

In this chapter, a hybrid numerical modeling tool based on TLM and modal approach is proposed to be used when dealing with complex structures, taking advantage of the two methods to get the most efficient solution to the electromagnetic problem to be analyzed. First, the two methods are presented. The hybrid approach implementation is described and validated by some numerical examples.

4.1 Transmission-Line Matrix Method

4.1.1 Overview

The concepts that the TLM is based on appear in 1944 in G. Kron's article [69], where it is described the electric circuit which models the Maxwell's equations for the electromagnetic field. Later, in 1971, P.B. Johns and R.L. Beurle publish the TLM method for calculating the electromagnetic scattering process for the case of two-dimensioned problems. In 1995, C. Christopoulos added its contribution to the TLM for numerical modeling of microwave components.

In all the numerical methods, the discretization plays a fundamental role. It introduces the numerical dispersion, which varies with the frequency and the propagation direction. In the case of the TLM, so that the results of the simulation can be taken into account the mesh step $\Delta x \ll \lambda$, has to be for example:

$$\Delta x \leq \frac{\lambda}{10} \tag{4.1}$$

The dispersion increases with frequency. The space discretized upon TLM represents a discrete system whose solutions gradually approach the solutions of a continuous system with the mesh step tending towards zero ($\Delta x \to 0$). If the frequency of the propagating

signal on the transmission line model increases, the mesh step size will not be very small with respect to the wavelength.

For open systems, such as a uniform waveguide where the field propagates at infinity, when applying the discretization of TLM, a termination of the mesh for the simulated system must be established. Thus, the TLM nodes that are on the termination must absorb all incident signal so that there is no reflected signal.

Such a termination is called absorbing boundary condition (ABC). If the impedance of the wave does not depend on frequency (the impedance of TEM mode) and the angle of incidence is known, then the ABC conditions are common place and are implemented at the end of the lines, on the edge of the mesh, by a suitable impedance.

TLM represents a numerical method for the electromagnetic simulation which fills the environment of the electromagnetic field propagation with a network of transmission lines. This model of the propagating field inside a given medium becomes possible thanks to the equivalence that exists between the electric and magnetic fields and voltages and currents in a transmission line network. The voltages and currents on a transmission line can model the magnetic and electric field propagating as a plane wave (1-dimension).

Two-dimension field propagation is studied by means of a two-dimensional transmission line network. In this case, *shunt* and *series* nodes are used to characterize the medium to simulate. Three-dimension field propagation is modelled with the Symmetrical Condensed Node (SCN) in three-dimensional transmission line networks. The plane wave has been considered in an unbounded, linear, isotropic, homogen space where the distribution of the electromagnetic field is uniform in parallel planes. The axis propagation is perpendicular to the planes of uniformity. By consequence, the electromagnetic field is constant with respect to x and y axes.

TLM offers two different approaches to modeling the non-uniform space: *graded mesh* and *multi-grid*. The first technique, *graded mesh* [101], models non-uniformities and irregular forms with the SCN stub loaded nodes or by the hybrid method. In the first case, the number of 12 transmission lines of the SCN node is increased by adding three open circuited transmission lines for the elevation of the relative permittivity and three short circuited transmission lines for the elevation of the relative magnetic permeability. The first 12 lines have the same characteristic impedance.

In the second case, the hybrid SCN node models the space by considering different characteristic impedances for the first 12 transmission lines of the SCN node, taking into consideration only the variation of relative permeability. It adds only three open circuited lines to consider the relative permittivity.

Given that the model with stubs uses more information than the hybrid model, then it is considered more accurate too. Transmission lines used in TLM have two tasks: consider the properties of the propagation medium anywhere in space and at the same time, always

keep the pulses running through the TLM lines synchronous.

The second technique, *multi-grid mesh*, divides the medium to be modelled in regions that have the same properties. Where the field varies very quickly a finer spatial resolution is used. Thus, this technique offers greater precision.

The TLM method discretizes the space where the electromagnetic field propagates along the three dimensions. Thus, using plans on all the three directions with equally distant with $\Delta x, \Delta y, \Delta z$ it will get small parallelepipeds with dimensions $\Delta x, \Delta y, \Delta z$. Indeed, the three dimensions $\Delta x, \Delta y, \Delta z$ represent the mesh step in each direction.

These small parallelepipeds whose space has been divided are called TLM cells. If mesh step has the same size of the three directions, $\Delta x = \Delta y = \Delta z = \Delta l$ (the dimensions of the TLM node) then the TLM cell is a cube. The tangential planes between neighboring cells define the ports. Transmission lines joining neighboring cells by coupling the centers of the cells by means of the ports. The centers of cells are called scattering centers.

The distance between two TLM nodes is equal to $\Delta l/2$. When we take the distance between two nodes A and B, we take into consideration the distance from the center of A up to the port connecting with B, it means that the length of the transmission line is $\Delta l/2$; therefore, the propagation time of an impulse between two nodes is:

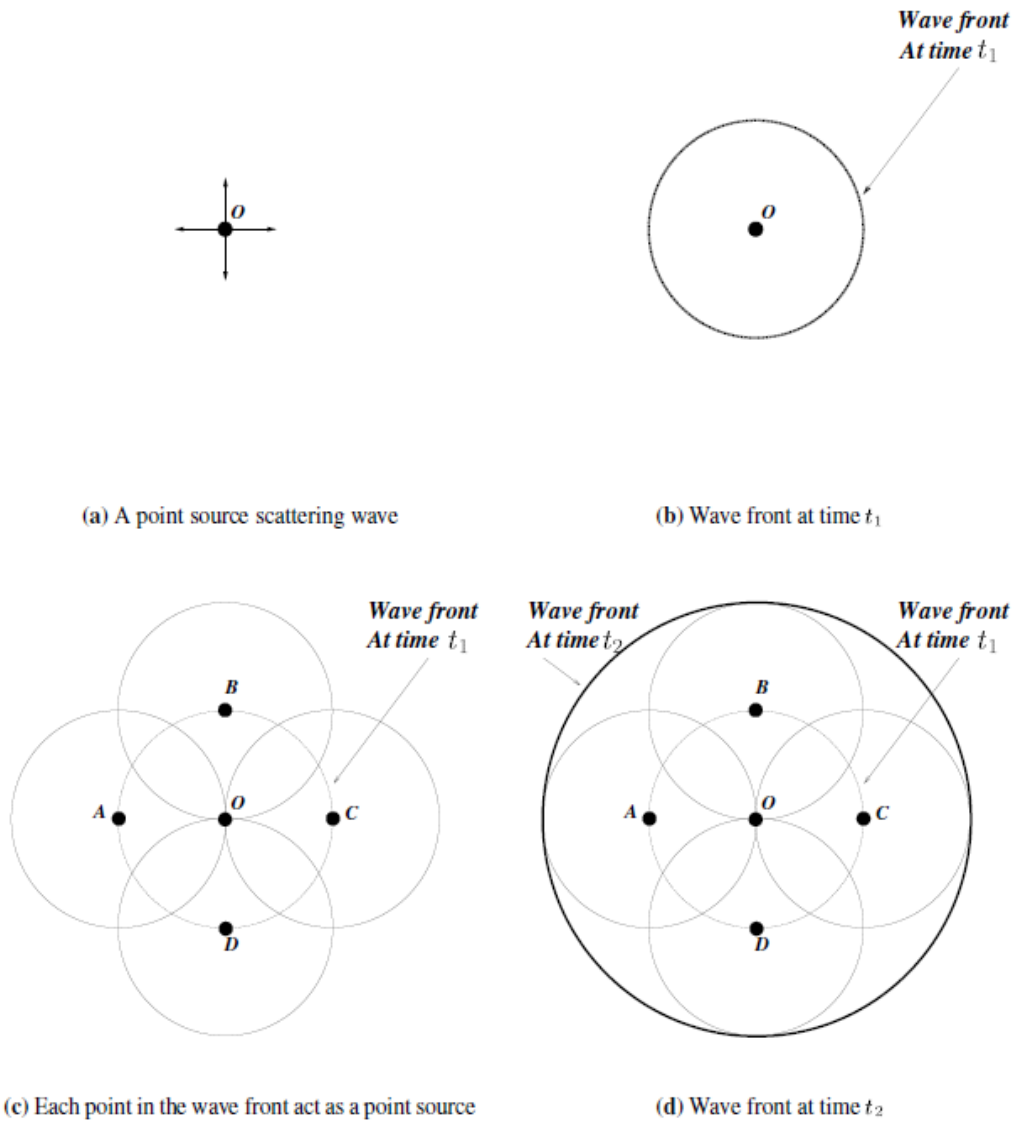
$$\Delta t = \frac{\Delta l}{2c} \tag{4.2}$$

Before applying the TLM method, a presentation of its basic concepts and the main electromagnetic applications is done.

4.1.2 Huygens principle

The model presented by Christian Huygens in the seventeenth century for the propagation of the electromagnetic radiation based on the wave nature of this phenomenon is used by P. B. Johns and R. L. Beurle in 1971 to define a new two-dimensional numerical method that would be called Transmission-Line Matrix (TLM), with practical applications in modeling the scattering electromagnetic problems.

Briefly, Huygens's principle, [53], - see Fig. 4.1- says that a wave front consists of several sources of radiation and that each one produces new spherical wavelets. By merging these wavelets, a new wave front raises continuing to propagate in the same manner. It is a continuous model for the propagation and scattering of the electromagnetic waves and is the algorithm that defines the TLM.



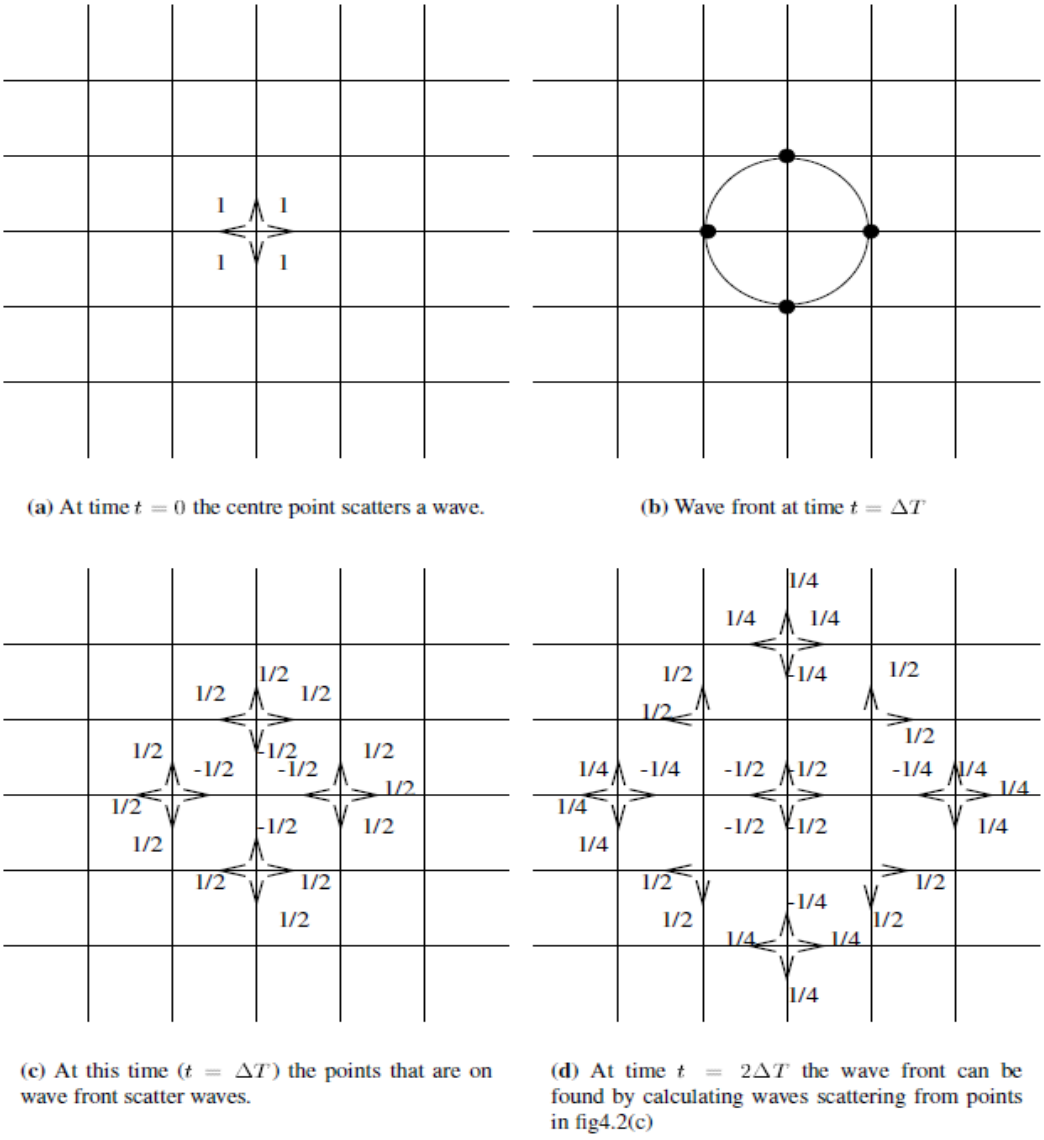
(a) A point source scattering wave

(b) Wave front at time t_1

(c) Each point in the wave front act as a point source

(d) Wave front at time t_2

Figure 4.1: Huygens principle [2].



(a) At time $t = 0$ the centre point scatters a wave.

(b) Wave front at time $t = \Delta T$

(c) At this time ($t = \Delta T$) the points that are on wave front scatter waves.

(d) At time $t = 2\Delta T$ the wave front can be found by calculating waves scattering from points in fig4.2(c)

Figure 4.2: Wave propagation in a two-dimensional TLM network [2].

4.1.3 TLM basics

To achieve a digital model of wave propagation phenomenon starting from the model described by Huygens, the space and time must be discretized as follows:

$$\Delta t = \frac{\Delta l}{c}, \tag{4.3}$$

where the two-dimensional space is represented by an array of points spaced with Δl. A Dirac impulse incident at one of the nodes propagates to neighboring nodes during Δt period with the speed of light c. By joining these nodes with orthogonal transmission lines of length Δl, an analog network model or the transmission-line matrix with shunt nodes is obtained - see Fig. 4.2.

Starting from the two-dimensional network of transmission lines it is been demonstrated the equivalence of the electromagnetic field that would propagate in a similar plan and the current and voltage pulses that propagate along the transmission line network. With this digital model of transmission line network, phenomena like propagation, reflection, refraction and attenuation of the electromagnetic field in two-dimensional structures can be calculated by means of digital computers.

If the mesh step Δl is small enough compared with the wavelength, the TLM model is not frequency or spatial dispersive, so all the spectral components of the signal which excites the two-dimensional network propagates at the same speed in all directions. Starting from the shunt node equivalent model with lumped elements L, C and applying Kirchoff's laws the two-dimensional wave equation in voltage or current is obtained. Maxwell's equations are applied in a similar plan to obtain the two-dimensional wave equation in electric or magnetic field. Comparing the two relations is found the equivalence between the TLM network equation parameters and the electromagnetic field components from Maxwell's equations, [64]:

$$\begin{aligned} E_y &\equiv V_y \\ H_z &\equiv I_x \\ H_x &\equiv -I_z \\ \mu &\equiv L \\ \epsilon &\equiv 2C \end{aligned} \tag{4.4}$$

Signal propagation velocity inside the two-dimensional TLM network is $c/\sqrt{2}$ - c is the velocity of light - , the environment of the network formed by the intersection of the transmission lines being characterized by a double relative permittivity of the vacuum due to the parallel connection between the circuits of the two line segments on node. The equivalence between the electromagnetic field components and the TLM network voltages and currents is valid only if the mesh step is small enough compared to the wavelength.

But, if the mesh step size approaches the wavelength, the shunt TLM node circuit model is not more valid and TLM does not simulate an isotropic wave propagation. In Fig. 4.3, it is shown that TLM can model Maxwell's equations only for a limited range of frequencies, between zero and the first network cutoff frequency corresponding to $\Delta l/\lambda = 1/4$. For frequencies lower the network cutoff frequency, the wave propagation velocity is approximated by $c/\sqrt{2}$.

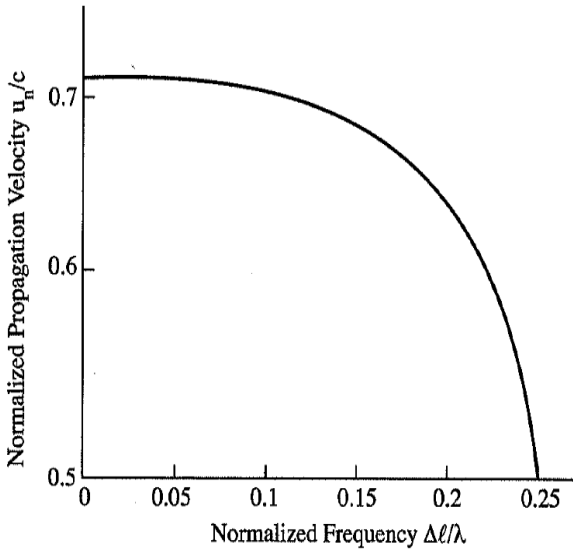


Figure 4.3: Dispersion of the velocity of waves in a two-dimensional TLM network [64].

4.1.4 Scattering matrix

After the network excitation with the delta function impulses on the input nodes, they begin to spread to the neighboring nodes over the transmission lines connecting them, reflecting on each node. At each iteration, the impulses reflected on a node become the incident signals to the neighboring nodes at the next iteration. A computation iteration is equivalent with the time period of an impulse propagation between two neighboring nodes, $\Delta l/c$. Thus, the network is covered by impulses.

The relation between the incident impulses at the time step k and the reflected impulses at *k+1* from a node is given by the scattering matrix, S:

$$_{k+1}\begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \end{bmatrix}^r = \frac{1}{2}\begin{bmatrix} -1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 \\ 1 & 1 & -1 & 1 \\ 1 & 1 & 1 & -1 \end{bmatrix}_k \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \end{bmatrix}^i \tag{4.5}$$

Impulse propagation between the TLM network nodes is modeled by the equation:

$$\begin{aligned} {}_{k+1}\mathbf{V}_1^i(z, x+\Delta l) &= {}_{k+1}\mathbf{V}_3^r(z,x) \\ {}_{k+1}\mathbf{V}_2^i(z+\Delta l, x) &= {}_{k+1}\mathbf{V}_4^r(z,x) \\ {}_{k+1}\mathbf{V}_3^i(z, x-\Delta l) &= {}_{k+1}\mathbf{V}_1^r(z,x) \\ {}_{k+1}\mathbf{V}_1^i(z-\Delta l, x) &= {}_{k+1}\mathbf{V}_2^r(z,x) \end{aligned} \tag{4.6}$$

Thus, an impulse reflected from the node of cartesian coordinates *(z, x)* is incident on

the neighboring nodes. Knowing the magnitude, the position and the propagation direction of network impulses at time step $k\Delta l/c$, the network state at $(k+1)\Delta l/c$ can be determined.

4.1.5 Stub loaded node

Non-homogeneous structures, with or without losses, are modeled using two-dimensional TLM network. If the structure modeling environment is homogeneous and lossless, the media properties should not be considered. It is only the ratio between the network phase velocity and the environment phase velocity that changes with the material properties of the modeled environment. If the simulated environment is non-homogeneous, the material properties must be considered when building the TLM network.

To model the permittivity, permeability and loss tangent angle that vary along the structure, corresponding reactive and dissipative elements to that region of the structure, called stubs, will be added to the shunt nodes. A stub of length $\Delta l/2$, finished with an open circuit, having the caracteristic admittance as a function of modeling environment relative permittivity, is seen inside the node as a capacitor.

For this reason a new model for a node is created, as in [59, 60].

A matched stub models the energy losses in the network, being seen inside the node as a resistor.

4.1.6 Boundary Conditions

The reflection of the electromagnetic field at two-dimensional structures boundaries is modeled using the reflection coefficients imposed on the TLM network. If the voltage impulses that propagate in the TLM network model the electric field and the current impulses model the magnetic field, then the electric wall boundary is simulated by a short circuit and the magnetic wall is simulated by an open circuit, at the corresponding nodes of the network boundaries.

The plan of the boundary is placed at the halfway between the nodes, so the reflected pulse at the boundary travels the same distance as the other impulses in the network, maintaining the synchronism. Considering the boundary noted with the letter C in Fig. 4.4. If this models a magnetic wall, the boundary nodes have a reflection coefficient equal to *+1*, the reflected signal having the same amplitude and phase with the incident signal. If the boundary is an electric wall, the reflection coefficient is *-1*.

$$ {}_{k+1}\mathbf{V}_4^i(p,q) = {}_k\mathbf{V}_2^r(p+1,q) = \Gamma[{}_k\mathbf{V}_4^r(p,q)] \tag{4.7} $$

If the boundary has a surface impedance Z_c, the reflection coefficient is defined with

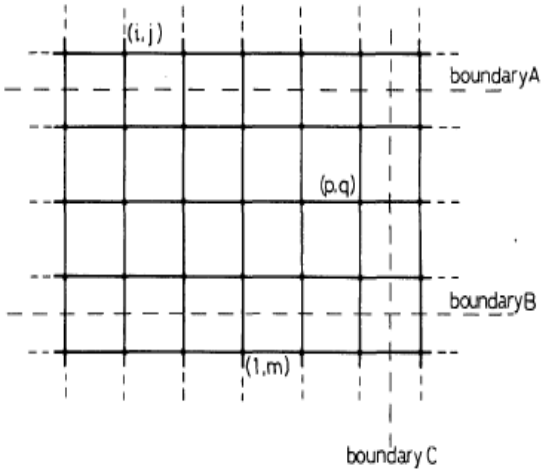


Figure 4.4: Boundaries of a two-dimensional TLM network [64].

respect to the free space impedance, Z_0, as:

$$\Gamma = \frac{Z_c - Z_0}{Z_c + Z_0} \tag{4.8}$$

4.1.7 Outputs

To obtain the frequency response of the TLM structure with impulsive excitation, Fourier transform is performed for the output impulse function on the network observation point. This output impulse function is built retaining at each time step impulses passing through this point. An impulsive excitation is applied when you wish the simultaneous presence of several spectral components in the TLM network.

For other types of excitations applied to the network, the structure response is obtained by the convolution product between the output impulse function and the signal excitation. By the inevitable limitation of the number of iterations, the spectral response of the network is no more a simple line but a $\sin x/x$ functions superposition (Gibbs phenomenon), called truncation error.

4.1.8 TLM 3D - SCN

To represent the propagation of the electromagnetic waves in a three-dimensional space, in [4] is shown a network of transmission lines in three-dimensions formed by the interconnection of two-dimensional three shunt nodes and three series nodes, to describe simultaneously all six components of the electromagnetic field. This network, called *expanded-node network*, allows the impulse propagation between two nodes in $\Delta t/2$; three of the six components of the electromagnetic field are concentrated in each node. The disadvantage of this network is given by the relatively complex topology which makes difficult the computation of the field

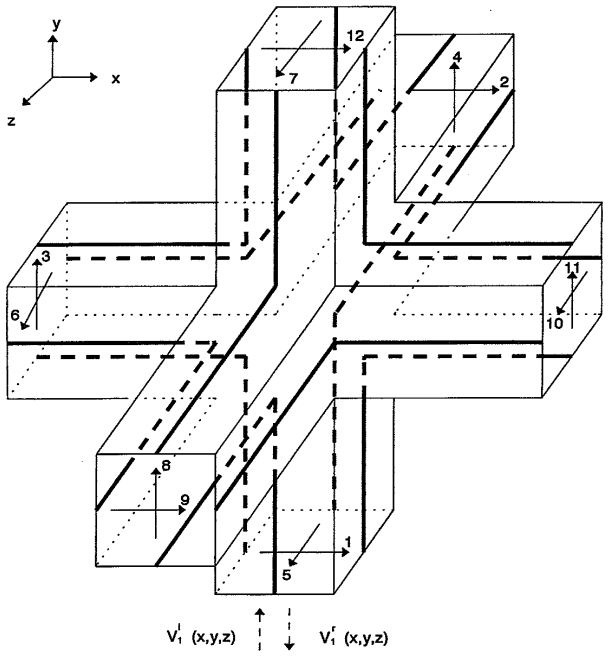


Figure 4.5: Symmetrical Condensed Node [63].

components.

P. Saguet and E. Pic present a different three-dimensional network topology based on the interconnection of shunt and series nodes, called *condensed node* structure [103], which simplifies the computation procedure. All the scattering processes that appear on a node at a given time step, occur in a single point in space. All the field components corresponding to a node are described in a single point in space.

The disadvantage of the condensed node structure is given by the difficulty to define the boundaries of the structure to be modeled at high frequencies because of the asymmetry of the network. Depending on the direction in which we connect to the condensed node, either a shunt node or a series node is met.

In 1987, P. B. Johns presents in an excellent paper [63] the *Symmetrical Condensed Node* (SCN) for transmission line modeling of electromagnetic waves. This node focuses the six components of electromagnetic field in a single point in space, eliminating the problems of asymmetry. Other structures derived from SCN node as the *Hybrid Symmetrical Condensed Node* (HSCN) or *General Symmetrical Condensed Node* (GSCN) are described in [17, 83, 105, 120, 119, 121]. Below, a brief description of the Johns's n-port - the symmetrical condensed node - is presented.

4.1.8.1 Symmetrical Condensed Node (SCN)

SCN without derivations (SCN without stubs) can be imagined as a cube with a branch on each of its six sides, as in Fig. 4.5. On each branch there are two gates leading pulses on

$$\mathbf{S} = 0.5 \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 & 1 & 0 & -1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & -1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & -1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \end{bmatrix}$$

Figure 4.6: Scattering matrix modeling homogeneous media.

two transmission lines. Therefore, the SCN has 12 ports, 12 transmission lines, each line has two threads. The polarization of the TLM pulses follows the direction vectors of the port it spreads, each pulse containing two field components, E and H. The E component is parallel to the direction of polarization of the pulse and the H component is perpendicular to the polarization direction of the pulse.

The 12 ports can have on each propagation direction the two types of polarization. The voltage pulses corresponding to these polarizations are launched on the even lines, which do not couple. The numbering of the ports of the SCN is very important in obtaining the dispersion matrix of the node.

The SCN topology cannot be represented by an equivalent electric circuit. The scattering matrix which characterizes the node is obtained using Maxwell's equations and the energy and charge conservation principles. The scattering matrix S of the SCN for homogeneous media, relating the reflected and incident impulses is a 12x12 matrix, as in Fig. 4.6.

For an axial propagation, the low frequency velocity of waves on the SCN network is half of the medium propagation velocity. Waves propagate without dispersion. The propagation at 45 degrees according to the axes, is altered by the dispersion as it is presented in [10, 84].

The computation process in the SCN network is similar to that described for two-dimensional mesh. Once the boundary conditions have been established, the structure is excited and the impulses start spreading according to the scattering matrix S characterizing the SCN node that models an environment: homogeneous or non-homogeneous, with or without losses.

$$V^r = SV^i \tag{4.9}$$

The reflected impulses from a node at time step k become the incident impulses to the neighboring nodes at time step *k+1*. This step makes the connection between the network nodes. The total number of iterations depends on the problem and on the required frequency resolution.

In order to excite a certain component of the field it is required to inject impulses on the corresponding ports of the node.

To simulate the electromagnetic field propagation in a three-dimensional non-homogeneous medium, six lines long $\Delta l/2$ called stubs are added to the SCN node structure. Depending on the material properties, three lines coupled with the three components of the electric field, are open circuited and add capacity to the node, and the other three lines coupled with the three magnetic field components, are short circuited and add inductance to the node.

The characteristic stub admittances corresponding to the capacity added to the node and normalized to the admittance of the background transmission lines ($Y_o = \sqrt{\varepsilon_o/\mu_o}$) are:

$$\begin{aligned} Y_x &= \frac{2\varepsilon_r}{u_o \Delta t}\frac{\Delta y \Delta z}{\Delta x} - 4 \\ Y_y &= \frac{2\varepsilon_r}{u_o \Delta t}\frac{\Delta x \Delta z}{\Delta y} - 4 \\ Y_z &= \frac{2\varepsilon_r}{u_o \Delta t}\frac{\Delta x \Delta y}{\Delta z} - 4 \end{aligned} \tag{4.10}$$

where, ε_o is the permitivity of the free space, μ_o is the permeability of the free space, ε_r is the relativ permitivity of the media to be modeled, u_o is the free space wave velocity, Δt is the time step, $\Delta x, \Delta y, \Delta z$ are the dimensions of the TLM cell according to the three Cartesian directions.

The characteristic stub impedance corresponding to the inductance added to the node and normalized to the impedance of the background transmission lines ($Z_o = \sqrt{\mu_o/\varepsilon_o}$) are:

$$\begin{aligned} Z_x &= \frac{2\mu_r}{u_o \Delta t}\frac{\Delta y \Delta z}{\Delta x} - 4 \\ Z_y &= \frac{2\mu_r}{u_o \Delta t}\frac{\Delta x \Delta z}{\Delta y} - 4 \\ Z_z &= \frac{2\mu_r}{u_o \Delta t}\frac{\Delta x \Delta y}{\Delta z} - 4 \end{aligned} \tag{4.11}$$

where, μ_r is the relativ permeability of the media to be modeled.

The connection process inside the TLM SCN network is not influenced by the stubs because they are internaly connected to the node and does not interact directly with the neighboring nodes. The scattering process is affected by the appearance of stubs and the scattering matrix is of the form 18x18 - see Fig. 4.7. The three capacitive stubs correspond

S =	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	a	b	d						b		-d	c	g					i
2	b	a				d			c	-d		b	g				-i	
3	d		a	b				b			c	-d		g				-i
4			b	a	d		-d	c			b			g		i		
5				d	a	b	c	-d		b					g	-i		
6		d			b	a	b		-d	c					g		i	
7				-d	c	b	a	d		b					g	i		
8			b	c	-d		d	a			b			g		-i		
9	b	c				-d			a	d		b	g				i	
10		-d			b	c	b		d	a					g		-i	
11	-d		c	b				b			a	d		g				i
12	c	b	-d						b		d	a	g					-i
13	e	e							e			e	h					
14			e	e				e			e			h				
15					e	e	e			e					h			
16				f	-f		f	-f								j		
17		-f				f			f	-f							j	
18	f		-f								f	-f						j

Figure 4.7: Scattering matrix modeling non-homogeneous media.

to the gates 13 through 15, and the three inductive stubs correspond to the gates 16 through 18. Applying Kirchoff's laws and energy conservation principle:

$$S^T Y S = Y \tag{4.12}$$

we can determine the S matrix coefficients, as in [28], by:

$$\begin{aligned}
a &= \frac{-Y}{2(4+Y)} + \frac{Z}{2(4+Z)} \\
b &= \frac{4}{2(4+Y)} \\
c &= \frac{-Y}{2(4+Y)} - \frac{Z}{2(4+Z)} \\
d &= \frac{4}{2(4+Z)} \\
e &= b \\
f &= Zd \\
g &= Yb \\
h &= \frac{Y-4}{Y+4} \\
i &= d \\
j &= \frac{4-Z}{4+Z}
\end{aligned} \tag{4.13}$$

At any iteration during the simulation, at any point in the discretized computation space, the electric field and the magnetic field can be expressed. To calculate a component of the electromagnetic field in a certain space point, at a given time step, all the impulses on lines which couple with that field component at that space point, have to be considered.

$$\begin{aligned} E_x &= -\frac{2(V_1^i + V_2^i + V_9^i + V_{12}^i + Y_x V_{13}^i)}{\Delta x(4 + Y_x)} \\ E_y &= -\frac{2(V_3^i + V_4^i + V_{11}^i + V_8^i + Y_y V_{14}^i)}{\Delta y(4 + Y_y)} \\ E_z &= -\frac{2(V_5^i + V_6^i + V_7^i + V_{10}^i + Y_z V_{15}^i)}{\Delta z(4 + Y_z)} \\ H_x &= \frac{2(V_4^i - V_5^i + V_7^i - V_8^i - V_{16}^i)}{\Delta x(4Z_o + Z_o Z_x)} \\ H_y &= \frac{2(-V_2^i + V_6^i + V_9^i - V_{10}^i + V_{17}^i)}{\Delta y(4Z_o + Z_o Z)} \\ H_z &= \frac{2(V_1^i - V_3^i + V_{11}^i - V_{12}^i + V_{18}^i)}{\Delta z(4Z_o + Z_o Z)} \end{aligned} \tag{4.14}$$

The SCN node loaded with lossy stubs simulating lossy media is described in [28](pg. 136). A cylindrical SCN is outlined in [134].

4.1.9 Error sources and corrections

The most common errors that occur in the electromagnetic structure models based on TLM are:

Truncation error

It is an error that is due to inevitable limitations on the number of iterations, ie the structure time-domain impulse response truncation. The spectrum of this response is not a line but a superposition of $\sin x/x$ functions that interfere creating a shift for the peak of the spectral components excited in the structure. Ways to reduce this error are displayed in [51, 102].

Coarseness error

It is an error that occurs when the mesh step is too large compared to the wavelength of the simulated field. A simulation with a smaller mesh step requires significant memory resources. A more efficient (flexible) approach would be a variable mesh size.

Velocity error

It is an error that occurs by considering the propagation velocity of the impulses being constant inside TLM network in all directions when the mesh step size approaches the wavelength. Because in this case the impulses propagate at different speeds depending on the direction, which alters the cutoff frequency of the simulated structure.

Misalignment error

It is an error that occurs simulating dielectric interfaces on the boundaries; dielectric interfaces appear halfway between the boundary nodes, while the electric and magnetic walls boundaries appear in the node.

A detailed presentation of TLM method, also in a threedimensional approach, can be found in a review paper [51] and in a book chapter [55] by Hoefer.

4.2 Modal Approach

Mode-Matching method [55, 78] is often used to model boundary problems, dispersions in guiding structures, that have separable geometry, each region having its own coordinate system. This method describes the electromagnetic field along the discontinuity by infinite series of normal modes at the junction surface. In numerical calculation, given the limited resources of calculation, the amputation of the series is a major issue in the validity of results.

In order to describe the electromagnetic field in the plane of the discontinuity, the tangential fields on both sides of the junction are decomposed into infinite series of normal modes. The problem lies in calculating the modal coefficients which can lead to an infinite number of simultaneous linear equations. By truncation of the modal series describing the field around the discontinuity, a finite number of unknowns, which can be determined analytically or numerically, is reached, obtaining a numerical solution. By this approximation, it is possible that the results accuracy decreases due to the convergence problems.

The numerical implementation of this method can deal with a wider range of problems. The Mode Matching method was applied to: scattering problems caused by the discontinuities inside the waveguides [127, 99], micro-strip lines [130] and finlines [87]; eigenvalue problems, obtaining the resonance frequency of a cavity, the cutoff frequency of a waveguide, the propagation constant of a transmission line or planar transmission line; analysis of composite structures such as E-plane filters [110], direct-coupled cavity filters, waveguide impedance transformers [12], power dividers [11] and microstrip filters [75].

A series can be truncated only if it converges. To demonstrate the convergence of the numerical results, they are displayed in relation with the number of modes considered in the truncated series. Comparing the results variation according to a specific criteria, a small variation means a higher accuracy.

When the problem to be analysed imposes the truncation of many series in the same time, the convergence problem is called relative convergence problem. The numerical results are affected by the ratio between the number of modes considered when describing the electromagnetic field in different regions of discontinuity. Depending on how the solution of the problem is calculated - determining the unknown coefficients of the field in each region of the junction - the relative convergence phenomenon may be more or less important.

When for different values of the number of modes considered to represent the field, different results are obtained, the behavior of field at the junction has to be examined. Thus, rules can be generated for the infinite series approximation, based on the ratio between the number of the considered modes and the different sizes of the junction regions.

4.3 TLM/Modal Hybrid Approach

4.3.1 State of the art

The analysis of microwave circuits is often realised discretizing the circuits by TLM time-domain method. In this case, the boundaries are usually modelled with diakoptics. Originated by Kron [70] for the frequency domain methods, diakoptics was developped for time-domain TLM by Johns and Akhtarzad in [57, 58]. Diakoptics is a method of decomposing the large structures into subdomains which are solved independently and later reassembled. [36] presents the implementation of wide-band ABCs with Johns's time-domain diakoptics approach applied to two-dimensional waveguide discontinuities. The impulse response or Green's function of the discretized domain is convolved with the impulse response of boundary condition which is represented by three-dimensional array called Johns matrix.

Three years later, an extension of the TLM diakoptics procedure outlined above, for higher order modes is presented in [99]. The TLM discretized domain is limited only around the discontinuity. The remaining computation domain is modeled by modal approach since the media is homogenous. This means saving numerical resources. The incident and reflected pulses field representation over the reference plane are transformed into a modal field representation. The reference planes and boundary conditions can coincide, very close to the discontinuity, without creating instabilities due to the higher order modes excited by this latter, because the convolution is carried out for each mode separately. This means the posibility of the simulation not only matched waveguides (ABC's) but also dispersive terminations.

A similar hybrid-modeling, based on the scattering phenomena and diakoptics procedure is presented in [98, 97], where the discontinuity is discretized upon TLM and the homogeneous space by modal approach, given the low computing resources.

The diakoptic procedure for three-dimensional structures discretized upon TLM - SCN is brought out in [98], making possible the hybrid analysis of waveguide components by time-domain analysis for the discontinuities and analytical modal approach for homogenous waveguide sections.

A hybrid approach [97], analyzing a packaged microstrip via-hole icluding coax-to-microstrip transitions, combines TLM for discretizing the planar structure with modal approach for a modal time-domain field representation inside the regular package.

In [91], the authors propose TLM for closed homogeneous waveguides and mode matching

for the absorbing boundaries. The method is demonstrated for thick inductive irises in rectangular waveguide, showing excellent wide-band match for all excited modes. But, in order to avoid the large amount of computing resources, the compromise done is reducing the discretized space by setting the absorbing boundaries a few cells away from the discontinuities and considering only the first 4 or 5 modes.

4.3.2 TLM/Modal coupling matrix

The calculation of the electromagnetic structures with three dimensional large geometries inside and very fine detail discontinuities is a big challenge for industry and accademia alike. The space and time electromagnetic modeling of such structures with a single numerical method, requires fine discretization to make possible the description of the field around the elements with very small gaps.

This would be impossible to calculate the field with a single numerical method because this one often leads to heavy computations that require significant computing resources. With a hybrid computing tool, a complex structure is segmented into subdomains, which by their nature can be discretized for numerical computation or described by analytical approach.

The realization of a hybrid algorithm coupling the time-domain TLM method and a modal approach, represents a solution to this problem which is implemented in this research work. Modeling a structure with two numerical methods, one temporal and one modal, means decomposing the problem into two substructures: a subdomain discretized with TLM and the other modeled with the modal approach.

The hybrid algorithm combining TLM method and modal approach takes the advantages of the two methods to get the most efficient solution to the electromagnetic problem to be analyzed.

A subdomain is defined in a reference plane by its impulse response in time or frequency domain. The reference plane separates two subdomains: one which is discretized in time and space and one modeled analytically. Electromagnetic field is represented in the reference plane by means of a modal basis, if a uniform waveguide section is considered.

If TLM is applied to calculate the electromagnetic field in a discretized volume, modal approach models the field of planar structures by decomposing it into linear combinations of TE and TM modes. Thus, the subdomain which is analytically modeled by modal approach is reduced from three dimensions to one dimension, being represented by modal transmission lines which do not couple together.

To have a representative solution of the entire structure to be modeled, it is necessary joining the two subdomains with modal diakoptics. Subdomains are computed independently and connected together at each time step by convolution product between the time-domain impulse response of the subdomain discretized with TLM and the numerical Green's func-

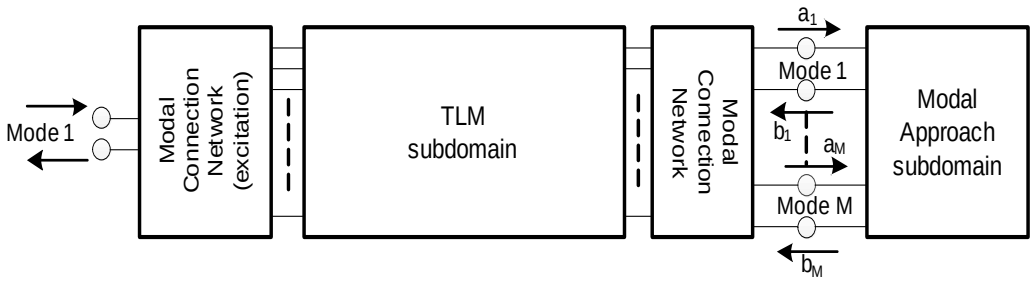


Figure 4.8: Schematic view of the domain decomposition.

tions for the subdomain calculated analytically with the modal approach. A general view over the TLM/modal hybrid approach can be observed in Fig. 4.8.

The unterminated boundaries of the discretized TLM domain are the reference planes. These reference planes are connected with a Modal Connection Network (MCN) to the modal representation of the electromagnetic field.

Each SCN node intersects the reference plane with the branch which is parallel to the direction of propagation, as in Fig. 4.5. This branch has two transmission lines which do not couple together, each one representing a polarization of the field, vertical or horizontal in the transverse plane as in Fig. 4.9. Thus, the TLM port number of the MCN is equal to the number of SCN points in the reference plane multiplied by two.

$$\begin{aligned} V_x^i(x,y,t) &= \tfrac{\Delta l}{2}[E_x^i(t) + Z_0 H_y^i(t)] \\ &= \tfrac{\Delta l}{2}\cos\left(\tfrac{m\pi}{a}x\right)\sin\left(\tfrac{n\pi}{b}y\right)\left(\tfrac{j\omega}{k_c^2}\tfrac{n\pi}{b}\mu_0 H_0 - \tfrac{\gamma}{k_c^2}\tfrac{m\pi}{a}E_0 + Z_0\left(\tfrac{\gamma}{k_c^2}\tfrac{n\pi}{b}H_0 - \tfrac{j\omega}{k_c^2}\tfrac{m\pi}{a}\epsilon_0 E_0\right)\right) \\ V_y^i(x,y,t) &= \tfrac{\Delta l}{2}[E_y^i(t) + Z_0 H_x^i(t)] \\ &= \tfrac{\Delta l}{2}\sin\left(\tfrac{m\pi}{a}x\right)\cos\left(\tfrac{n\pi}{b}y\right)\left(-\tfrac{j\omega}{k_c^2}\tfrac{m\pi}{a}\mu_0 H_0 - \tfrac{\gamma}{k_c^2}\tfrac{n\pi}{b}E_0 - Z_0\left(\tfrac{\gamma}{k_c^2}\tfrac{m\pi}{a}H_0 - \tfrac{j\omega}{k_c^2}\tfrac{n\pi}{b}\epsilon_0 E_0\right)\right) \end{aligned} \tag{4.15}$$

$$\begin{aligned} E_x^i &= E_x^{TE} + E_x^{TM} \\ &= \left(\tfrac{j\omega}{k_c^2}\tfrac{n\pi}{b}\mu_0 H_0 - \tfrac{\gamma}{k_c^2}\tfrac{m\pi}{a}E_0\right)cos\left(\tfrac{m\pi}{a}x\right)\sin\left(\tfrac{n\pi}{b}y\right) \\ E_y^i &= E_y^{TE} + E_y^{TM} \\ &= \left(-\tfrac{j\omega}{k_c^2}\tfrac{m\pi}{a}\mu_0 H_0 - \tfrac{\gamma}{k_c^2}\tfrac{n\pi}{b}E_0\right)sin\left(\tfrac{m\pi}{a}x\right)\cos\left(\tfrac{n\pi}{b}y\right) \\ H_x^i &= H_x^{TE} + H_x^{TM} \\ &= \left(\tfrac{\gamma}{k_c^2}\tfrac{m\pi}{a}H_0 + \tfrac{j\omega}{k_c^2}\tfrac{n\pi}{b}\epsilon_0 E_0\right)sin\left(\tfrac{m\pi}{a}x\right)\cos\left(\tfrac{n\pi}{b}y\right) \\ H_y^i &= H_y^{TE} + H_y^{TM} \\ &= \left(\tfrac{\gamma}{k_c^2}\tfrac{n\pi}{b}H_0 - \tfrac{j\omega}{k_c^2}\tfrac{m\pi}{a}\epsilon_0 E_0\right)cos\left(\tfrac{m\pi}{a}x\right)\sin\left(\tfrac{n\pi}{b}y\right) \end{aligned} \tag{4.16}$$

The incident and reflected voltage waves for both polarizations have to be transformed in modal waves in order to be related with the modal subdomain through the time-domain modal impulse response of this latter. The incident TLM link-line voltages for both polar-

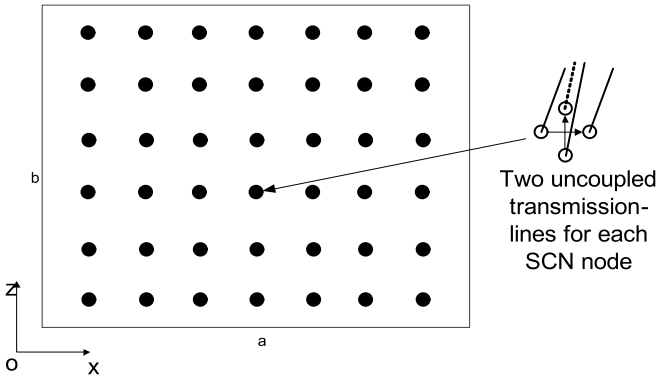


Figure 4.9: SCN nodes on reference plane.

izations are given by (4.15), where the incident transversal field components are formulated in (4.16).

The MCN converts the TLM representation of the electromagnetic field in the form of voltage impulses into the modal form described by the modal basis defined on the waveguide cross section contained in the reference plane. The MCN can be represented by a transformer network [100] as shown in Fig. 4.10, or equivalently, by a coupling matrix that converts the electromagnetic modal field to the TLM representation. This TLM / modal coupling matrix is built on two elements: SCN-TLM nodes coordinates used to discretize the space of the waveguide cross section and the mode eigenfunctions describing the field in that section.

These voltage pulses of the transmission lines from the discretized subdomain are converted into the corresponding voltage of the modes from the modal basis, by means of the network outlined in Fig. 4.10, characterized by the matrix Q,

$$Q = \begin{bmatrix} q_{11} & q_{12} & q_{13} & \cdots & q_{1N} \\ q_{21} & q_{22} & q_{23} & \cdots & q_{2N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ q_{M1} & q_{M2} & q_{M3} & \cdots & q_{MN} \end{bmatrix} \tag{4.17}$$

where, q_{ij} are the network transformer ratios based on the eigenfunctions of the modes described in the modal basis. SCN nodes polarizations of the two ports normal to the reference plane impose the choice of the eigenfunctions forming the matrix Q.

A set of Gaussian pulses excites the modes of the MCN at the excitation plane. The MCN transforms these pulses to the TLM form and so excites the TLM link lines. The free space impedance has been used as the reference impedance of all ports of the MCN network.

This network, presented in [100], is lossless and consequently has the property that every mode not represented by the modal decomposition but present on the TLM link lines will be reflected back to the TLM domain. To relax this effect, as many modes have been used in the domain decomposition as there are TLM link lines. However, all the non-considered higher

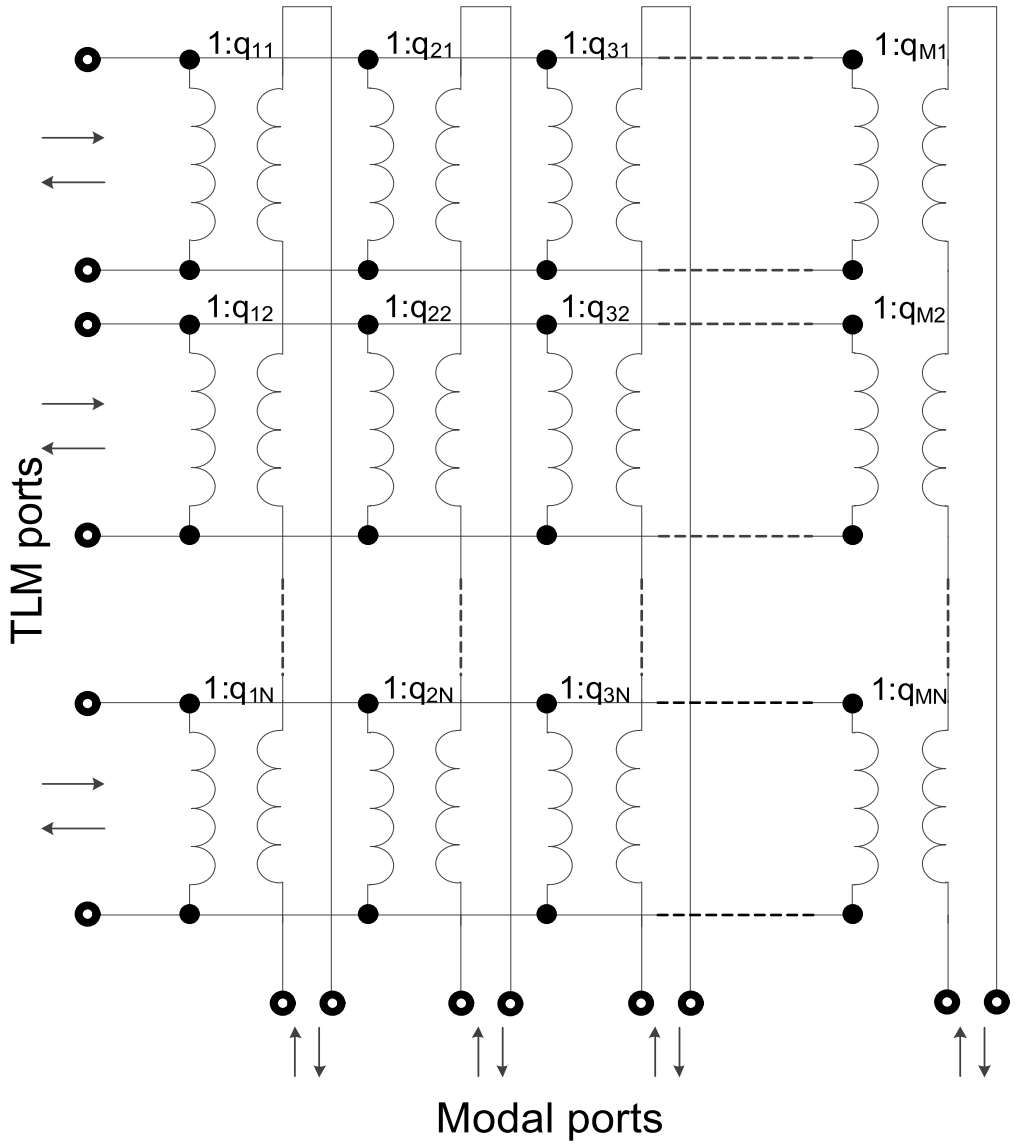


Figure 4.10: TLM/modal connection network.

order modes are terminated with the free space impedance - which is the high frequency limit of the characteristic impedance of each mode. This is not an ideal termination, as the real frequency-dependent characteristic mode impedance, but it relaxes the problem of reflections. If the higher order modes are not present at the reference planes, the terminating impedances are not excited and they play no role.

4.3.3 Generation of the analytical modal time-domain Green's functions

The reflected-mode field amplitudes are computed as time-domain convolution product between the incident-mode field amplitudes and the modal impulse response Johns matrix, which is formed by the time-domain modal Green's functions.

Green's functions, characterizing the modal domain, are generated analytically as S parameters in frequency domain, deriving from the free space impedance used as the reference impedance of modal ports of the MCN network and the different TE and TM modes impedances.

Johns matrix elements have to be transformed in time domain in order to achieve the convolution product at each TLM iteration. The causality of the time-domain signals in the Inverse Fourier Transform (IFT) is enforced by imposing symmetry on the magnitude and antisymmetry on the phase of S parameters. Depending on the modal domain termination complexity, the computational effort of the Johns matrix increases as the modes couple each other.

4.3.4 TLM convolution product

The convolution process can be written in discrete form as:

$$y[n] = x[n] * f[n] = \sum_{i=0}^{T-1} x[i] f[n-i], \tag{4.18}$$

where *y[n]* is the TLM input data in a modal form, *x[n]* is the TLM output data in a modal form, *f[n]* is the time-domain impulse response of the modal domain and T is the number of iterations of the simulation.

The convolution process is executed during the TLM simulation, for each time-step. In fact, while the samples of the TLM output data are known step by step, the samples of the impulse response *f[n]* are fully known. Thus, each sample of the TLM output data when is catched, is used to compute his contribution for all the TLM input data samples.

When dealing with a structure similar as in the Fig. 4.8, the reflected modes field

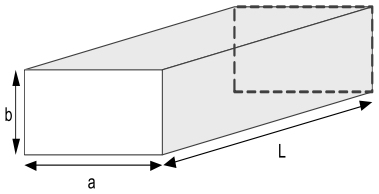


Figure 4.11: Hollow waveguide.

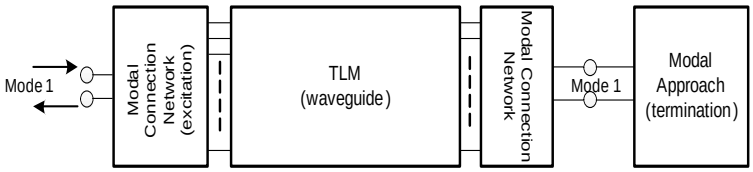


Figure 4.12: General view over the structure modeled upon the hybrid approach.

amplitude is given by:

$$\begin{aligned} b_1 &= a_1 * S_{11} + a_2 * S_{12} + \cdots + a_M * S_{1M} \\ b_2 &= a_1 * S_{21} + a_2 * S_{22} + \cdots + a_M * S_{2M} \\ &\vdots \\ b_M &= a_1 * S_{M1} + a_2 * S_{M2} + \cdots + a_M * S_{MM}, \end{aligned} \tag{4.19}$$

where * represents the convolution product. If the termination of the modal domain does not couple the modes describing this domain, the elements S_{ij}, $i \neq j$, are zero.

4.3.5 Numerical examples

To validate the coupling between TLM and Mode Matching, it is simulated the propagation of TE_{10} mode in a rectangular waveguide with electric walls, filled with vacuum, with the dimensions $a = 10$ mm, $b = 5$ mm and $L = 25$ mm as in Fig. 4.11.

The volume of the structure is discretized with TLM while the termination is modeled with the modal approach. The analysis of the propagation of the electromagnetic field is conducted for different terminations of the waveguide.

First scenario - the guide is matched.

A general view over the hybrid approach can be observed in Fig. 4.12. The impulse response of the termination is obtained analytically in frequency domain by the reflection

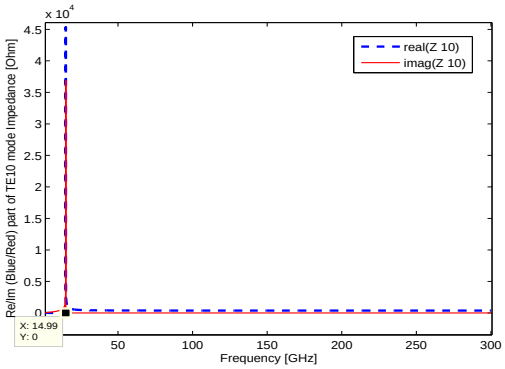


Figure 4.13: TE_{10} mode characteristic impedance.

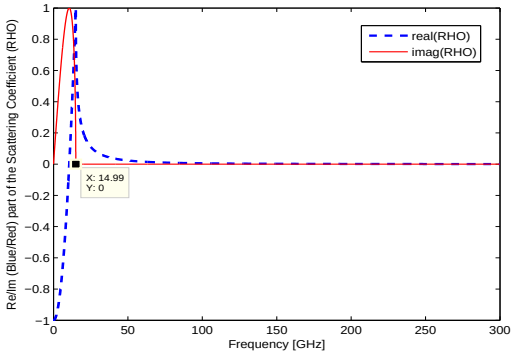


Figure 4.14: Frequency-domain representation of the Scattering Coefficient Γ.

coefficient formula (4.20) and is given in Fig. 4.14.

$$\Gamma = \frac{Z^{TE_{10}} - Z_o}{Z^{TE_{10}} + Z_o} \tag{4.20}$$

$Z^{TE_{10}}$ is the fundamental mode impedance shown in Fig. 4.13 and Z_o is the free space impedance seen from the modal port of the MCN network.

To achieve the causality of the time-domain signals in the IFT, the transformation is applied to a reflection coefficient prepared before by imposing symmetry on the magnitude and antisymmetry on the phase, shown in Fig. 4.15. The time-domain impulse response of the subdomain simulating an infinite waveguide is given in Fig. 4.16.

The time-domain signals as the waveguide excitation and the input reflected mode-field are depicted in Fig. 4.17(a). This latter is attenuated due to the matched waveguide termination. The input S and Z parameters are shown in Fig. 4.17(b) and respectively Fig. 4.17(c). Above the cutoff frequency of the fundamental mode at 15 GHz, the wave propagates toward the termination modeled as a perfect ABC. The curves validate the hybrid approach being in excellent agreement with the analytical results.

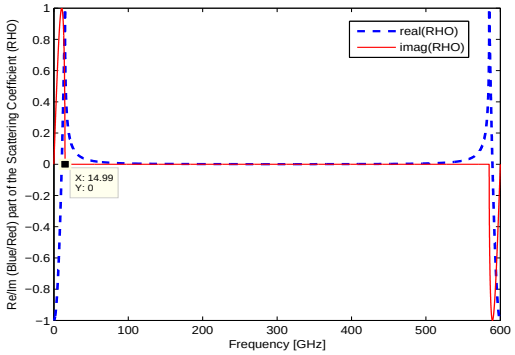


Figure 4.15: Frequency-domain representation of the Scattering Coefficient Γ prepared for IFFT.

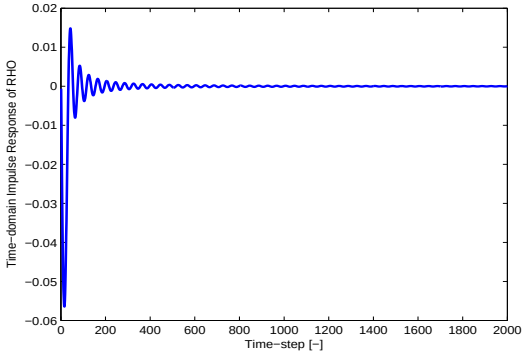
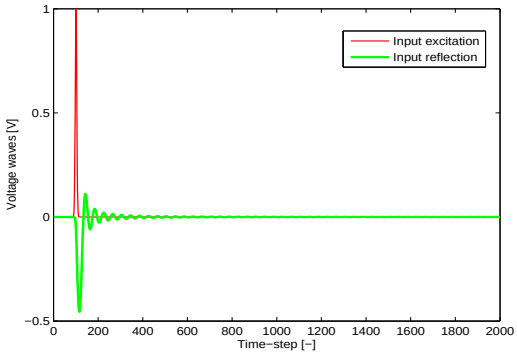
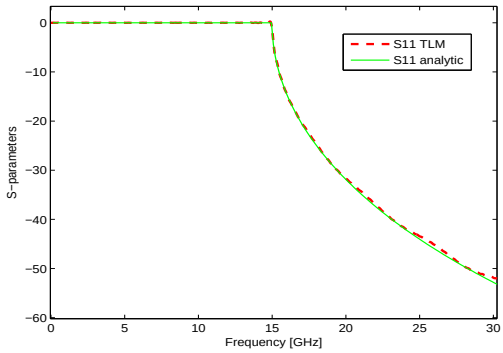


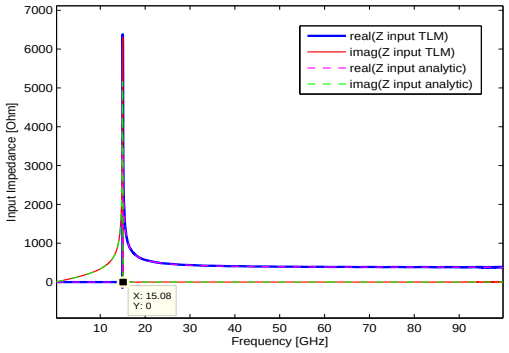
Figure 4.16: Time-domain impulse response of the Scattering Coefficient Γ.



(a) Time-domain signals at the input port.



(b) Input reflection coefficient, S_{11}, TLM model computation (red dashed curve) vs. analytic computation (green curve).

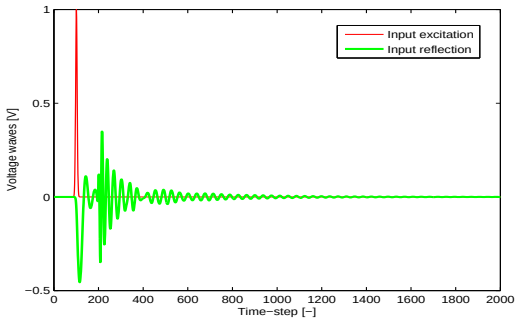


(c) Input impedance, Z_{11}, TLM model computation (real part - blue curve, imaginary part - red curve) vs. analytic computation (real part - magenta dashed curve, imaginary part - green dashed curve).

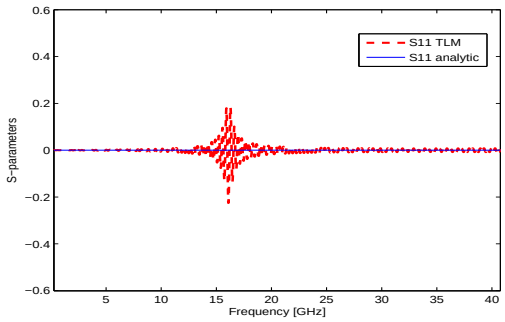
Figure 4.17: Results for the TE_{10} mode propagation in a matched waveguide.

Second scenario - the waveguide is short-circuited.

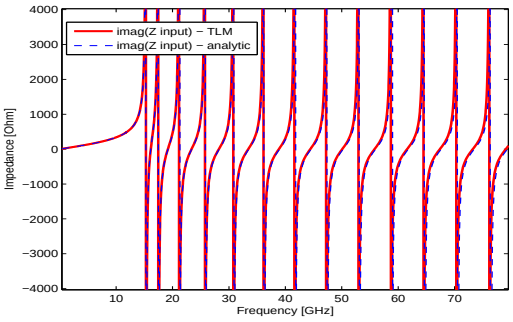
A similar procedure to the matched case it is performed in order to determine the time-domain impulse response of the termination. In Fig. 4.18(a), the time-domain signals as the waveguide excitation and the input reflected mode-field are shown.



(a) Time-domain signals at the input port.



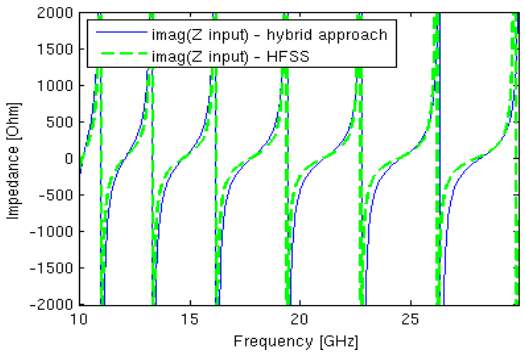
(b) Input reflection coefficient, S_{11}, TLM model computation (red dashed curve) vs. analytic computation (blue curve).



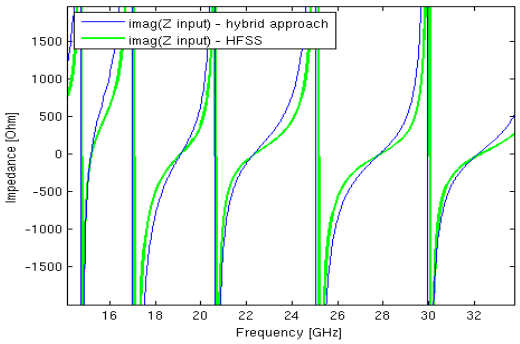
(c) Input impedance, Z_{11}, TLM model computation (imaginary part - red curve) vs. analytic computation (imaginary part - blue dashed curve).

Figure 4.18: Results for the TE_{10} mode propagation in a short-circuited waveguide.

This latter is reflected due to the waveguide termination. The input S and Z parameters are presented in Fig. 4.18(b) and respectively in Fig. 4.18(c). Numerical errors appear around the cutoff frequency. The curves validate the hybrid approach being in excellent



(a) The waveguide is filled with dielectric.



(b) The waveguide is filled with air and a block of dielectric.

Figure 4.19: Input impedance (imaginary part) of a short-circuited lossless waveguide with dielectric $\epsilon_r = 2.54$.

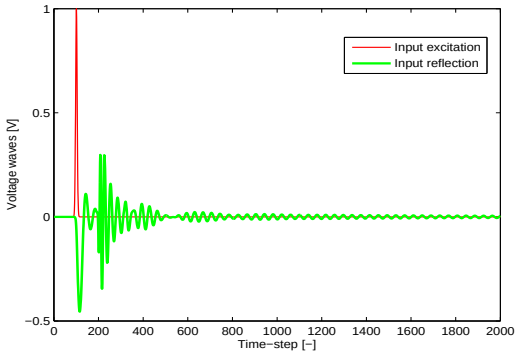
agreement with the analytical results.

Also, in this scenario the waveguide is filled with dielectric medium of ϵ_r=2.54, modeled with stubs as in (4.10), (4.11) added to the TLM-SCN nodes whose Scattering matrix is obtained by means of (4.13). The validation is given by the input Z-parameter as in Fig. 4.19(a). The input impedance computed with the hybrid approach follows the result obtained by High Frequency Structural Simulator (HFSS) [50].

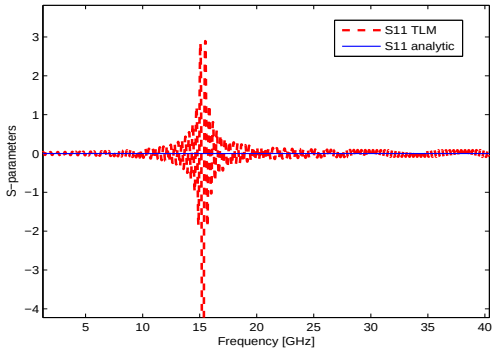
The TLM/modal hybrid approach is validated for non-homogeneous volumes by the simulation with the waveguide filled with air and having a centered block of dielectric ϵ_r=2.54, with the size of 4 mm width, 3 mm height, 3 mm length. In Fig. 4.19(b), the imaginary part of the input Z-parameter of this waveguide is in excellent agreement with the curve given by HFSS.

Third scenario - the guide is open-circuited.

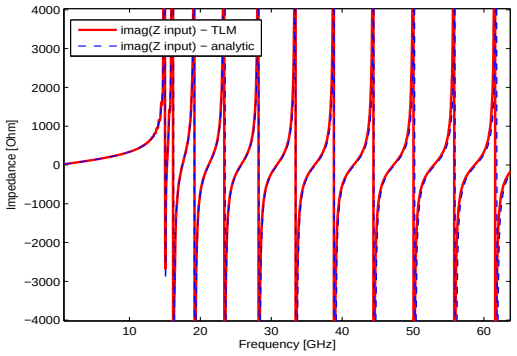
Also for this case, in Fig. 4.20(a), the time-domain signals as the waveguide excitation and the input reflected mode-field are depicted. This latter is reflected due to the waveguide termination. The input S and Z parameters are shown in Fig. 4.20(b) and respectively Fig. 4.20(c). The curves validate the hybrid approach being in excellent agreement with the analytical results.



(a) Time-domain signals at the input port.



(b) Input reflection coefficient, S_{11}, TLM model computation (red dashed curve) vs. analytic computation (blue curve).



(c) Input impedance, Z_{11}, TLM model computation (imaginary part - red curve) vs. analytic computation (imaginary part - blue dashed curve).

Figure 4.20: Results for the TE_{10} mode propagation in an open-circuited waveguide.

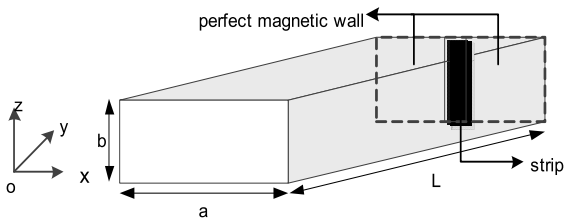


Figure 4.21: Hollow waveguide terminated by a metallic strip printed on a perfectly magnetic wall.

Forth scenario - the waveguide is terminated by a metallic strip.

In this case, a length of lossless waveguide with electric walls, filled with vacuum, with the dimensions $a = 10$ mm, $b = 5$ mm and $L = 25$ mm as in Fig.4.21 is terminated by a narrow metallic strip (1mm width) printed on a perfectly magnetic wall. The TLM mesh step is 0.2 mm. When TE_{10} is incident, the strip excites an infinite number of modes, propagating or evanescent coupling each other, of the form $TE_{(2n+1)0}$, $n = 0, 1, 2, ...$, given the uniformity of the discontinuity along z axis.

The modal basis constructed in the discontinuity plan is defined as follows:

$$\mathbf{f}_n^{TE} \equiv \left[\sqrt{\frac{2}{a}} \cos(2n+1)\frac{\pi}{a}x \right]^{0}, \tag{4.21}$$

where n=0,1,2,...; $\mathbf{f}_0$ is the fundamantal mode TE_{10} .

The modal basis is formed exclusively by TE modes and is orthonormal with respect to the hermitian scalar product(4.22):

$$\langle \mathbf{f}_m | \mathbf{f}_n \rangle = \int_D \mathbf{f}_m^* \mathbf{f}_n dS = \int\int_{D-\Sigma} \mathbf{f}_m^* \mathbf{f}_n dS + \int\int_{\Sigma} \mathbf{f}_m^* \mathbf{f}_n dS = \int\int_{\Sigma} \mathbf{f}_m^* \mathbf{f}_n dS = \delta_{mn} \tag{4.22}$$

where:

$$\delta_{mn} = \begin{cases} 1 & \text{if } m{=}n \\ 0 & \text{if } m \neq n \ , \end{cases}$$

* is the complex conjugate,
Σ is the discontinuity surface and D is the transverse plane containing the discontinuity.

The field through that surface is modeled by virtual sources (which do not produce power in the context of the boundary conditions), according to [15]. The evanescent modes excited by the discontinuity represents electromagnetic energy which is primarily inductive, which

accumulates in the vicinity of the obstacle. The multi-modal surface admittance matrix modeling the non-homogeneous discontinuity is given by :

$$[Y] = \frac{1}{\left\langle \mathbf{g}_e | \widehat{Z} \mathbf{g}_e \right\rangle} \begin{bmatrix} \langle \mathbf{g}_e | \mathbf{f}_1 \rangle^2 & \langle \mathbf{g}_e | \mathbf{f}_1 \rangle \langle \mathbf{g}_e | \mathbf{f}_2 \rangle & \dots & \langle \mathbf{g}_e | \mathbf{f}_1 \rangle \langle \mathbf{g}_e | \mathbf{f}_M \rangle \\ \langle \mathbf{g}_e | \mathbf{f}_2 \rangle \langle \mathbf{g}_e | \mathbf{f}_1 \rangle & \langle \mathbf{g}_e | \mathbf{f}_2 \rangle^2 & \dots & \langle \mathbf{g}_e | \mathbf{f}_2 \rangle \langle \mathbf{g}_e | \mathbf{f}_M \rangle \\ \vdots & \vdots & \ddots & \vdots \\ \langle \mathbf{g}_e | \mathbf{f}_M \rangle \langle \mathbf{g}_e | \mathbf{f}_1 \rangle & \langle \mathbf{g}_e | \mathbf{f}_M \rangle \langle \mathbf{g}_e | \mathbf{f}_2 \rangle & \dots & \langle \mathbf{g}_e | \mathbf{f}_M \rangle^2 \end{bmatrix} \tag{4.23}$$

where, M is the number of propagating modes excited by the metallic strip, g_e is an entire-domain trial function used in Galerkins method for modeling the current density on the metallic strip, f_n is the n^{th} basis function of the normal modal basis in the waveguide and the operator $\widehat{Z}$, defined as:

$$\widehat{Z} = \sum_{n=M+1}^{\infty} |\mathbf{f}_n\rangle \, Z_n \, \langle \mathbf{f}_n| \tag{4.24}$$

represents the contribution of the evanescent modes excited by the discontinuity, to the field that is in the vicinity of the structure. Z_n is the impedance of the evanescent mode $\mathbf{f}_n$.

The admittance matrix can also be formulated as:

$$[Y] = \frac{1}{\left\langle \mathbf{g}_e | \widehat{Z} \mathbf{g}_e \right\rangle} \begin{bmatrix} \langle \mathbf{g}_e | \mathbf{f}_1 \rangle \\ \langle \mathbf{g}_e | \mathbf{f}_2 \rangle \\ \vdots \\ \langle \mathbf{g}_e | \mathbf{f}_M \rangle \end{bmatrix} \begin{bmatrix} \langle \mathbf{g}_e | \mathbf{f}_1 \rangle & \langle \mathbf{g}_e | \mathbf{f}_2 \rangle & \dots & \langle \mathbf{g}_e | \mathbf{f}_M \rangle \end{bmatrix} \tag{4.25}$$

We make the notation:

$$[A] = \begin{bmatrix} \langle \mathbf{g}_e | \mathbf{f}_1 \rangle \\ \langle \mathbf{g}_e | \mathbf{f}_2 \rangle \\ \vdots \\ \langle \mathbf{g}_e | \mathbf{f}_M \rangle \end{bmatrix} \tag{4.26}$$

So, [A] transposed is:

$$[A]^T = \begin{bmatrix} \langle \mathbf{g}_e | \mathbf{f}_1 \rangle & \langle \mathbf{g}_e | \mathbf{f}_2 \rangle & \dots & \langle \mathbf{g}_e | \mathbf{f}_M \rangle \end{bmatrix} \tag{4.27}$$

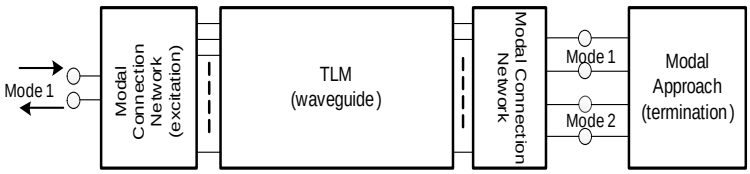


Figure 4.22: General view over the structure modeled upon the hybrid approach.

Finally, the admittance matrix of the discontinuity is:

$$[Y] = \frac{1}{\sum_{n=N+1}^{\infty} Z_n \langle \mathbf{g}_e | \mathbf{f}_n \rangle^2} [A][A]^T \tag{4.28}$$

The M-port representing the modal subdomain - see Fig. 4.8 - is lossless and reciprocal; thus, the admittance and impedance matrices contain only pure imaginary terms and are symmetric. The impedance matrix Z represents the inverse of the admittance matrix Y.

In order to connect the discontinuity and the waveguide, the scattering matrix S is needed. As there is no power actually absorbed by the M-port, the scattering matrix is unitary. Due to reciprocal network, S is also symmetric. According to the admittance matrix from (4.28), the scattering matrix is given by:

$$[S] = ([Y_0] - [Y])([Y_0] + [Y])^{-1} \tag{4.29}$$

where $[Y_0]$ is the diagonal matrix of the inverse free-space impedance. The amplitude of the reflected modes is calculated upon (4.19).

According to Fig. 4.22, it is considered the excitation of the only two first lower-order TE propagating modes and 300 evanescent modes.

The computed 2x2 input scattering matrix -see Fig. 4.23- and input impedance matrix -see Fig. 4.24- are in excellent agreement with results derived from analytical approach based on IE Formulation [81], on the two modes frequency band.

4.4 Conclusions

A not exhaustive presentation of the numerical methods TLM and Mode matching is exposed in this chapter. The numerical tool developped based on the hybridization of the two numerical methods is described. This model allows full-wave simulation of complex structures, in three dimensions containing discontinuities, by saving computational resources. A simulation of the entire computation domain with a rigorous method as TLM would require significant computing resources. This hybrid decomposes the computation domain into two subdomains, based on diakoptics approach: volumes are discretized with TLM and pla-

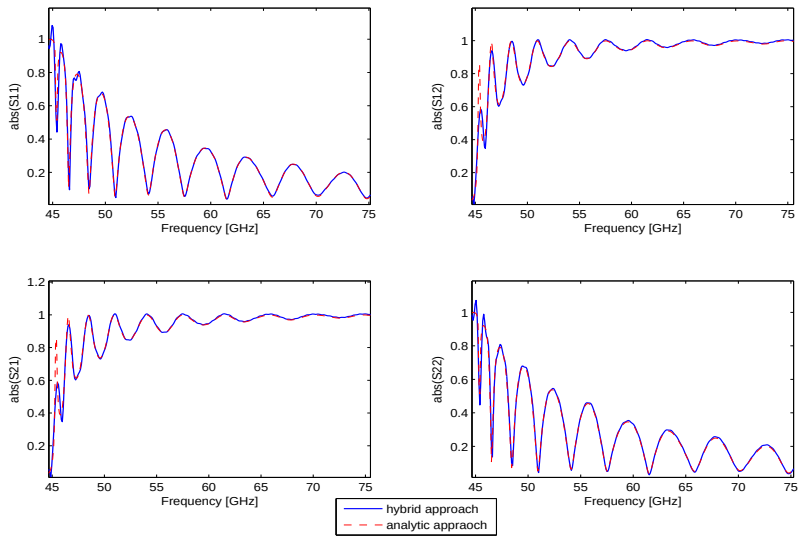


Figure 4.23: Input scattering matrix elements (absolute value) of a lossless waveguide, terminated by a metallic strip printed on a perfectly magnetic wall (1 denotes the TE_{10} mode while 2 denotes the TE_{30} mode); hybrid model computation (blue curve) vs. analytic computation (red dashed curve).

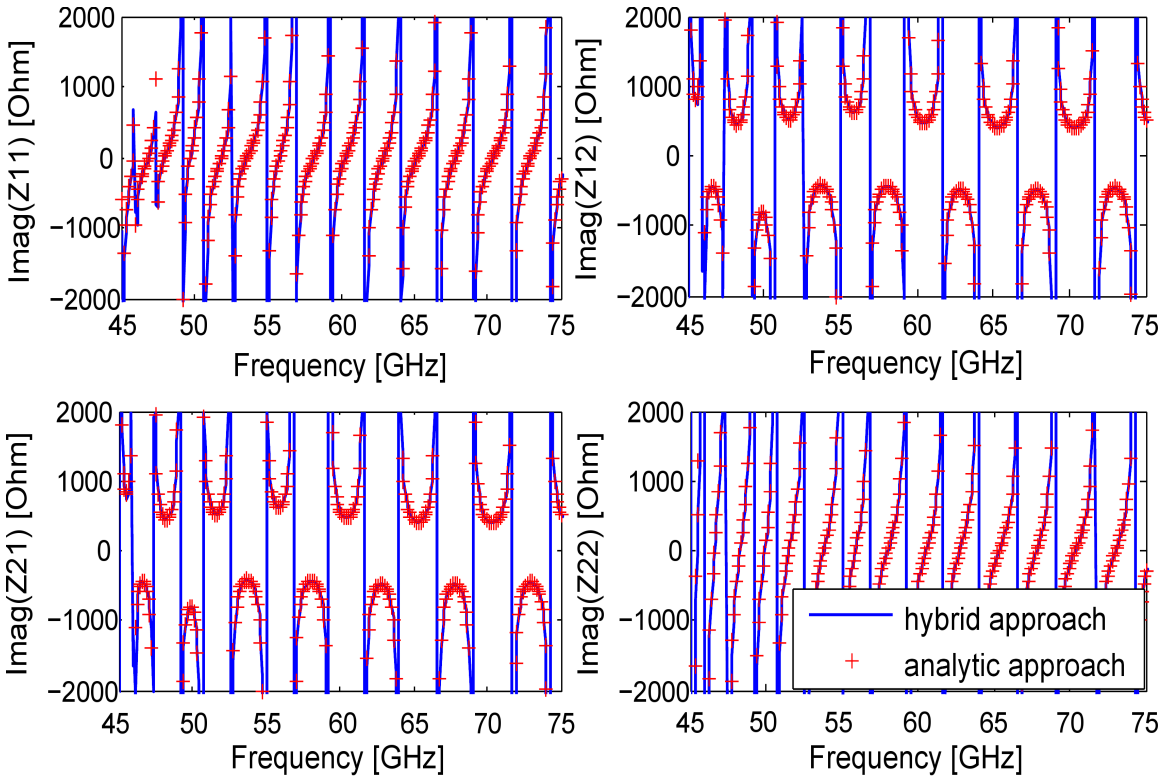


Figure 4.24: Input impedance matrix elements (imaginary part) of a lossless waveguide, terminated by a metallic strip printed on a perfectly magnetic wall (1 denotes the TE_{10} mode while 2 denotes the TE_{30} mode); hybrid model computation (blue curve) vs. analytic computation (red curve).

nar structures are modeled by modal approach. The link between the two subdomains is based on integral equation approach by binding the tangential electromagnetic field of the TLM subdomain to the interface between the two subdomains with the propagating modes of the modal subdomain. This coupling is achieved by convolution product between the incident-mode field amplitude on modal subdomain and the modal impulse response of the latter. The hybrid approach is validated by propagating TE_{10} mode in a rectangular homogeneous waveguide terminated by homogeneous and non-homogeneous loads. The results are in excellent agreement with the analytical curves. Also, results obtained for heterogeneous environment modelled with stubs are validated with HFSS.

Chapter 5

Parallel TLM/Modal approach

This chapter deals with the electromagnetic modeling of large and complex electrical structures by means of large scale parallel systems, such as GC and supercomputer. Grid5000 - the French national research infrastructure for large scale parallel and distributed computing - is the grid platform used in this work to run large TLM applications. The supercomputer that houses the experiments in this research work is Hyperion, a supercomputer belonging to Calmip group.

The implementation of the TLM parallel algorithm for distributed computing systems is presented. Numerous experiments for parallel computing using the TLM/modal hybrid numerical approach described in the previous chapter, are executed on both, cluster and grid computing. The application performance obtained by parallel computing is analyzed. Also, some aspects related to the application implementation and computing platform which directly influence the computing performance, such as data precision, communication modes between processes, memory contention, cache misses are outlined.

Results comparable to those obtained on grid computing are realized also on supercomputer. The results prove the benefits of the grid computing and supercomputer environments to solve electrically large structures. A prediction model for the computation time on grid is developped step by step in order to improve the accuracy of the estimation.

5.1 Context

In [9], parallel computing is presented as a useful tool for solving large problems faster. The results in terms of parallel computing speedup show the importance of a good match between the problem to calculate and the number of allocated computing resources. TLM has been implemented on massively parallel SIMD computers [113]. The effects on computing performance when TLM calculation is distributed across a network of workstations using PVM, are analysed in [90]. In order to illustrate the improvements in computational electromagnetics that are achievable by MPP, a parallel finite element code is used to model a

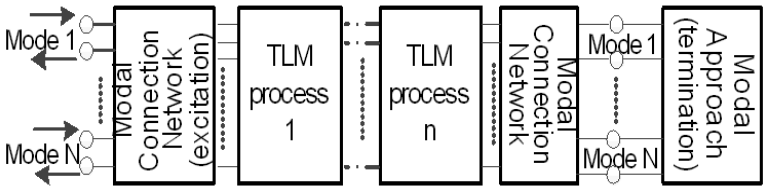


Figure 5.1: Schematic view of the modeling approach.

low frequency magnetics problem [16]. Processors communicate by passing messages.

The Research Institute of Electronic Science and Technology from Chengdu, China, provides a numerical tool for the parallel computation on grid, of the electromagnetic scattering fields in large complex environments, based on method of moment [72]. A guide [35] introducing CEM researchers in the field of computational grid, demonstrates that GC is a viable environment for parallel computing using FDTD in a distributed programming paradigm, and also that is an effective way to produce lowcost and flexible cooperative and distributed engineering on computer-aided engineering (CAE) of aperture-array antennas. Another grid experiment [73] based on TLM modeling code computes highly complex electromagnetic structures with a good scalability and an optimal performance in terms of computation time by adapting the distributed resources to the given problem.

5.2 Parallel TLM/modal hybrid approach

5.2.1 Parallel algorithm

In order to avoid a heavy TLM calculation, the discretized domain can be divided into several sub-domains, as in Fig. 5.1 that will be computed in parallel on multiple CPUs that communicate between them to achieve the job. The parallel hybrid approach, based on MPI, is designed for Single Program Multiple Data (SPMD) programming model. Thus, all the tasks run the same program on different data. The user specifies the number of subdomains the structure will be decomposed, i.e. the number of processes MPI. Each subdomain is attributed to a process. Each process receives a copy of the program.

Communication between these processes is a point-to-point communication, based on sending and receiving messages. Dividing the whole domain discretized into several subdomains, each one belonging to one MPI process, makes the processes communicate between them since the first iteration, even if the messages do not contain information representing the electromagnetic field, which is not yet propagated.

A schematic view of the parallel hybrid approach implementation can be seen in Fig. 5.2. On each process, the substructures are discretized and the boundary conditions are imposed. The excitation is performed only by the first task, while the termination is computed by the last process. At each time step, the TLM's core formed by the steps *Scattering* and

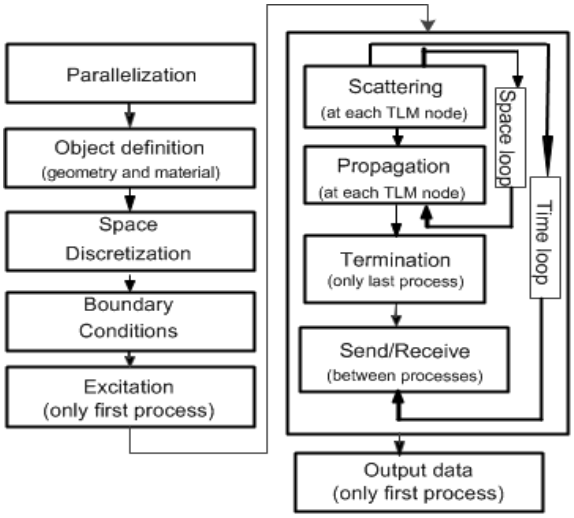


Figure 5.2: Schematic view of the parallel hybrid approach implementation.

Table 5.1: Summary of Computed Structures

No.	Millions of TLM cells	X	Y	Z	t
1	145	345	173	2432	10000
2	36	172	86	2432	10000
3	290	345	173	4864	19610
4	580	345	173	9728	39061
5	194	400	200	2432	10000
6	1400	400	200	17500	70549

Propagation is applied to the discretized cells. Before moving to the next time step, each process communicates to his neighboring processes the intermediate simulation data.

5.2.2 Experiments

In order to evaluate the performance of our application on a large scale parallel system in terms of computation time, simulations have been performed on cluster and on grid environment using different rectangular waveguide structures discretized with a mesh step of 1mm, matched at their terminations, outlined in Table 5.1, where X, Y, Z are the number of TLM cells on the three spatial directions and t is the number of time steps.

Blocking vs non-blocking communications:

In Table 5.2, the values of the simulation time for the structure number 1 from Table 5.1, are displayed. Computing nodes from cluster Griffon, whose communication through an ethernet network (1 and 10 Gb/sec) is managed by blocking or non-blocking MPI primitives, are used. It can be observed a maximum gain of 10% for the simulation time in case of non-blocking communication. The difference obtained between the simulation time values of the two types of MPI communications is not very high. The gain can even be altered

Table 5.2: Blocking vs Non-Blocking MPI Communications on Cluster

No. of nodes	No. of processes	Time [h] non-block. com.	Time [h] block. com.	Gain [%]
1	1	46.13	46.13	0
2	4	13.8	13.79	-0.07
4	8	7.08	7.1	0.31
8	16	3.52	3.54	0.48
16	32	1.44	1.47	2.5
32	64	0.69	0.77	10.53
64	128	0.37	0.39	6.79
87	174	0.3	0.29	-4.18
92	184	0.29	0.27	-6.94

sometimes, taking into account that the cluster network is also used by other users at the same time.

The blocking send call (MPI_Send) uses the synchronous mode. The process restarts the computation once the entier sender buffer was sent and a matching receive is posted while the neighbor process has started to receive the message. The non-blocking send call (MPI_Isend) uses the standard mode. Once the send operation is started, the process restarts also the computation. As soon as the message has been sent, the send buffer is refilled. The receive call uses a blocking primitive in a standard mode (MPI_Recv).

When the process receives data from his neighbor, it is required to stop the computation operation until the message is fully received. When the computation starts, the message must be fully received, because it is used in complete form. The message contains the voltage impulses representing the two polarizations of each TLM cell lying on the cross section which separates two different discretized subdomains belonging two different processes. Given the nature of this application, the gain obtained by reducing the simulation time when the communication between processes is controlled by non-blocking primitives, is not important.

Time, speedup and efficiency for cluster experiments:

Fig. 5.3 displays the simulation times on cluster Griffon versus the number of processes, for various numbers of TLM cells, considering the first four structures from Table 5.1. Only two processes are executed on the computing node, each one on a different processor. As we increase the number of processes, the simulation time becomes smaller.

The simulation time of the *structure 1* in one process is about 46 hours, but using 184 processes, it takes only 16 minutes. The speedup is 164 and the efficiency is 89%. The computing and communication times of the *structure 2* are shorter, due to the lower number of TLM cells by which it is discretized. The speedup value is 181 while the efficiency, 98%.

In Table 5.3, the values for speedup end efficiency obtained for the simulations of these two

Table 5.3: Speedup and Efficiency on Cluster

	No. of processes								
	1	4	8	16	32	64	128	174	184
Structure 1									
Speedup	1	3.34	6.49	13.02	31.22	59.37	115.73	155.21	164.88
Efficiency [%]	100	83.5	81.1	81.3	97.5	92.7	90.4	89.2	88.6
Structure 2									
Speedup	1	3.28	6.68	16.88	35.06	68.64	130.02	170.21	181.71
Efficiency[%]	100	82	83.5	105.5	109.5	107.2	101.5	97.8	97.5

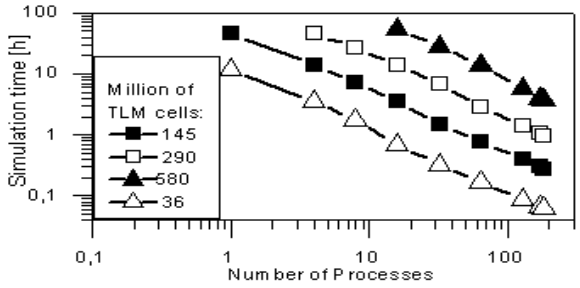


Figure 5.3: Simulation times on cluster for different TLM cell numbers.

structures are summarized. It can be observed that the efficiency has some discontinuities for the simulations done with a small number of processes. These discontinuities are due to the hierarchical organization of the memory present on the computing nodes and the large number of memory accesses which increase the computation time. This is especially important for the simulations with few processes where the discretized computation domain is higher, because if a sequential algorithm requires Q bytes of memory, it will occupy only Q / p bytes when p parallel processes are used. Sequential simulation being hardest hit by the cache misses, require a longer execution.

However, the memory access time for the sequential simulation is not so great to get a superlinear speedup. As the cache misses effect is strongly present in simulations, the efficiency decreases because the parallel simulation time consists of the computation time, the RAM access time and the communication time; then, the curve will slope upward, the parallel simulation time being reduced with the RAM access time. The efficiency decreases again as the number of processes is increasing due to communication costs, idle time due to synchronization and serial parts of the algorithm.

Time limitations and memory resources constraints do not allow us exploring the simulation times for the structures 3 and 4 with a small number of processes. These two structures keep the same transversal dimensions as the first one, but the lengths are multiplied by two, respectively four compared to the first one. Thus, the communication time is still constant, but the computing time is much larger. The simulation time for 290 million of TLM cells structure on 184 processes, is about one hour, and for 580 million of TLM cells the forth one is 3.7 hours. The curves show the scalability of our application as increasing the number of

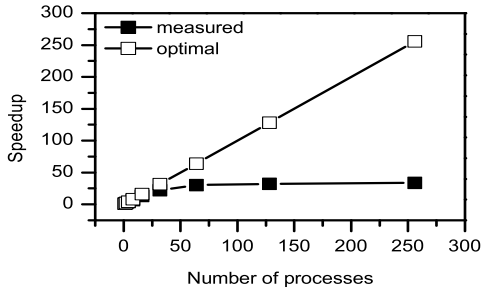


Figure 5.4: Parallel computation Speedup on shared memory multiprocessors - measured vs optimal values.

processes.

The computing time increases, almost linearly, when the numbers of TLM cells and time steps increase. The ratio between the number of TLM cells of the first and second structures equals the ratio of their computing times: this ratio is found to be 4 because the height and width of the *structure 1* are 2 times larger than ones of the *structure 2*. The same is available for the *structures 3* and *4*, because of the length and the number of time steps parameters. The number of TLM nodes on cross-sectional area of the waveguide represents the amount of transferred data between processes which influences the communication time.

Memory contention:

During several tests, we observed that the performance in terms of computing time drastically decreases when two or more processes run simultaneously on the same processor, which is related to the contention phenomena that appears to the shared memory multiprocessors. This is shown in Fig. 5.4, by means of the speedup curve obtained by the simulation of the *structure 1* from Table 5.1 using computing nodes from Chinqchint cluster (Lille) interconnected by ethernet 1 Gb/sec.

The speedup increases until the number of computing nodes reaches 32; each node having two CPUs, only two MPI processes are submited per node, each one on a different CPU. It can be observed that the speedup is very close to the optimal value. Given the limited number of resources on cluster, the number of processes launched per node has increased to eight, the nodes having 8 cores. For the simulations with 128 processes (16 nodes) and respectively 256 processes (32 nodes), the parallel computing performance is obviously affected.

The eight processes, simultaneously performing computation on the same node, suddenly increase the memory contention by shared memory bus saturation. The shared memory is the most important factor that limits the scalability of the parallel systems. The parallel system performance is cadenced by the memory latency and the bus memory bandwidth, even if the processor speed is high. In our experiment, the memory bus saturation is caused by the CPU demands and not by the Network Interface Controllers (NICs) that require

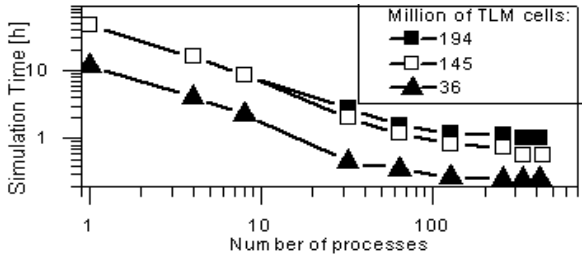


Figure 5.5: Simulation times on grid for different TLM cell numbers.

Table 5.4: Speedup and Efficiency on Grid

	No. of processes								
	1	4	8	32	64	128	256	332	422
Structure 1									
Speedup	1	2.8	5.3	22.9	38.4	55.1	61.9	80.6	81
Efficiency [%]	100	71.6	66.9	71.6	60	43	24.2	23.5	18.34
Structure 2									
Speedup	1	2.8	5.06	25.32	31.9	43.24	44.58	44.59	44.68
Efficiency[%]	100	70.67	63.34	79.14	49.85	33.78	17.41	13.35	10.58

memory bus even if there are only two processes per node to connect with other processes running on the computing nodes.

Time, speedup and efficiency for grid experiment:

The Fig. 5.5 displays the simulation times on grid versus the number of processes, for various numbers of TLM cells, considering the *structures 1,2* and *5* from Table 5.1. The computing nodes are chosen from three different sites: Nancy (Griffon cluster), Lille (Chinqchint cluster) and Rennes (Paradent and Parapide clusters) connected through a WAN at 10 Gb/sec. The nodes on a cluster are interconnected by ethernet 1 and 10 Gb/sec. For performance reasons, only two processes are executed on node, each on a different processor. Due to the heterogeneity of grid resources, nodes belonging to each of the three clusters above have been used during each parallel simulation, in order to keep the homogeneity between the experiment results concerning the simulation time.

The simulation times are higher than those obtained on cluster, for the same structures, because the communication times are larger due to the informatics network latency. The speedup and efficiency values, summarized in Table 5.4, are also smaller. As for the cluster experiments, the same discontinuities can be observed on the efficiency values because of the large number of the memory accesses.

The *structure 1* is simulated on 128 processes on cluster in 0.39 hours and on grid in 0.83 hours, with a speedup value of 55 and an efficiency of 43%. However, as we increase the number of processes, the efficiency becomes smaller. For an important number of processes, the simulation time tends to be invariant. Consequently, in order to launch efficient

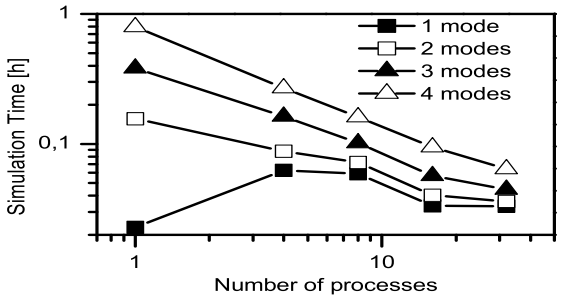


Figure 5.6: Simulation times on grid for different number of modes.

simulations we have to match the size of the structure with the number of resources used, which is much higher on a grid.

Multimodal approach on grid:

Fig. 5.6 shows the simulation times on grid versus the number of processes, when the waveguide (*structure 2* from Table 5.1) is loaded by a metallic strip. Nodes from the same clusters as for the monomode grid experiment presented above are used. Simulation times are given by considering various numbers of modes generated during the electromagnetic scattering of the incident field by the strip. In this case, different mesh steps are used according to the number of modes. So, the structure is discretized with $\Delta x = 12$ mm, $\Delta x = 7$ mm, $\Delta x = 5$ mm and $\Delta x = 4$ mm for one mode, two modes, three modes and respectively four modes. As the number of modes increases, the simulation time increases too, since the number of TLM cells in the discretized computation domain increases. There is more data to be processed and exchanged between processes. When one single mode is propagated, due to the small number of the TLM cells, the computation time is lower than the communication time. So, the simulation time increases sharply when the calculation is parallel.

Performance related to the sites position on grid:

Regarding the parallel computing developed on grid, should be noted that the performance is influenced also by the sites position where the computing nodes are chosen as the latencies between sites are heterogeneous. For example, making a TCP ping pong between two nodes, one from Nancy (Griffon cluster) and one in Lille (Chinqchint cluster), the measured latency is 0.0044 sec, while the same experiment between nodes from Nancy (Griffon cluster) and Bordeaux (Bordereaux cluster) shows a latency of 0.0093 sec. Using MPI blocking communications in a standard mode, without TCP optimization, between the two processes, messages of 3.35 Mo are sent and received, obtaining a throughput of 149 Mb/sec in the first case and 76 Mb/sec in the second case, respectively. Nodes are connected via WAN 10

Table 5.5: Compressed vs Non-Compressed message on Cluster

No. of processes	Time [h] (compress.)	Efficiency [%]	Time [h] (no compress.)	Efficiency [%]	Gain [%]
1	47.07	100	47.07	100	0
4	12.32	95.49	12.25	96.06	-0.59
8	6.44	91.31	6.43	91.51	-0.21
16	3.46	84.95	3.45	85.17	-0.25
32	2.03	72.44	2.04	71.85	0.81
64	1.34	54.61	1.43	51.15	6.33
128	1.09	33.71	1.11	33.04	2
152	1.06	29.08	1.07	28.72	1.23
174	1	26.86	1.07	25.08	6.63

Gb/sec which interconnects the sites. This experiment was launched for 1000 iterations in order to reduce the possible grid traffic interference.

Thus, in the grid simulation the sites selection where computing nodes are taken, depending on their position in the grid network, is very important. The grid performance in point-to-point communications is obtained for large messages whose transmission time is much greater than the latency. As the message size is larger, the calculation is more efficient. In Table 5.4 it can be observed that regardless of the number of processes that are used, the simulation of the first structure has greater efficiency. This is because the ratio between the communication time and the computation time is lower for the first structure. The throughput varies with the size of the message [46].

Decrease the simulation time by message compression:

Trying to decrease the simulation time, we performed the simulation of the *structure 1* from Table 5.1 on Griffon cluster by compressing messages exchanged between processes at the sender and decompressing them to the receiver. Nodes are interconnected by ethernet network, communicating by non-blocking primitives. Message compression / decompression is performed using Lempel-Ziv-Oberhumer (LZO) 2.04 [86], a GNU General Public License portable lossless data compression library written in ANSI C, using the compression algorithm LZO1x_1.

The message size is 466.28 Ko and after compression, 3.3 Ko. As in Table 5.5, the simulation time is reduced by up to 6.6%. For the first simulations, where the computing domain is larger, the simulation time is affected by a high memory access time and there is no gain obtained through the message compression. As the memory access time decreases, the reduction of the communication time by data compression at sender begins to be successful. As the efficiency is greater for the simulation with compression than the normal simulation, the gain is greater.

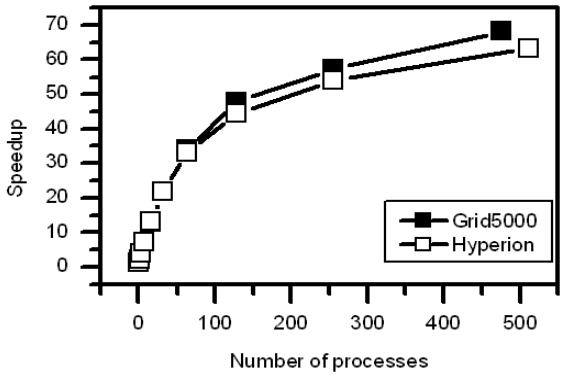


Figure 5.7: Parallel computation Speedup for a waveguide modelled with 145 million of TLM cells, obtained on grid computing and supercomputer.

Grid computing vs supercomputer experiment:

In order to evaluate the performance of our application on a large scale parallel system in terms of computing time, simulations have been performed on grid environment - Grid'5000 and on supercomputer environment - Hyperion. The TLM/Modal hybrid approach is used to execute the electromagnetic simulation of the TE_{10} mode propagating inside a matched rectangular waveguide.

Fig. 5.7 displays the speedup versus the number of processes in case of a simulation with a large waveguide structure such as *structure 1* from Table 5.1. The speedup values are obtained from simulations performed on Grid'5000 platform (Griffon, Chinqchint, Paradent, Genepi and Parapide clusters) and on Hyperion supercomputer. For performance reasons, only two processes are executed on node, each one on a different processor. temps-G5-Hyperion The architectures of the computing nodes from Grid'5000 are different from cluster to cluster. Nodes belonging to each of the first four clusters have been used during each parallel simulation, in order to keep the homogeneity between the experiment results concerning the simulation time. For the last simulation, with 476 processes, nodes from Parapide cluster were used also. The nodes from this cluster are faster than those from the other clusters, so that the simulation is not slowed.

As the number of processes is increased, the gain in speedup becomes smaller because the computation time becomes smaller than the communication time. Consequently, to perform efficient experiments on parallel architectures, the number of computational resources has to match the size of the problem. The speedup values obtained on grid are bigger than those from the supercomputer Hyperion because the grid processors from cluster Griffon are slower, which means that the computation time is higher and the speedup also, even if the nodes on Hyperion are connected by Infiniband. This can be observed easily as the computation becomes smaller.

Table 5.6: Simulation time on Cluster: measured vs predicted values.

No. of processes	Time [h] (measured)	Time [h] (predicted)	Error [%]
1	47.07	47.05	0.04
4	12.25	12.28	0.24
8	6.43	6.48	0.86
16	3.45	3.58	3.86
32	2.04	2.13	4.49
64	1.43	1.41	1.59
128	1.11	1.05	5.42
152	1.07	0.99	7.66
174	1.07	0.95	11.23

5.3 Prediction model for an application execution time

In order to reserve resources efficiently on GC platform to launch simulations using the parallel TLM/modal hyrid application, a computation time prediction model has been developed.

5.3.1 Prediction model based on least squares method

The model based on least squares method is a linear model based on the fact that the computation time increases with the number of TLM cells forming the discretized computing domain.

$$y = ax + b \tag{5.1}$$

where a, b are the time coefficients obtained by means of least squares method using a historic of experiments that have been performed on similar structures as *structure 1* from Table 5.1, varying only Z parameter.

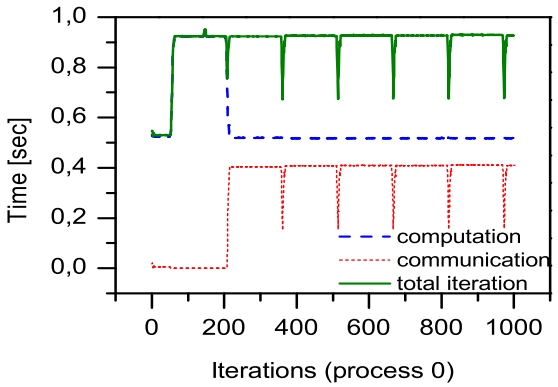
Thus, our approach is to check if the simulation time values obtained for experiments with different numbers of TLM cells, sits on the right. So, *structure 1* was simulated on computing nodes belonging to Griffon cluster, interconnected by ethernet network. The point-to-point communication between processes is controlled by non-blocking MPI primitives. For performance reasons, only two MPI processes are assigned per node, one for each CPU. The measured values for the simulation time are shown in Table 5.6.

The predicted values of simulation time using the model (5.1) are moving away from the measured values as the computation time decreases, as can be seen in Table 5.6. The errors show that the computation effort is not proportional to the number of TLM cells.

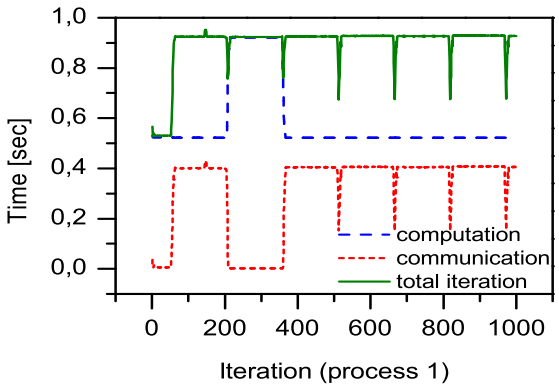
5.3.1.1 Single precision vs double precision

To better understand what happens with the simulation time obtained by the parallel TLM modal hybrid application, the computation time and communication time are displayed at each iteration in Fig. 5.8. For this, the simulation with 32 processes it is launched.

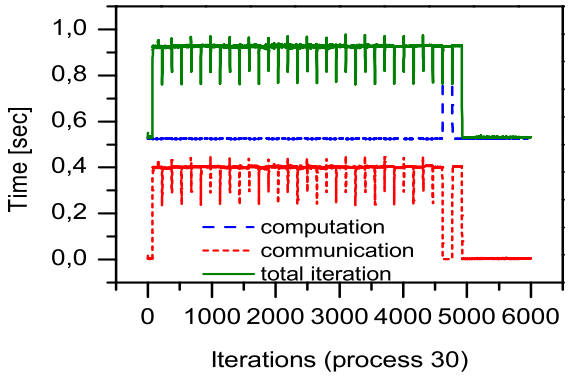
The sharply increase in the communication time for a number of iterations followed by a



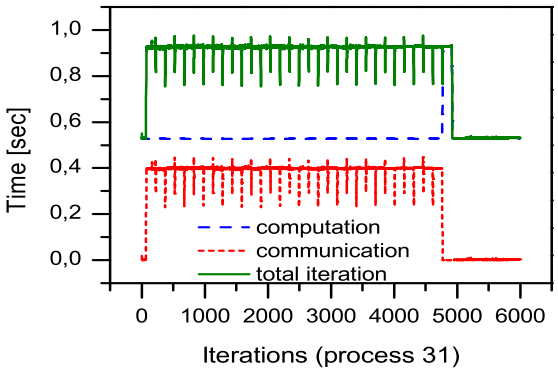
(a)



(b)



(c)



(d)

Figure 5.8: Computation time and communication time at each time step during a parallelized simulation over 32 processes.

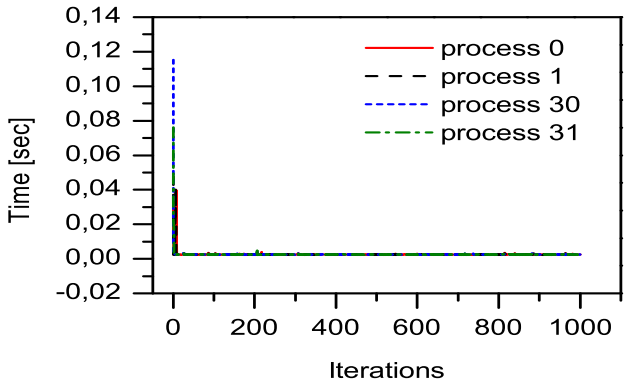


Figure 5.9: Simulation time when no computation is performed.

drop and a further increase is a behavior which is repeated according to the number of tasks used to parallelize the sequential code.

In order to explore this problem, the non-blocking MPI primitives performing the communication between processes are changed by blocking primitives, with a synchronous mode send, providing synchronous communication semantics. Eliminating the calculation from the simulation and keeping only the communication between processes, sending and receiving values of zero, the simulation time represented only by the communication is shown in Fig. 5.9 and is uniform, without deviations, which means that the phenomena displayed above, in Fig. 5.8, is due to the calculation.

Fig. 5.10 depictes the communication and computation time when the first process is excited, the TLM algorithm (Scattering and Propagation) is performed on all the processes, but the processes exchange only values of zero. Thus, at time step $t = 50$ when the first process begins to be excited, its computation time increases greatly, remaining at this level until the signal reaches the second process. Meanwhile, the other processes perform a much shorter computation but the communication time increases because all these processes wait for the first process to finish the computation, in order to communicate. After several similar tests, it is concluded that the computation time increases when the excitation becomes different from zero and at the same time, the two steps of TLM (Scatter and Connect) are executed.

In this scenario, it is considered that the computation time explodes due to the relatively large number of cache misses when the processor tries to access the data needed for processing instructions. In order to verify this scenario, Valgrind [123], an instrumentation framework for building dynamic analysis tools, is used to create the profile of the parallel TLM/modal hybrid approach concerning the cache misses occuring during the execution of the application. For the identification of possible cache misses, two different simulations have been performed using Valgrind:

valgrind −−tool cachegrind ./tlm_application

a simulation executing only the excitation of the first process, and a second simulation

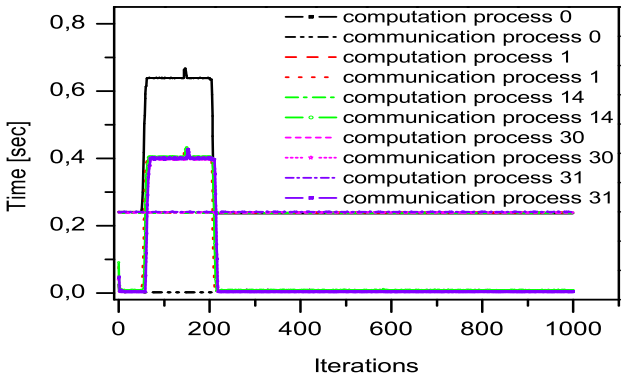


Figure 5.10: Computation time of the *process 0* increases once the excitation begins.

with the TLM computation in addition. The two profiles obtained are almost identical. The number of the cache misses occuring at reading or writing data to the memory is almost the same, for the two simulations. The sharply increase of the communication time is not explained by these profiles.

As it has seen that the computation time of the first task increases rapidly during the first iterations of the excitation, a test of the hypothesis of the data precision during the execution, is done. The signal excitation used for the structure discretized with TLM cells is Gaussian signal. Due to the small range of values, it is possible that the values exchanged between TLM cells during Scatter and Connect, are smaller than the smallest representable value with the precision of a 64-bit computer. In this situation, during the computation, the processor throws an underflow exception, defined in the IEEE Standard for Binary Floating-Point Arithmetic (IEEE 754) [54]. This floating-point exception is needed to approximate the number very close to zero with zero. While the exception is treated [48], the computation process is stopped. This operation takes some time, which adds to the computation time.

To verify this scenario, an underflow exception primitive is added to the application code, producing an interruption for each exception occuring during the execution. Making the test, the simulation is interrupted many times. So, this means that there is an underflow exception that arises during the simulation, modifing the computation time. As the application code is written in single precision, it is transformed to double precision in order to avoid this exception which does not permit the time prediction.

In Fig. 5.11, the communication and computation time obtained at each iteration, with a double precision application are displayed. The curves are uniform and predictable. The abrupt increase in the communication time for a number of iterations and followed by a fall by another increase, does not appear any more. The change of the accuracy of representation of floating-point data has solved the problem of increasing the computation time and communication time.

The values of the simulation time, on cluster Griffon, for the first structure in Table 5.1 obtained using the application with the two data format precision, are compared in Table

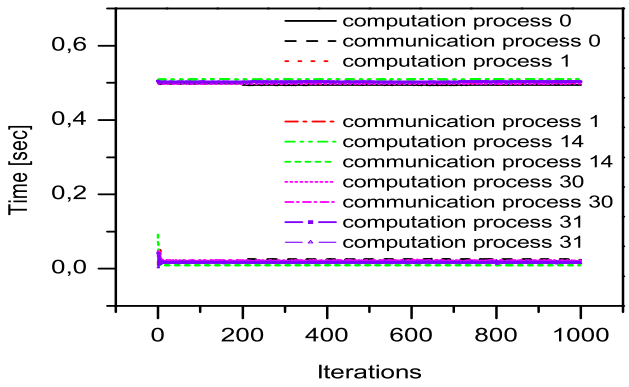


Figure 5.11: Simulation time using the application with double-precision.

Table 5.7: Simulation time on Cluster: single vs double precision

No. of processes	Time [h] (single precision)	Time [h] (double precision)	Gain [%]
1	47.07	46.13	2
4	12.25	13.8	12.7
8	6.43	7.08	10.11
16	3.45	3.52	2.05
32	2.04	1.44	29.64
64	1.43	0.69	51.66
128	1.11	0.37	66.62
174	1.07	0.30	71.29

5.7. The maximum gain in time when using double precision is about 71%. In Fig. 5.12, the speedup curves, single vs double precision, are depicted. Eliminating the time due to solving exceptions, the speedup is much higher.

From now on, only the application in double precision is used. Now, trying to predict only the computation time using the prediction model given in (5.1), when the first structure from Table 5.1 is simulated on cluster Griffon, for different number of processes, important errors are still found up to 43%. The errors increase as the number of processes increases, ie the computation domain is smaller. So, the variation of the computation time with the number of the TLM cells is not linear. The model (5.1) cannot be used to predict the computation time for discretized domains with number of TLM cells ranging on large intervals.

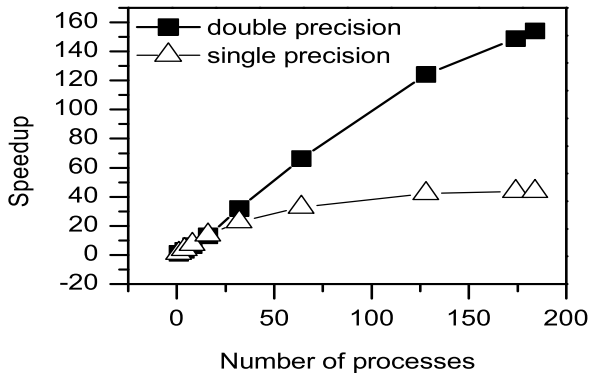


Figure 5.12: Parallel computation Speedup: single vs double prediction.

5.3.2 Application profile based prediction model

In order to estimate the simulation time on GC, a computation time prediction model based on a hybrid approach has been developed. This prediction model combines both prediction techniques: historic and profile-based prediction. The simulations forming the history of experiments for the model were done on cluster *Griffon*, from Nancy site of Grid'5000 platform. For the simulations that are included in the historic database, only the computation time has been considered.

5.3.2.1 "Four coefficients" model

To estimate the computation time T_{cal} , we define a prediction model according to the algorithm of the application:

$$T_{cal} = c_1 + XYc_2 + tXYc_3 + tXYZc_4, \tag{5.2}$$

where *X,Y,Z* represent the number of TLM cells on the three spatial directions, t is the number of time steps and c_i, i=1..4, are the time coefficients corresponding to different blocks in the code of the hybrid approach implementation. Thus, according to Fig. 5.2, c_2 corresponds to the block *Space Discretization*, c_3 models the block *Boundary Conditions* and *Output data* and c_4 represents the two blocks of the TLM's core, *Scattering* and *Propagation*. The first coefficient, c_1, corresponds to the initial time of the algorithm, which does not depend on the number of TLM cells nor the number of time steps. The coefficients are determined using a linear programming formulation based on a history of M experiments:

$$\begin{aligned} t_1 &= c_1 + X_1Y_1c_2 + t_1X_1Y_1c_3 + t_1X_1Y_1Z_1c_4 \\ t_2 &= c_1 + X_2Y_2c_2 + t_2X_2Y_2c_3 + t_2X_2Y_2Z_2c_4 \\ &\vdots \\ t_M &= c_1 + X_MY_Mc_2 + t_MX_MY_Mc_3 + t_MX_MY_MZ_Mc_4, \end{aligned} \tag{5.3}$$

where M is greater than or equal to the number of the coefficiets, t_i, X_i, Y_i, Z_i with i=1..M, are given and the coefficients are unknowns; t_i represents the computation time of the experiment i. In this linear formulation, constraints are imposed in order to have coefficients grater than zero.

In Table 5.8 there are presented the predicted values, using the model 5.2, for the computation time on grid of the structures 1 and 2 outlined in Table 5.1. The computing nodes are located on clusters Griffon (Nancy), Chinqchint (Lille), Parapide and Paradent (Rennes). The computation time is imposed by the slowest computing nodes which are those from Griffon. The prediction model is also based on experiments executed on Griffon cluster.

The prediction errors are still important for the simulation with only one process and also when a large number of processes is used.

Table 5.8: Computation time on grid: measured vs predicted values

	No. of processes								
	1	8	16	32	64	128	256	332	422
Structure 1									
Measured [h]	46.13	5.82	2.89	1.19	0.58	0.3	0.18	0.13	0.11
Predicted [h]	32.1	4.06	2.05	1.05	0.55	0.3	0.17	0.14	0.12
Error [%]	30.4	30.3	28.9	11.7	5.16	0.17	3.2	11.9	10.4
Structure 2									
Measured [h]	11.53	1.41	0.58	0.27	0.14	0.07	0.04	-	-
Predicted [h]	7.97	1.01	0.5	0.26	0.14	0.07	0.04	-	-
Error[%]	30.9	28.4	13.5	2.27	0.25	6.2	16.18	-	-

Table 5.9: Computation time on grid: measured vs predicted values.

	No. of processes								
	1	8	16	32	64	128	256	332	422
Structure 1									
Measured [h]	46.13	5.82	2.89	1.19	0.58	0.3	0.18	0.13	0.11
Predicted [h]	45.5	5.71	2.87	1.45	0.74	0.38	0.21	0.16	0.13
Error [%]	1.41	1.83	0.59	22	28	29	16.4	29.9	22.5
Structure 2									
Measured [h]	11.53	1.41	0.58	0.27	0.14	0.07	0.04	-	-
Predicted [h]	11.3	1.44	0.73	0.38	0.2	0.12	0.07	-	-
Error[%]	1.84	2.32	27.1	42.5	48.8	64.6	90	-	-

5.3.2.2 "Seven coefficients" model

In order to improve the estimation of the computation time T_{cal}, we define a prediction model according to the algorithm of the application, deepening the application representation in the model by increasing the number of coefficients:

$$T_{cal} = c_1 + XYc_2 + tc_3 + tXYc_4 + tXZc_5 + tYZc_6 + tXYZc_7, \tag{5.4}$$

where X, Y, Z represent the number of TLM cells on the three carthesian directions, t is the number of time steps and c_i, $i=1..7$ are the time coefficients corresponding to different blocks in the code of the application. Thus, according to Fig. 5.2, c_2 corresponds to the block *Space Discretization*, c_3, c_4, c_5, c_6 models the blocks *Boundary Conditions* and *Output data* and c_7 represents the two blocks of the TLM's core, *Scattering* and *Propagation*. The coefficients are determined using a linear programming formulation similar to (5.3), based on a history of experiments.

In Table 5.9 there are presented the predicted values, using the model (5.4), for the computation time on grid of the structures 1 and 2 outlined in Table 5.1. The computing nodes are located on clusters: Griffon (Nancy), Chinqchint (Lille), Parapide and Paradent (Rennes). The computation time is imposed by the slowest computing nodes which are those from Griffon. The prediction model is also based on experiments executed on Griffon cluster.

The prediction errors are still important for the simulation using a large number of processes. In fact, the errors increase as the computation domain becomes smaller. This is

Table 5.10: Cache misses evolution when increasing the number of TLM cells.

X	Z	No. of TLM cells (*10^5)	Time [h] (measured)	No. of cache misses (*10^9)	Memory 2*X*Z [Mo]
10	20	0.042	3.84E-4	0.23	0.03
20	40	0.168	0.0013	0.84	0.14
40	80	0.672	0.0095	3.21	0.58
75	130	2.04	0.0488	9.56	1.78
80	160	2.68	0.0649	12.52	2.34
120	240	6.04	0.15	28.21	5.27
160	320	10.75	0.33	64.96	9.37

even more evident in the case of the second structure, which is smaller than the first one. This behavior of the model is due to the experiments used in determining the coefficients c_i, which are represented by simulations whose time is superior to two hours, with a greater computation domain. It can be observed that as the simulated structure differs increasingly according to those used in building the prediction model, the estimated computation time has a larger error. The set of the input parameters X, Y, Z plays an important role in the prediction of the computation time.

The "Seven coefficients" and "Four coefficients" models have a good accuracy when the simulation to be predicted is nearly to the simulations underlying the coefficients determination.

5.3.2.3 "Cache-misses" model

For a more rigorous assessment of the application algorithm and to understand the influence of parameters X, Y, Z on the computation time, we performed several test simulations, varying only the parameters X and Z and maintaining Y and t constant ($Y = 21, t = 10000$). The test simulations are done using computing nodes from Griffon cluster. To highlight the impact of the input parameters on the computation time, the simulations are performed on a single process, without parallelization and communication time. Using Valgrind tool profiler, the number of cache misses for writing and reading data from RAM memory for each simulation, has been counted in Table 5.10.

It can be observed that as the number of TLM cells increase by varying X and Z, the number of cache misses increases too. But, the variation is not linear. In Fig. 5.13, the total number of cache misses vs the number of TLM cells are presented. Also, a detail of the number of cache misses occuring inside the *Scattering* and *Propagation* blocks during the simulation is shown. It is evident that the total number of cache misses that occurs during the simulation is given especially by the cache misses that occur when executing the TLM's core. However, the variation of the cache misses given by the *Scattering* operation with the number of TLM cells is linear.

The non-linearity of the cache misses variation is produced during the *Propagation* step. The cache misses appearing during the simulation influence directly the computation time, and their sharp increase cannot be predicted. This non-linearity in the variation of the

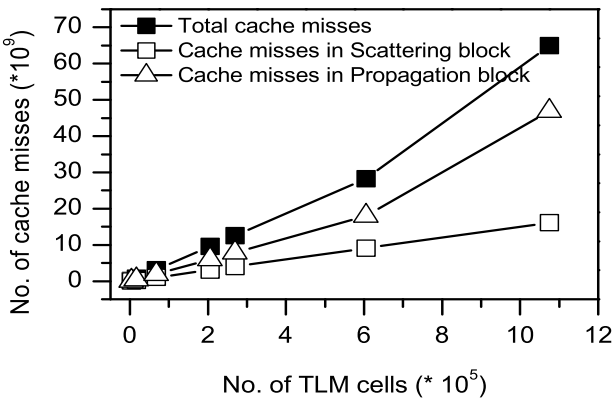


Figure 5.13: Cache misses evolution when increasing the number of TLM cells.

number of cache misses with the number of TLM cells explains the non-linearity of the variation of the computation time with the number of TLM cells, by the time required to read or write data in RAM memory during the simulation, that adds to the computation time. Therefore, the prediction models outlined above have high accuracy only for simulations similar to those that were used to calculate c_i coefficients, having the same number of cache misses that occur while simulating.

During the simulation, the processor passes more than 90% of the computation time on the TLM's core. In *Scattering* block, the scattering phenomena that occured at each TLM cell, at each time step, is modelled. After the scattering process, in *Propagation* block the connection between the neighboring cells is realised. Thus, the incident signal on each cell at next time step is computed.

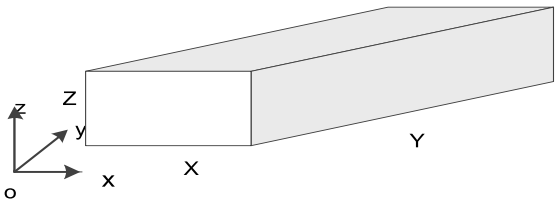


Figure 5.14: 3D view of the structure (X, Y, Z - TLM cell number on the three cartesian directions).

During the connection step, a lot of data is computed by the processor. In the case of memory hierarchical architectures, the data required for the TLM cells connection has to be charged in cache memory. As the discretized structures are larger, the RAM access time increases.

So, the speed of the sequential execution will be slower than on a parallel computer with similar processors and memory architectures. Let's take a look on *Propagation* block implementation, outlined in Listing 5.1. All the TLM cells of the discretized structure are placed in a vector, which is processed with three loops corresponding to the number of TLM cells along the three axes - see Fig. 5.14. The connection is realized for each cell. In order to

connect the current cell to his neighbors, jumpers inside the vector are done - see Fig. 5.15. Let's consider the discrete coordinate of the current cell as *ind*. The jumpers required to make the connections of the indexed node *ind* with his neighbors are: *ind_ip*, *ind_jp*, *ind_kp*.

In order to not charge twice the same data in the cache memory, the cell referenced with the index *ind_kp* has to remain in the memory until it becomes the current cell. So, during the connection step the processor needs to access a total volume of data of: *2*X*Y*12*8* bytes, where: *2* is the number of *x*-*y* plans of TLM cells required (the plan of the current cell and the upper neighboring plan); X, Y are the number of TLM cells on the x and y axes; 12 represents the values caracterizing the field polarizations on each cell; 8 is the number of bytes required to stock each value in memory.

Listing 5.1: Propagation algorithm

```
connect()
{...
 nxy = X*Y;
 for (int k = 0; k < Z; k++) {
   for (int j = 0; j < Y; j++) {
     for (int i = 0; i < X; i++) {
        ind = i + j*X + k*X*Y;
        if(i < X-1) {
          ind_ip = ind + 1;
          //...connection on x axis
        }
        if(j < Y-1) {
          ind_jp = ind + X;
          //...connection on y axis
        }
        if(k < Z-1) {
          ind_kp = ind + nxy;
          //...connection on z axis
        }
     }
   }
 }
```

As the discretized structure is larger, the amount of data to process becomes important and the memory used by the application increases too. In this scenario, the number of RAM access times (cache misses) during the simulation increases in order to give to the processor the required data. The computing time increases too. If the sequential time of the problem

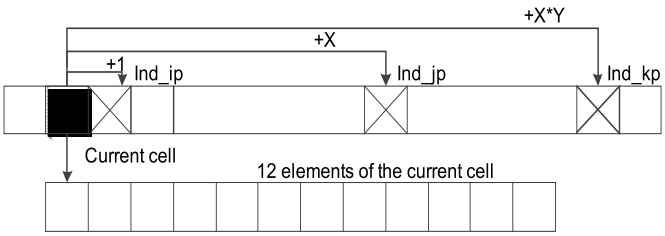


Figure 5.15: Memory structure representation.

is affected by the cache misses phenomena, the performance metrics of the simulations will have discontinuities. The speedup could be superlinear and the parallel overhead negative.

In order to estimate the computation time, T_{cal}, of a simulation, we define a prediction model according to the algorithm of the application:

$$T_{cal} = c_1 + XYZtc_2, \tag{5.5}$$

where X, Y, Z represent the number of TLM cells on the three spatial directions, t is the number of time steps and c_i, i=1,2, are the time coefficients. c_2 corresponds to *Scattering* and *Propagation* blocks in the code of the application. The coefficients are determined by a linear programming formulation similar as in (5.3), of a system of historic experiments. Only two coefficients are considered because the TLM's core computing block represents more than 90% of the computation time; thus, the ill-conditioned matrices computation when dealing with more coefficients is avoided too.

To have a good agreement between the measured and the predicted values for the computation time, the problem of the cache misses presented above has to be considered in the profile-based prediction of the TLM/modal hybrid application. For this, two predictive models for computation time have been designed.

First model for the estimation of the computation time is designed for the structures whose space required to store *2*X*Y*12*8* bytes does not exceed the processor cache size. In this case the number of the cache misses has not an important variation. The simulations forming the database of this model meet the same condition. The second model estimates the computation time for the structures whose space required to store *2*X*Y*12*8* bytes exceeds the processor cache size. This model takes into account the cache misses phenomena during the simulation of large structures. The simulations forming the database of this model show a significant number of cache misses during execution.

The nodes that have conducted these simulations are equipped with Intel Xeon 5420 Processor. Up to 6 MB of L2 Cache can be allocated to one core. As the simulations were performed sequentially, the space required to store *2*X*Y*12*8* bytes is limited to 6MB. According to the Table 5.10 and Fig. 5.13, the point where the variation - number of cache

misses vs number of TLM cells - changes, corresponds also to this limit of 6MB imposed by the architecture of the computing node used. So, using the two models, the prediction is splitted in two parts, each one having a linear variation, number of cache misses vs number of TLM cells.

For parallel computations, the communication time between processes has to be determined too. The communication time for one single sent message at a time step, T_{com}, is defined by:

$$T_{com} = lat + msg/deb \tag{5.6}$$

where *lat* is the network latency, *msg* represents the size of the sent message between two processes and *deb* is the network throughput. The size of the sent message depends on the number of TLM cells that are on the transversal surface.

The total simulation time of a parallel application with n processes, is given by:

$$T_n = c_1 + XY(Z/n)tc_2 + 4tT_{com} \tag{5.7}$$

where the coefficient *4* represents the worst case, when a task sends and receives from the two neighboring processes.

The required resources to compute a problem with a given efficiency:

Considering a given structure (X, Y, Z, t), we can determine also the maximum number of processes n required for computing the structure with the efficiency of at least e:

$$n \leq \frac{c_1^A + XYZt(c_2^A - ec_2^B)}{e(c_1^B + 4tT_{com})}, \tag{5.8}$$

where the coefficients c_i^A, *i=1,2*, are the time coefficients from (5.5), corresponding to a structure with a memory space required of *2*X*Y*12*8* bytes; the coefficients c_i^B, *i=1,2*, are the time coefficients from (5.5), corresponding to a structure with a memory space required of *2*X*Y/n*12*8* bytes. Comparing these memory required spaces with the limit of the L2 cache memory of 6MB, one of the two models for the prediction of the computation time presented above can be chosen.

Prediction results:

Structures 1 and *2* from Table 5.1 have been computed on grid, using different number of processes. Fig. 5.16 displays the computation times versus the number of processes (the communication time has not been considered). The predicted values are in excellent agreement with the measured values. The two prediction models presented above, of the form

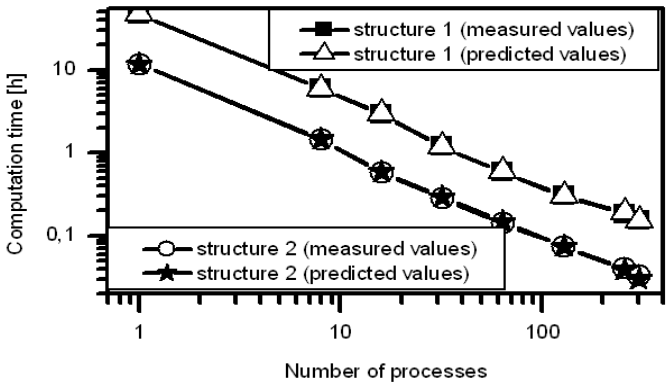


Figure 5.16: Computation time on grid for different TLM cell numbers: measured values versus predicted values.

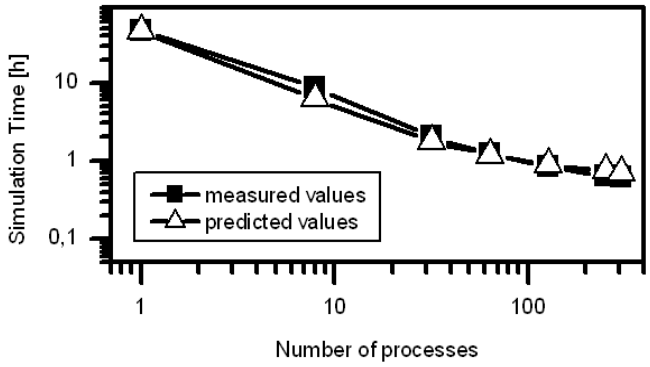


Figure 5.17: Total simulation time on grid: measured values versus predicted values.

given in (5.5) are used to determine the estimated values for the computation times. The average error for the first structure is 1.51% while the maximum error is 6.09%. For the second structure, the average error is 2.18% and the maximum error is 6.14%.

Fig. 5.17 shows the simulation times on grid (Griffon, Chinqchint, Paradent and Parapide clusters) versus the number of processes, for *structure 1*. The predicted values are presented too. The communication time between the processes, given by (5.6) has been considered. The predicted values are in excellent agreement with the measured values. The average error is 10.6%.

In Fig. 5.18 it has been estimated the number of processes that are necessary for the grid calculation of a problem with the efficiency of 43%, according to the prediction model given in (5.8). The discontinuities that appear in the surface are caused by the different calculation of the number of processes using the two predictive models of computation time presented above, depending on the structure size, i.e. the memory space required for storing *2*X*Y*12*8* bytes. For *structure 1*, 107 processes are estimated, with an error of 16%, compared to the measured value, 128.

In order to prove the real benefits of the grid environment, the structure 6 from Table 5.1

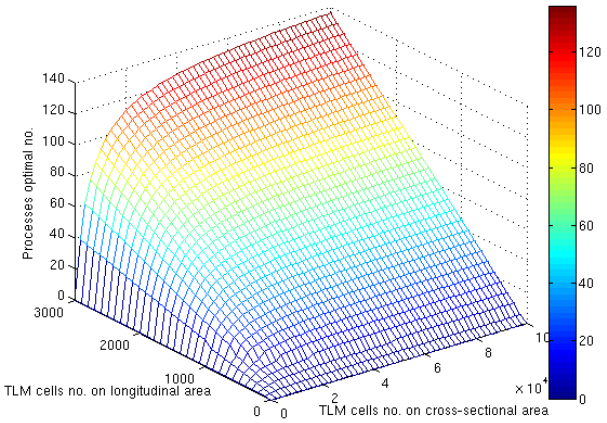


Figure 5.18: Processes optimal number.

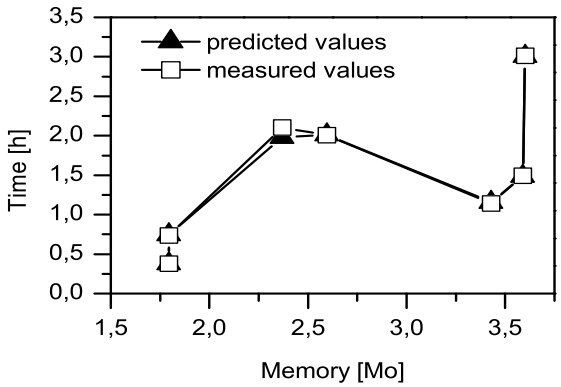


Figure 5.19: Computation time prediction for various simulations when TLM cells model dielectric media - the memory space required for storing *2*X*Y*18*8* bytes is smaller than the processor cache memory size.

is simulated. A supersized rectangular matched waveguide, is discretized by 1.4 billion TLM cells, that would require 135 GB of RAM memory and more than 130 days of simulation time on a single process. In terms of memory, a shared memory systems should be used to deal with the problem, but the time needed to reserve the computer is enormous. On grid, we have computed this structure in 14 hours using 318 processes. The memory ressources required on each process is about 0.5 GB.

A prediction model similar to that shown in (5.5), considering the cache misses problem, is designed to estimate the computation time when simulating the electromagnetic field propagation inside dielectric media, modeled by means of TLM SCN cells with stubs, with 18 values of polarization. In Fig. 5.19, the measured and predicted values for the computation time corresponding to various simulations performed on cluster Griffon are displayed. The input parameters X, Y, Z, t are varied so that the product *2*X*Y*18*8* does not exceed

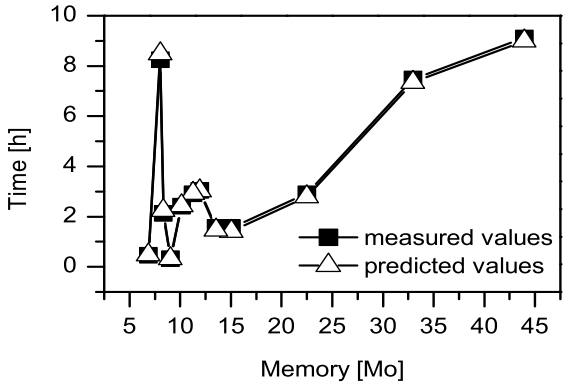


Figure 5.20: Computation time prediction for various simulations when TLM cells model dielectric media - the memory space required for storing *2*X*Y*18*8* bytes is larger than the processor cache memory size.

the processor cache memory size of 6Mo. The maximal error is about 5.7%.

In Fig. 5.20, the measured and predicted values for the computation time corresponding to various simulations performed on cluster Griffon are displayed. The input parameters X, Y, Z, t are varied so that the product *2*X*Y*18*8* exceeds the processor cache memory size of 6Mo. The maximal error is about 5.29%.

5.4 Conclusions

This chapter presents an original approach which combines hybrid CEM techniques with large scale parallel systems such as GC and supercomputer in order to speed up the modeling of large electromagnetic problems. The study highlights the role of parallelization scheme based on *message passing* paradigm, grid versus supercomputer, with respect to the size of the problem and its repartition. The computation performance is analyzed when the communication between MPI processes is managed by blocking and non-blocking primitives. It is concluded that given the nature of the TLM application, the simulation times are almost equal, for the two types of communications.

It is also outlined how drastically decreases the computation performance when more MPI processes run simultaneously on the same processor, phenomena that appears on the shared memory multiprocessors due to the shared memory bus saturation. To launch efficient simulations, the size of the structure has to be matched to the number of computing resources, according to their architecture (memory latency, bus memory bandwidth).

The application performance on GC is influenced also by the sites position where the computing nodes are placed due to heterogeneous latencies between sites. Also, the computing nodes belonging to different clusters are heterogeneous, so their choice must be made so that the computation performance not be affected. The grid simulations performed using

the multimodal approach show that as the number of modes increases, the simulation time increases too. The communication time has to be smaller than the computation time, but also the message has to be as larger as the transmission time be much greater than the network latency, especially on the grid where the communication costs are larger then those on cluster.

To relax the simulation time constraints, an algorithm for compressing/decompressing messages exchanged by the MPI processes is implemented, in order to reduce the communication time. The gain obtained in the simulation time is up to 6.6%. Several simulations are done on Hyperion supercomputer. The speedup values are similar to those obtained on grid, due to the fact that the grid processors are slower and thus, the computation time is higher.

Four prediction models for the grid computation time of the TLM/modal hybrid application are developped, in order to reserve efficiently the computing resources. The models accuracy is improved step by step, by considering the application specificity and the computing architecture features. Through a detailed analysis of the application algorithm, floating-point exceptions are identified during the TLM simulation increasing significantly the computation time, and also making extremely weak the accuracy of the time prediction. Eliminating these exceptions by passing the application in a double precision representation of data, a gain of 71% for the simulation time is obtained.

Taking in consideration also the cache misses phenomena occuring during the simulation, with respect to the TLM Propagation algorithm and to the hierarchical memory architecture lying on computing nodes, a computation time prediction model outlined in - (5.5), let us estimate the ressource reservation time to simulate a given structure on grid, with an error under 10%. Also, characterizing the performance of the simulations on GC, rules for the estimation of the required resources have been given in (5.8).

In order to prove the real benefits of the grid environment, a supersized rectangular matched waveguide, discretized by 1.4 billion TLM cells, requiring more than 130 days of simulation time by a traditional computation which makes a reservation very difficult, it is simulated in 14 hours using 318 processes in parallel.

Chapter 6

Supersized and Complex Structures

In this chapter are presented the modeling results of a complex structure using the hybrid numerical tool, based on TLM and modal approach, presented in Chapter 4 and the parallel computing techniques presented in Chapter 5. Then, the same approach is applied for the treatment of multiscale problems - simulation of the electromagnetic field propagation within a complex and oversized structure as an aircraft cabin. The computing resource usage issues in this case are exposed. The GC and a supercomputer are two computation ways proposed to solve the problem. The prediction model is used to estimate the simulation time and the optimal required resources for structure calculation.

6.1 State of the art

Nowadays aircraft manufacturers are interested to offer more services to their passengers. The development of wireless networks has stimulated the research toward on board wireless services. On board Wireless Local Area Networks (WLAN) permit to passengers to use their new portable devices over the internet. The replacement of the cables found inside the aircraft with a wireless network means a gain of space, loss of weight and simplifies the manufacturing and maintenance steps. The Global System for Mobile Communications (GSM) networks on airplane cabins represent another challenge. The EMI phenomena made by the different PEDs to the communication and navigation systems has to be considered. However, the phones and the wireless internet connection are still not available today inside the cabin. Their interaction with the aircraft is still subject for research studies.

Electromagnetic simulation of the propagation channel, which is less expensive than the experimental approach, provides the derivation of the field propagation inside the cabin and permits to adequately design the wireless services. The cabin of an aircraft represents a complex media with seats, luggage, passengers etc. In order to realize a full-wave simulation of the propagation channel at wireless standard frequencies inside the aircraft cabin, the mesh step has to be to small in comparison to the problem dimensions. So, the traditional

computing resources cannot provide support for such a challenge.

In [56], authors use an efficient FDTD based commercial software to analyze the electric field inside an aircraft (Airbus A319) radiated by a dipole of a GSM900 cellular network placed inside the passengers cabin. For accuracy in predicting results, CPU time and memory requirements can hardly be met for a full-wave simulation. Using an electromagnetic propagation prediction tool, in [135]-[136] it was proven that the internal components of an aircraft play a vital role in propagation and should be considered in model for wireless network performance. This numerical tool is based on ray-tracing as a method of calculation, which offers a quick but limited calculation of the propagation. A numerical approach, based on an asymptotic code and a PoWer Balance (PWB) method, for the assessment of the high frequency coupling in a complex oversized structure is described in [65].

6.2 Modeling Complex Structures

Using the numerical hybrid tool described in Chapter 4 and the parallel computing resources reported in Chapter 5, the modeling of large complex structures is the main objectif of the present chapter. First, it is presented a complex structure like an airplane cabin, i.e 1:300 scale with respect to the regular one, having a rectangular cross section, in order to validate the parallel TLM/modal hybrid approach. An overview of the structure is shown in Fig. 6.1 and Fig. 6.2.

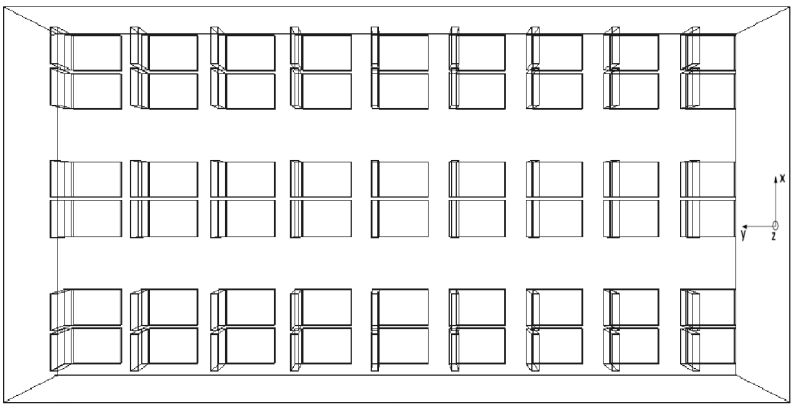


Figure 6.1: Perspective view over a small airplane model with seats.

As shown in Fig. 6.3, the structure with perfect electric walls is excited by a rectangular aperture with TE_{10} mode distribution, placed on the transversal wall opposite to the short circuited termination. The simulation is performed for a mesh step of 1 mm and 2000 iterations.

The seats with the dimensions outlined in Fig. 6.4 are modeled with the material $\epsilon_r = 4.3$

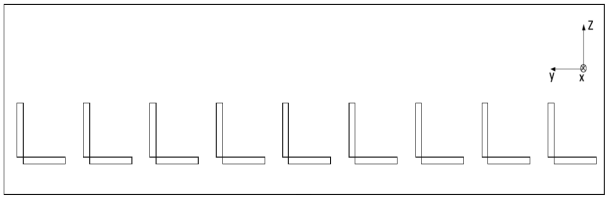


Figure 6.2: Longitudinal section of the small airplane model with seats.

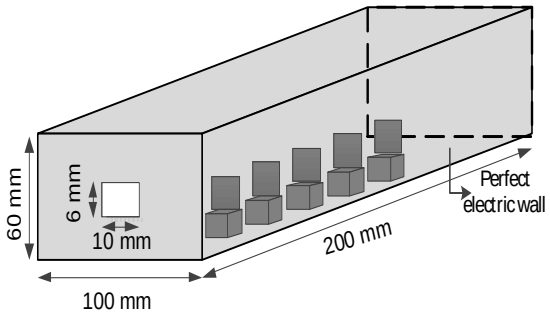


Figure 6.3: Schematic view of the whole structure.

and $\mu_r = 1$ by means of stubs added to the SCN node structure as in (4.10) and (4.11).

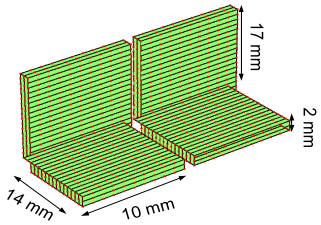
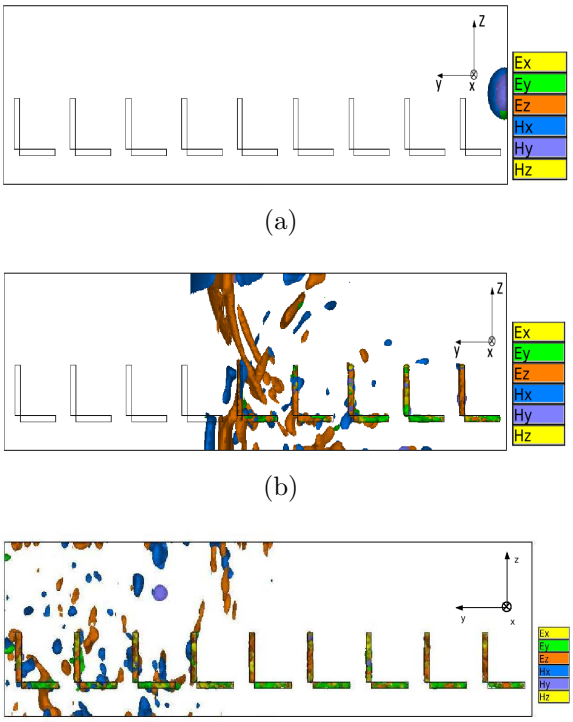


Figure 6.4: TLM seats model with $\epsilon_r = 4.3$ for a small airplane model.

In Fig. 6.5 is presented the electromagnetic field propagation by the six components representing TE and TM modes excited by the aperture, inside the structure at different time steps. An overview of the structure at the moment $t = 700$ is given in Fig. 6.6. After exciting the structure, it can be observed that the electric field components are concentrated inside the material of the seats as a standing wave. The magnetic field continues to come in and out of chairs for each iteration. Keeping the same structure, electrical losses for the material of seats are modeled with a loss tangent $\tan\delta$ of 0.01.



(a)

(b)

(c)

Figure 6.5: Field propagation inside the small airplane structure at (a) excitation time, $t = 103$, and at time steps (b) $t = 350$ and (c) $t = 700$.

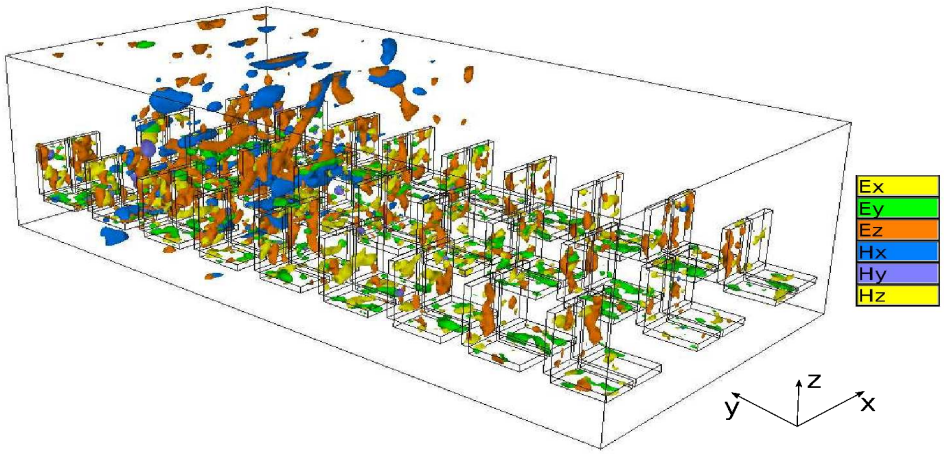


Figure 6.6: Sight over the field propagation inside the small airplane structure at time step $t = 700$.

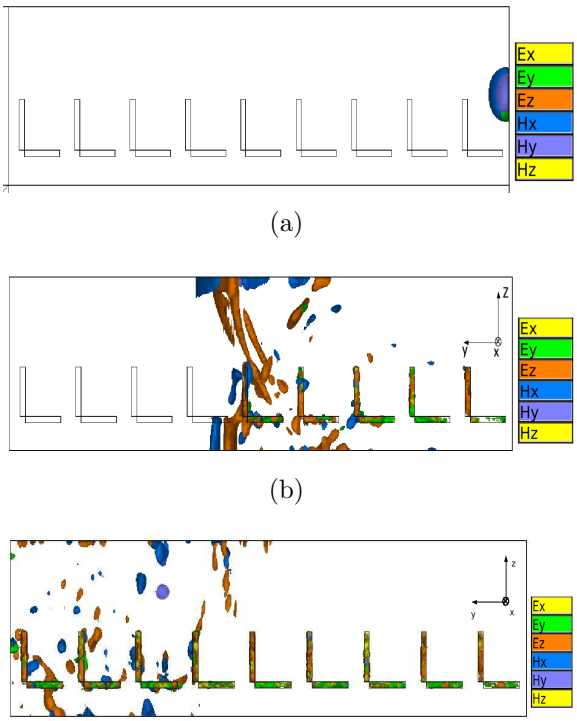


Figure 6.7: Field propagation inside the small airplane structure modelled with $\tan\delta = 0.01$ of the dielectric, at (a) excitation time, $t = 103$, and at time steps (b) $t = 350$ and (c) $t = 700$.

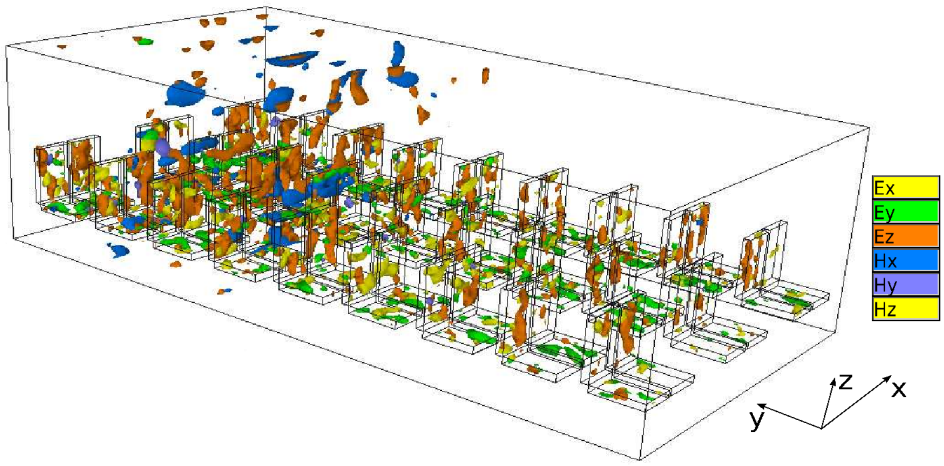


Figure 6.8: Sight over the field propagation inside the structure modelled with $\tan\delta = 0.01$ of the dielectric, at time step $t = 700$.

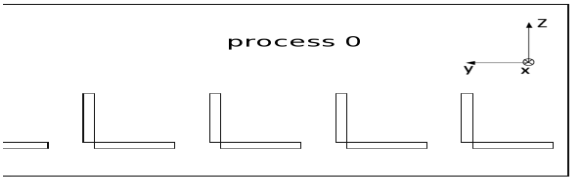


Figure 6.9: A subdomain of the parallelized small airplane structure corresponding to the process 0.

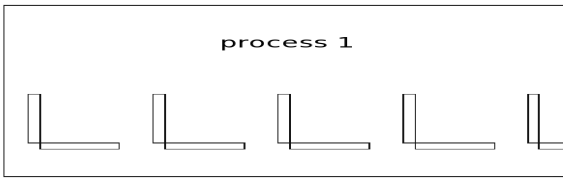


Figure 6.10: A subdomain of the parallelized small airplane structure corresponding to the process 1.

The electrical energy loss can be observed in Fig. 6.7 compared with the previous results from Fig. 6.5. All the six components of the field are displayed at the same level of amplitude. An overview of the structure at the moment $t = 700$ is given in Fig. 6.8.

Parallelization of the TLM algorithm:

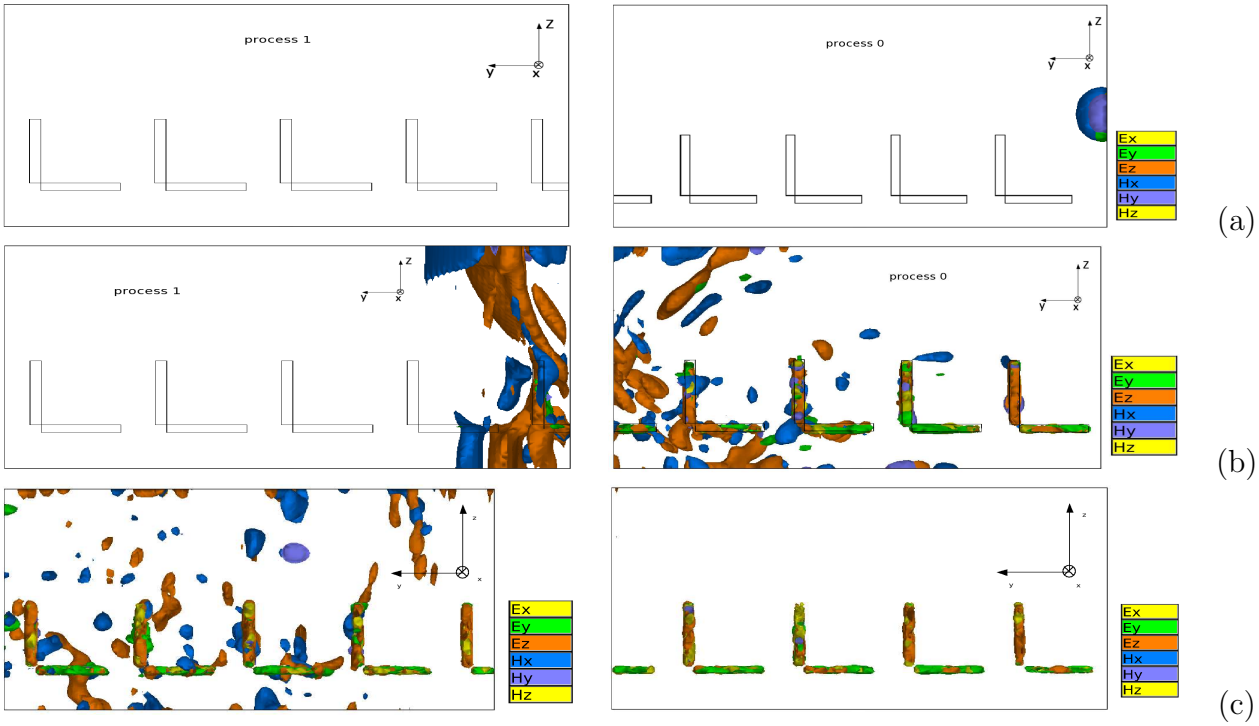
The next step is the parallelization of the structure. For this, the lossless cabine model is chosen. The structure is divided into two subdomains, each of which is assigned to an MPI process, as in Fig. 6.9 and 6.10. Thus, each process separately calculates a portion of the entire discretized computation domain defined by the cabine model structure, communicating with other process at each iteration intermediate data in order to obtain a simulation of the whole structure.

The propagation of the electromagnetic field inside the parallel structure is summarized in Table 6.1. The evolution of the field inside the two subdomains of the parallel structure, assigned to Process 0 and Process 1, is presented at three different time steps: the moment when the structure is excitated, corresponding only to the first process (Process 0) and two other moments at $t = 350$ and $t = 700$.

The two processes communicate between them by an exchange of messages, based on MPI, at each time step of the simulation. The message contains the two polarization values of the ports placed at the interface between the neighbour processes, corresponding to each TLM-SCN cell.

Table 6.1: Process 0 and Process 1 at different time steps during the simulation.

(a) $t = 103$; (b) $t = 350$; (c) $t = 700$



Even if the two processes start the computation at the same time, however, they begin to exchange the real data representing the electromagnetic field only when it has already completed half of the structure, ie, the subdomain corresponding to the first process (Process 0). So, the computation effort for each process is halved, while the simulation results remain unchanged. The parallel computation and the communication between the processes take place in real-time during the simulation.

6.3 Modeling Supersized and Complex Structures

The same parallel hybrid approach is used to model the effect of the objects found inside the airplane cabin on the electromagnetic field. The model is created for Airbus 350-1000 [3]. The dimensions of the cabin passengers are: 5.61 m width, 2.5 m height and 61.44 m length. The model is accomplished for studying the electromagnetic field behavior at 3.7 GHz, the frequency of the IEEE standard for WLAN, 802.11y-2008 [1].

The model has a mesh step of 8 mm upon (4.1), and 30869 time steps in order to allow the wave propagation in a round trip. The entire structure is discretized upon 1.6 billion of TLM SCN cells. The model includes 360 seats with the dimensions: 56 cm width, 72 cm backrest height, 64 cm length and 7.2 cm thickness as it can be seen in Fig. 6.11. The seats are placed six abreast. It has been assumed that the cabin fuselage is a Perfect Electrical

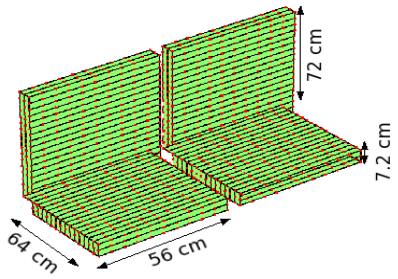


Figure 6.11: TLM seats model with $\epsilon_r = 4.3$ for the airplane cabin model.

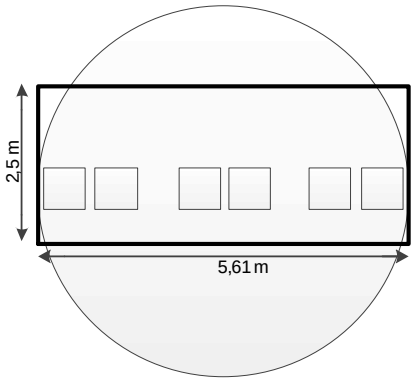


Figure 6.12: Schematic cross-sectional view of the cabin model.

Conductor (PEC), having a rectangular cross section as in Fig. 6.12.

The seat material is the polyamide with the relative electrical permittivity of 4.3. The antenna used is a rectangular aperture with TE_{10} mode distribution, placed on the transversal wall opposite to the short-circuited load, as in Fig. 6.13. The dimensions of the aperture are: 56 mm width and 25 mm height. The computation of the structure described above is not possible with the traditional computers because of the limitation of the memory resources. The model at 3.7 GHz needs about 226 Go of memory resources, according to a trivial analysis of the TLM model implementation based on a double precision data representation:

$$\frac{5610}{8} \times \frac{2500}{8} \times \frac{61440}{8} \times 18 \times 8(bytes)$$

where the first three terms represent the number of TLM cells on the three spacial directions with respect to the mesh step, *18* is the number of polarization values for each TLM SCN cell of the discretized computation domain and *8* is the number of bytes required to represent a value.

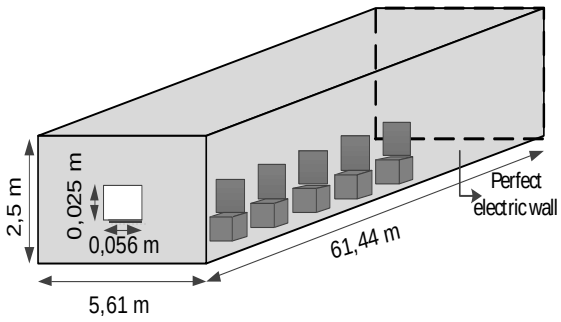


Figure 6.13: Schematic view of the whole structure.

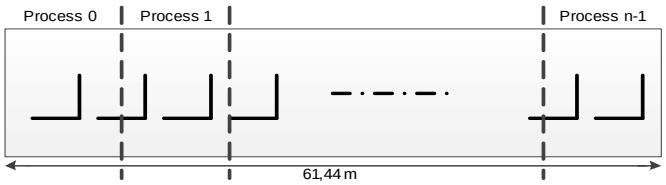
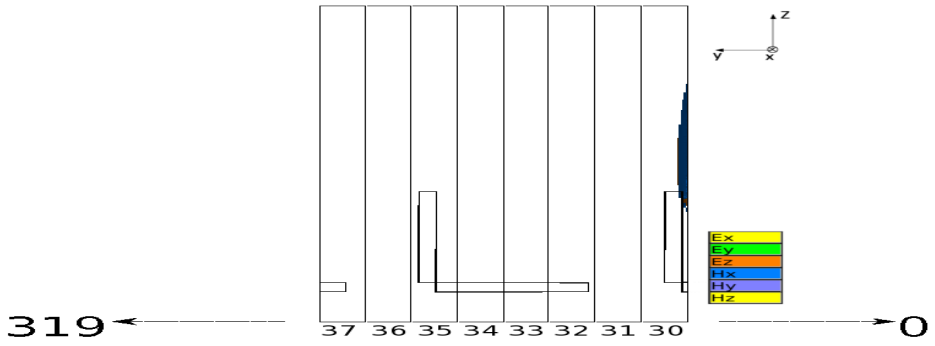


Figure 6.14: Parallel implementation of the aircraft cabin model with seats (longitudinal view).

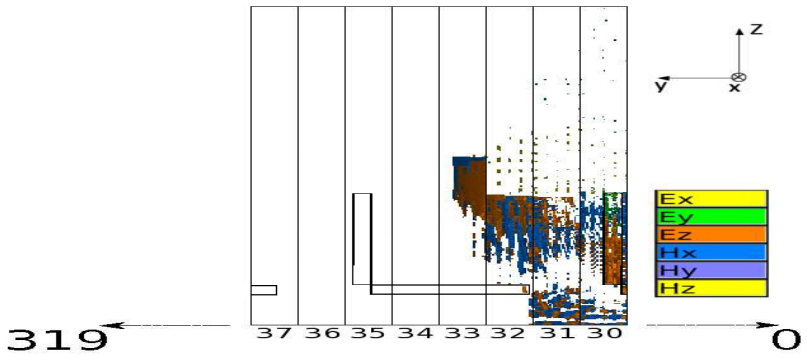
The high computation time is another constraint in terms of continuous access to the computing resources. So, in order to compute the electromagnetic fields in the overall aircraft cabin it is necessary to divide the structure in several subdomains to be calculated at the same time on different computing machines.

A schematic view of the parallel hybrid approach implementation can be seen in Fig. 6.14. The excitation is performed only by the first task, while the short-circuited termination is computed by the last process. At each time step, the TLM algorithm is applied to the discretized cells. Before moving to the next time step, each process communicates to his neighboring processes the intermediate simulation data. TLM cells that modeling various objects inside the cabin are identified, so after the parallelization these cells are attributed to the process that deals with the calculation of that subdomain. Thus, the parallelization process does not change the structure to be simulated and the simulation results of the whole structure coincide with the results obtained by the parallel approach.

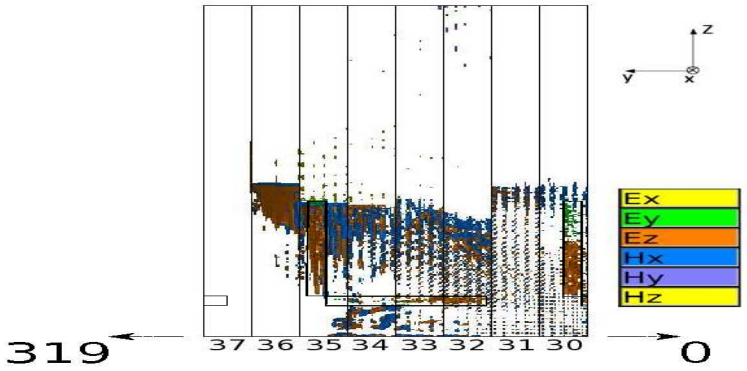
The simulation of this structure, performed on Hyperion supercomputer, uses 40 computing nodes with eight processes per node, considering the RAM memory limits and the performance reasons concerning the CPU time. So the cabin aircraft structure is divided into 320 processes. This is possible because of the RAM resources of 4 GB per core available on a Hyperion node, while the required RAM for each process is about 2 GB. The simulation with 320 processes is performed in only 40 hours.



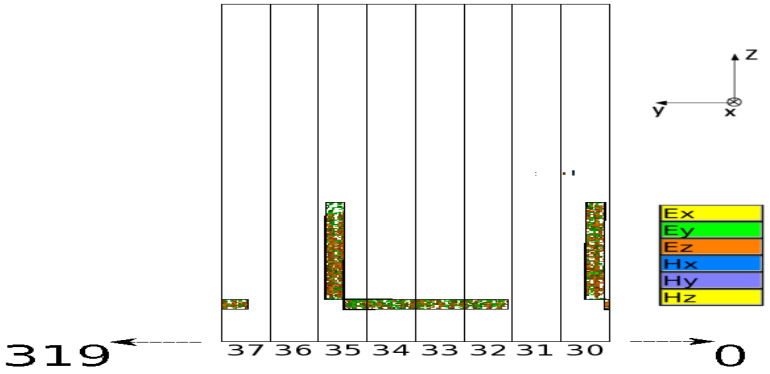
(a)



(b)



(c)



(d)

Figure 6.15: Field propagation inside eight consecutive subdomains corresponding to the processes 30 through 37, at time steps: (a) t= 1549, (b) t = 1716, (c) t = 1876 and (d) t = 29364.

Table 6.2: Summary of Simulation Time on Grid: measured and predicted values

No. of processes/node	No. of nodes	Million TLM cells/process	Time [h] (measured)	Time [h] (predicted)	Error [%]
4	80	5.3	51	50.6	0.7
8	165	1.3	16.5	16.1	2.42
1	1	1683	-	7520	-

In Fig. 6.15 is presented the electromagnetic field propagation by the six components, inside the structure at different time steps. The figures show a sequence of eight consecutive subdomains computed in parallel by different processes from 30 to 37. At time step t=1549, the wave already propagated through the first 30 subdomains and just enters the 31 subdomain managed by the process 30. After exciting the structure, it can be observed that the electric field components are concentrated inside the material of the seats as a standing wave. All the field components are displayed at the same level. The small values of the field are concentrated in the lower part of the structure, even under the seats. As the field grows, it takes higher positions in the structure.

The same simulation is performed also on Grid'5000 platform, using 80 nodes distributed on two sites: Lille and Nancy. Due to the heterogeneity of the grid resources (the cluster Chinqchint has nodes with only 8 GB of RAM while the cluster Griffon has nodes with 16 GB of RAM) only four processes can be released per node.

In Table 6.2 are presented the simulation time values when this structure is calculated on GC using 320 and 1320 processes. In the case of the latter simulation, 165 computing nodes were used, each managing about eight processes due to the smaller volume of the subdomain of each process. Computing nodes are distributed over three sites: Lille, Nancy and Rennes.

Together, are added also the predicted values for the simulation time using the prediction model given by (5.5), based on the application profile related to the TLM SCN discretization cell with 18 values of polarization. The model takes in consideration the cache miss problem. The errors are small and they are due to the shortcoming in the estimate of the throughput and latency variations over the grid computing network, while it is used also by other users. These variations affect the correct estimation of the communication time between processes, which is an important component of the simulation time. Thus, calculating the aircraft cabin structure without parallelization means treating 1.68 billion of TLM cells using a single process. The predicted value of the computation time is about 7520 hours.

On Hyperion supercomputer the simulation time is less than on GC because the Hyperion nodes are more powerful having a Nehalem Intel micro-architecture comparing with the nodes on Grid'5000 clusters which are based on the older Intel Penryn micro-architecture, thus obtaining a lower computing time. Then, concerning the network, Hyperion nodes are connected by Infiniband, which makes the communication time between processes much

shorter.

But, pecuniary aspect is not negligible, so that the access to a supercomputer resource is more expensive than on a grid. Another important aspect for the simulations generating a large amount of data is the generous storage space provided by the supercomputer for these results. GC offers no space for temporary storage of the data obtained from the simulation, so that the user is forced to retrieve data to be analyzed on other support before the computing nodes reservation expires.

The visualization of the propagation of the electromagnetic field inside the aircraft cabin is realized with Vis5D tool, a free OpenGL-based volumetric visualization program for scientific datasets in 3+ dimensions [125], extended for a TLM application in electromagnetism by T. Mangold and W. Dressel at the Institut for Highfrequency, University of Technology, Munich (1997-2003).

The input of this program contain a matrix in five dimensions, generated during the simulation, containing data representing the six components of the electromagnetic field at different time steps at any point of the computing domain discretized with TLM. Depending on TLM model designed to analyze the problem (mesh step, structure dimensions, ie the number of TLM cells in the computing domain) and the number of iterations to be captured, the 5D matrix can have various dimensions. The creation of these files during the execution of the TLM application can greatly influence the simulation time. For the aircraft cabin, the 5D matrix is so large that it can not be stored in RAM memory during the simulation, and requires writing data to the Vis5D file at each iteration that is intended to be visualized.

The process of determining the six field components by means of the voltage impulses of the TLM cell using (4.14) and writing these values at each iteration in a file on the hard disk of the computing node is very costly, the time required for this process being almost equal to the computation time given by the two TLM steps, scatter and connect, ie the writing time is almost 40% from the computation time for each time step when data capture is done. The TLM subdomain corresponding to a process when the cabin structure model is divided in 320 subdomains, each one having the sizes 192 mm, 5610 mm and 2500 mm consists of 5 million TLM nodes when the mesh step is 8 mm. The V5d file size expressed in bytes, corresponding to one iteration is:

$$\frac{5610}{8} \times \frac{2500}{8} \times \frac{192}{8} \times 6 \times 4(bytes)$$

where 6 is the number of field components and 4 is the number of bytes required for representing a field value.

All these files obtained for each process require 3.9 TB of storing space, for 30869 iterations. Storing these files is a problem not only during the simulation, but also after the nodes reservation expires. Regarding the simulation executed on Hyperion nodes, the data write process costs 10 hours, so the total simulation time is 50 hours.

6.4 Conclusions

The parallel hybrid computing approach is used to simulate the propagation of electromagnetic field inside the cabin of an airplane. Due to the complexity and size of this structure, it can not be simulated by means of traditional computing because of time and memory constraints. The estimated value of the simulation time using a single computing machine on Grid'5000 is about 7520 hours. The structure is computed on GC and supercomputer, in only 50 hours and 40 hours respectively, by using 320 processes. The measured values for the simulation time on GC are in agreement with the estimated values by the model prediction. Using GC or the supercomputer depends on the structure needs and also the computing nodes constraints.

Chapter 7

Conclusions and Perspectives

The aim of this chapter is to summarize the work presented in this thesis and to provide a constructive conclusion that would help in the future developement based on this research work.

The main objectif of this thesis was to give a solution to the rigorous calculation of the electromagnetic scattering inside very large and complex structures such as an airplane cabin, process involving the numerical solution of very large systems inaccessible by traditional resources.

In order to accomplish this task, a new TLM/modal hybrid numerical modeling approach is proposed in this thesis. As TLM - one of the most used rigorous numerical methods in electromagnetic modeling - requires important computational resources when dealing with 3D supersized problems, which are not available on traditional computing machines, a parallel implementation solves this issue, taking full advantage of the high performance computing platforms.

The hybrid numerical modeling tool provides important savings on computational resources when deal with the electromagnetic complex structures. Based on a computing domain decomposition, according to diakoptics procedure, the volumes are discretized upon TLM-SCN cells and the planar structures are modelled by modal approach as a multi-port surface impedance. The two methods are coupled by relating the tangential electromagnetic fields at the interface between the two subdomains: the field on the volume subdomain surface to the active modes of the planar subdomains.

Due to the large electrical size of the planar structures, a conventional full-wave TLM analysis, without domain decomposition, would require an enormous amount of computational resources to solve a large number of unknowns related to a small mesh step. So, the hybrid TLM/modal approach is the first step in saving computational resources when dealing with complex structures. This numerical hybrid tool is validated by analytical curves and also by HFSS results.

An original approach which combines hybrid CEM techniques with large scale parallel systems such as GC and supercomputer in order to run fast and full-wave electromagnetic

simulation of large and complex structures, is presented. The TLM algorithm has been parallelized in order to perform simulations on distributed computing architectures. The volumes of the structure to be analyzed are divided in subdomains, each one being asigned to a process on a CPU. During the simulation, the processes communicate between them exchanging intermediate data. The communication is realized through a send/receive of messages, based on a message passing standard.

Grid'5000 and Hyperion supercomputer are the two platforms used for the experiments performed in this thesis. An experiment protocol is proposed in order to easily migrate the computational electromagnetics applications on the computational grid platform, Grid'5000. The speedup curves highlights the role of parallelization scheme on cluster, grid and supercomputer with respect to the size of the problem and its repartition.

Problems related to the floating-point exceptions, memory contention and cache misses were encountered during the experiments and improved, in order to efficiently use the computational resources and also to have a predictible application, in terms of execution time. As the batch scheduler on Grid'5000 clusters requires the execution time of an application when reserving computing nodes, a time prediction model has been implemented step by step to increase the accuracy of the estimated values. The errors are under 10%. The prediction model is based on a hybrid approach, combining a set of past experiments in order to take in consideration the computing platform, with the profile of the application. The model considers also the important cache miss phenomena given by the TLM algorithm when dealing with large structures, related to the hierarchical memory architecture found on the computing nodes.

Rules for the estimation of the required resources to compute a given structure with a certain efficiency have been given, with an error of 16%. The propagation of TE_{10} mode inside a supersized rectancular waveguide, discretized by 1.4 billion TLM cells, requiring more than 130 days of simulation time by a traditional computation, was simulated in 14 hours by 318 MPI processes in parallel, proving the benefits of GC.

Finally, the parallel TLM/modal hybrid approach is used to simulate the propagation of the electromagnetic field inside a real complex and supersized structure - an airplane cabin with seats. In the context where the airplane manufacturers analyze the possibility to provide on board wireless services for their passengers, the model is accomplished for studying the electromagnetic field behavior up to 3.7 GHz, the frequency of WLAN standard 802.11y-2008. Modelled by 1.68 billion of TLM cells, the required memory is about 226 Go and the estimated time required for the computation using a single process is about 7520 hours, or 313 days. The structure is computed on GC and supercomputer, in only 50 hours and 40 hours respectively, by using 320 processes. The measured time values agree with those given by the prediction model.

Using the computational grid or the supercomputer depends on the structure needs and

also the computing nodes constraints. Simulating a structure on a cluster or on a supercomputer is more efficient than the computational grid, because of the network latency which is a critical point. But, the number of resources placed on a cluster or a supercomputer can sometimes be limited according to the structure size and the time constraints. Also, the computational resources are more expensive on a supercomputer, due to the high performance architecture (processors and networks). The computational grid joins together a huge amount of resources which permit the user to deal with large workloads, avoiding the time and memory constraints with low cost.

In perspective, running simulations over several billion of TLM cells and developping a parallel hybrid numerical method coupling TLM to SCT, represents the final goal toward fast and full-wave electromagnetic simulation of complex and electrically large structures. SCT [14] is a frequency-domain approach and can be advantageously applied to the modeling of microwave and millimeter wave circuits with high aspect ratios, MEMS-controlled coupled microstrip reflectarrays and multiscale pre-fractal structures.

In a hybrid numerical method based on domain decomposition approach, SCT can be used to model 2D or 2.5D planar structures while TLM provides the volumes discretization. Using the parallel computing technologies, the TLM subdomains can be divided into small parts to be computed in parallel, in order to speed the simulation.

Also, due to the modular nature of SCT, the modelisation of the planar structures can be parallelized. Even the convergence study (looking for the appropriate number of active and passive modes at each scale of the subdomain modelled by SCT) can be parallelized by running convergence tests as separate processes.

Another exciting opportunity is given by Compute Unified Device Architecture (CUDA) [66], a parallel computing platform and programming model providing large increases in computing performance based on the graphics processing units (GPUs) power. CUDA architecture supports many languages and programming environments, including C, Fortran, OpenCL, and DirectX Compute, which makes easier the application code migration to a GPU platform.

Using a MPI-based parallel TLM/SCT hybrid approach on a CUDA cluster platform represents a very challenging perspective that will drastically increase the computation performance, providing also the possibility to simulate large and complex electromagnetic structures with an accuracy that was not possible until now.

***

Experiments presented in this thesis were carried out using the Grid'5000 experimental testbed, being developed under the INRIA ALADDIN development action with support from CNRS, RENATER and several Universities as well as other funding bodies (see

https://www.grid5000.fr).

Also, this work was performed using HPC resources from CALMIP (Grant 2012 - P0522).

Authors wish to acknowledge the French Midi Pyrénées region for financial support.

References

[1] IEEE 802.11y 2008. http://standards.ieee.org/findstds/standard/802.11y-2008.html, october 2012. 93

[2] M. Ahmadian. *Transmission Line Matrix (TLM) Modelling of Medical Ultrasound*. University of Edinburgh, 2002. ix, 30, 31

[3] Airbus. A350-1000, dimensions and key data. http://www.airbus.com/aircraftfamilies/passengeraircraft/a350xwbfamily/a350-1000/specifications/, october 2012. 93

[4] S. Akhtarzad and P.B. Johns. Solution of maxwell's equations in three space dimensions and time by the t.l.m. method of numerical analysis. *Electrical Engineers, Proceedings of the Institution of*, 122(12):1344 –1348, december 1975. 35

[5] M. Alexandru, T. Monteil, F. Coccetti, P. Lorenz, and H. Aubert. Transmission-line modeling computational electromagnetics on grids. In *Workshop on Present challenges in computational electromagnetics: complexity management, multi-scales, multi-physics, uncertainty management, statistics*, 2010. 20

[6] M. Alexandru, T. Monteil, P. Lorenz, F. Coccetti, and H. Aubert. Efficient large electromagnetic problem solving by hybrid tlm and modal approach on grid computing. In *International Microwave Symposium*, 2012. 20

[7] M. Alexandru, T. Monteil, P. Lorenz, F. Coccetti, and H. Aubert. Electromagnetic modeling of complex structures by tlm/modal hybrid approach with efficient parallel computing. In *European Microwave Conference*, 2012. 20

[8] M. Alexandru, T. Monteil, P. Lorenz, F. Coccetti, and H. Aubert. Large electromagnetic problem on large scale parallel computing systems. In *International Conference on High Performance Computing and Simulation*, 2012. 20

[9] Kevin P. Allen. Efficient parallel computing for solving linear systems of equations. *Journal of Undergraduate Research and Creative Works*, 5:8–17, 2004. 61

[10] R. Allen, A. Mallik, and P. Johns. Numerical results for the symmetrical condensed tlm node. *Microwave Theory and Techniques, IEEE Transactions on*, 35(4):378 – 382, apr 1987. 37

[11] F. Arndt, I. Ahrens, U. Papziner, U. Wiechmann, and R. Wilkeit. Optimized e-plane t-junction series power dividers. *Microwave Theory and Techniques, IEEE Transactions on*, 35(11):1052 – 1059, nov 1987. 41

[12] F. Arndt, U. Tucholke, and T. Wriedt. Computer-optimized multisection transformers between rectangular waveguides of adjacent frequency bands (short papers). *Microwave Theory and Techniques, IEEE Transactions on*, 32(11):1479 – 1484, nov 1984. 41

[13] I. Foster at al. The grid2003 production grid: Principles and practice. *High-Performance Distributed Computing, International Symposium on*, 0:236–245, 2004. 17

[14] H. Aubert. The concept of scale-changing network in global electromagnetic simulation of complex structures. *Progress In Electromagnetics Research B*, 16:127–154, 2009. 8, 103

[15] H. Aubert and H. Baudrand. *L'Électromagnétisme par les schémas équivalents*. Cépaduès-Éditions, Paris, 2003. 55

[16] M. Barton and J.R. Rattner. Parallel computing and its impact on computational electromagnetics. *Magnetics, IEEE Transactions on*, 28(2):1690 –1695, mar 1992. 62

[17] P. Berini and Ke Wu. A pair of hybrid symmetrical condensed tlm nodes. *Microwave and Guided Wave Letters, IEEE*, 4(7):244 –246, july 1994. 36

[18] BOINC. Berkeley open infrastructure for network computing. `http://boinc.berkeley.edu/trac/wiki/DesktopGrid`, october 2012. 18

[19] C. Burguire and C. Rochange. History-based schemes and implicit path enumeration. In *In Proceedings of the 6th Workshop on Worst-Case Execution Time Analysis (WCET06)*, pages 17–22, 2006. 24

[20] G. Burke and E. Miller. Modeling antennas near to and penetrating a lossy interface. *Antennas and Propagation, IEEE Transactions on*, 32(10):1040 – 1049, oct 1984. 10

[21] G. C. Buttazzo. *Hard Real-time Computing Systems: Predictable Scheduling Algorithms And Applications (Real-Time Systems Series)*. Springer-Verlag TELOS, Santa Clara, CA, USA, 2004. 23

[22] R. Buyya, D. Abramson, and J. Giddy. An economy driven resource management architecture for global computational power grids. In *INTERNATIONAL CONFERENCE ON PARALLEL AND DISTRIBUTED PROCESSING TECHNIQUES AND APPLICATIONS*, 2000. 18

[23] Calmip. Hyperion supercomputer. http://www.calmip.cict.fr/spip/spip.php?rubrique90, october 2012. 22

[24] N. Capit, G. Da Costa, Y. Georgiou, G. Huard, C. Martin, G. Mounie, P. Neyron, and O. Richard. A batch scheduler with high level components. In *Proceedings of the Fifth IEEE International Symposium on Cluster Computing and the Grid (CCGrid'05) - Volume 2 - Volume 02*, CCGRID '05, pages 776–783, Washington, DC, USA, 2005. IEEE Computer Society. 20

[25] F. Cappello, E. Caron, M. Dayde, F. Desprez, Y. Jegou, P. Primet, E. Jeannot, S. Lanteri, J. Leduc, N. Melab, G. Mornet, R. Namyst, B. Quetier, and O. Richard. Grid'5000: a large scale and highly reconfigurable grid experimental testbed. In *Grid Computing, 2005. The 6th IEEE/ACM International Workshop on*, page 8 pp., nov. 2005. 18

[26] E. Caron, C. Klein, and C. Perez. Efficient grid resource selection for a CEM application. In *9me Rencontres francophones du Paralllisme, Renpar'19, Toulouse, France, September 9-11, 2009*, 2009. 17

[27] Z. Chen, M.M. Ney, and W.J.R. Hoefer. A new finite-difference time-domain formulation and its equivalence with the tlm symmetrical condensed node. *Microwave Theory and Techniques, IEEE Transactions on*, 39(12):2160 –2169, dec 1991. 11

[28] C. Christopoulos. *The Thransmission-Line Modeling Method*. Oxford University Press, New York, 1995. 8, 39, 40

[29] A. Colin and G. Bernat. Scope-tree: A program representation for symbolic worst-case execution time analysis. In *Proceedings of the 14th Euromicro Conference on Real-Time Systems*, ECRTS '02, Washington, DC, USA, 2002. IEEE Computer Society. 24

[30] DEISA. Distributed european infrastructure for supercomputing applications. http://www.deisa.eu/, october 2012. 17

[31] Jack J. Dongarra, G. A. Geist, Robert Manchek, and V. S. Sunderam. Integrated pvm framework supports heterogeneous network computing. In *Computers in Physics*, pages 166–175, 1993. 15

[32] EGEE. Enabling grids for e-science. http://www.eu-egee.org/, october 2012. 17

[33] EGI. European grid infrastructure. http://www.egi.eu/, october 2012. 17

[34] eScience. http://www.escience-grid.org.uk/, october 2012. 17

[35] A. Esposito and L. Tarricone. Grid computing for electromagnetics: a beginner's guide with applications. *Antennas and Propagation Magazine, IEEE*, 45(2):91 – 100, april 2003. 62

[36] Eswarappa, G.I. Costache, and W.J.R. Hoefer. Transmission line matrix modeling of disperse wide-band absorbing boundaries with time-domain diakoptics for s-parameter extraction. *Microwave Theory and Techniques, IEEE Transactions on*, 38(4):379 –386, apr 1990. 42

[37] M. J. Flynn. Some computer organizations and their effectiveness. *IEEE Trans. Comput.*, 21(9):948–960, 1972. 14

[38] I. Foster and C. Kesselman, editors. *The grid: blueprint for a new computing infrastructure*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1999. 15

[39] I. Foster, C. Kesselman, and S. Tuecke. The anatomy of the grid: Enabling scalable virtual organizations. *Int. J. High Perform. Comput. Appl.*, 15(3):200–222, 2001. 18

[40] Ganglia. http://ganglia.sourceforge.net/, october 2012. 20

[41] Y. Georgiou, J. Leduc, B. Videau, J. Peyrard, and O. Richard. A tool for environment deployment in clusters and light grids. In *Proceedings of the 20th international conference on Parallel and distributed processing*, IPDPS'06, pages 362–362, Washington, DC, USA, 2006. IEEE Computer Society. 20

[42] R. Gibbons. A historical application profiler for use by parallel schedulers. In *In Job Scheduling Strategies for Parallel Processing*, pages 58–77. Springer Verlag, 1997. 23

[43] Globus. Open source grid software. http://www.globus.org, october 2012. 18

[44] Grid5000. https://www.grid5000.fr/mediawiki/index.php/SSH, october 2012. 19

[45] Anshul Gupta and Vipin Kumar. Performance properties of large scale parallel systems. Technical report, Department of Computer Science, University of Minnesota, 1993. 22

[46] Ludovic Hablot, Olivier Glück, Jean-Christophe Mignot, and Pascale Vicat-Blanc Primet. Etude d'implementations MPI pour une grille de calcul. In *RenPar 2008*, Fribourg, Suisse, 2008. 69

[47] R. F. Harrington. *Field Computation by Moment Methods*. Macmillan, New York, 1968. 8

[48] John R. Hauser. Handling floating-point exceptions in numeric programs. *ACM Transactions on Programming Languages and Systems*, 18:139–174, 1996. 74

[49] HEMERA. https://www.grid5000.fr/mediawiki/index.php/Hemera, october 2012. 20

[50] HFSS. High frequency structural simulator. http://www.ansoft.com/products/hf/hfss/, october 2012. 53

[51] W.J.R. Hoeffer. The transmission-line matrix method–theory and applications. *Microwave Theory and Techniques, IEEE Transactions on*, 33(10):882–893, oct 1985. 40, 41

[52] M. Hook. The use of large electromagnetic computer models in emc applications. In *Does Electromagnetic Modelling Have a Place in EMC Design, IEE Colloquium on*, pages 7/1 –7/4, feb 1993. 7

[53] C. Huygens. Traité de la lumière, 1690. Leiden. 29

[54] IEEE. Ieee standard for binary floating-point arithmetic. *ANSI/IEEE Std 754-1985*, pages 0–1, 1985. 74

[55] T. Itoh. *Numerical Techniques for Microwave and Millimeter-Wave Passive Structures*. Wiley, New York, 1989. 8, 41

[56] S. Jingling and N. Zhenyi. Simulation of electromagnetic interference from gsm wireless phones in transport airplanes. In *Antennas Propagation and EM Theory (ISAPE), 2010 9th International Symposium on*, pages 1180 –1183, 29 2010-dec. 2 2010. 88

[57] P. B. Johns and K. Akhtarzad. Use of time-domain diakoptics in time discrete models of fields. *Int. J. num. Meth. Engng.*, 17(1):1–14, 1981. 42

[58] P. B. Johns and K. Akhtarzad. Time domain approximations in the solution of fields by time domain diakoptics. *International Journal for Numerical Methods in Engineering*, 18(9):1361–1373, 1982. 42

[59] P.B. Johns. Application of the transmission-line-matrix method to homogeneous waveguides of arbitrary cross-section. *Electrical Engineers, Proceedings of the Institution of*, 119(8):1086 –1091, august 1972. 34

[60] P.B. Johns. The solution of inhomogeneous waveguide problems using a transmission-line matrix. *Microwave Theory and Techniques, IEEE Transactions on*, 22(3):209 –215, mar 1974. 34

[61] P.B. Johns. The art of modelling. *Electronics and Power*, 25(8):565 –569, august 1979. 5

[62] P.B. Johns. On the relationship between tlm and finite-difference methods for maxwell's equations (short paper). *Microwave Theory and Techniques, IEEE Transactions on*, 35(1):60 – 61, jan 1987. 11

[63] P.B. Johns. A symmetrical condensed node for the tlm method. *IEEE Trans. on Microwave Theory and Tech.*, 35(4):370–377, apr 1987. ix, 36

[64] P.B. Johns and R.L. Beurle. Numerical solution of 2-dimensional scattering problems using a transmission-line matrix. *Electrical Engineers, Proceedings of the Institution of*, 118(9):1203–1208, september 1971. ix, 32, 33, 35

[65] I. Junqua, J. Parmantier, and M. Ridel. Modeling of high frequency coupling inside oversized structures by asymptotic and pwb methods. In *International Conference on Electromagnetics in Advanced Applications (ICEAA)*, pages 68–71, sept. 2011. 88

[66] N.P. Karunadasa and D.N. Ranasinghe. Accelerating high performance applications with cuda and mpi. In *Industrial and Information Systems (ICIIS), 2009 International Conference on*, pages 331–336, dec. 2009. 103

[67] J. B. Keller. Geometrical theory of diffraction. *Journal of the Optical Society of America*, 52(2):116–130, 1962. 9

[68] R.G. Kouyoumjian. Asymptotic high-frequency methods. *Proceedings of the IEEE*, 53(8):864 – 876, aug. 1965. 9

[69] G. Kron. Equivalent circuit of the field equations of maxwell-i. *Proceedings of the IRE*, 32(5):289 – 299, may 1944. 27

[70] G. Kron. *Diakoptics*. MacDonald, London, 1963. 42

[71] Legion. A worldwide virtual computer. `http://legion.virginia.edu/`, october 2012. 18

[72] Jin-Yan Li and Wei Zhang. Research on parallel computing of electromagnetic scattering in complex environment based on cgsp. In *Apperceiving Computing and Intelligence Analysis, 2009. ICACIA 2009. International Conference on*, pages 39 –43, oct. 2009. 62

[73] P. Lorenz, J.V. Vital, B. Biscontini, and P. Russer. A grid-enabled time domain transmission line matrix (tlm-g) system for the analysis of complex electromagnetic structures. In *Workshop on Computational Electromagnetics in Time-Domain(CEM-TD)*, pages 48–51, nov. 2005. 62

[74] L. Medgyesi-Mitschang and J. Putnam. Electromagnetic scattering from extended wires and two- and three-dimensional surfaces. *Antennas and Propagation, IEEE Transactions on*, 33(10):1090 – 1100, oct 1985. 10

[75] R. Mehran. Computer-aided design of microstrip filters considering dispersion, loss, and discontinuity effects. *Microwave Theory and Techniques, IEEE Transactions on*, 27(3):239 – 245, mar 1979. 41

[76] B. Miegemolle and T. Monteil. Hybrid method to predict execution time of parallel applications. In *CSC*, pages 224–230. Proceedings of the 2008 International Conference on Scientific Computing, CSC 2008, July 14-17, 2008, Las Vegas, Nevada, USA, 2008. 24

[77] E.K. Miller. A selective survey of computational electromagnetics. *Antennas and Propagation, IEEE Transactions on*, 36(9):1281 –1305, sep 1988. 5, 7, 9

[78] R. Mittra and W. W. Lee. *Analytical Techniques in the Theory of Guided Waves*. Macmillan, New York, 1971. 41

[79] MPI. The message passing interface standard. http://www.mcs.anl.gov/research/projects/mpi/, october 2012. 15

[80] MPICH2. http://www.mcs.anl.gov/research/projects/mpich2/, october 2012. 15

[81] M. Nadarassin, H. Aubert, and H. Baudrand. Analysis of planar structures by an integral approach using entire domain trial functions. *IEEE Trans. on Microwave Theory and Tech.*, 43(10):2492–2495, oct 1995. 57

[82] NAREGI. National research grid initiative. https://www.naregi.org/ca/index.html, october 2012. 17

[83] P. Naylor. New three dimensional symmetrical condensed lossy node for solution of electromagnetic wave problems by tlm. *Electronics Letters*, 26(7):492 –494, march 1990. 36

[84] J. Nielsen and W. Hoefer. A complete dispersion analysis of the condensed node tlm mesh. *Magnetics, IEEE Transactions on*, 27(5):3982 – 3983, sep 1991. 37

[85] OAR. http://oar.imag.fr/, october 2012. 20

[86] M. F. X. J. Oberhumer. Lzo – a real-time data compression library. http://www.oberhumer.com/opensource/lzo/lzodoc.php, october 2012. 69

[87] A.S. Omar and K. Schonemann. Transmission matrix representation of finline discontinuities. *Microwave Theory and Techniques, IEEE Transactions on*, 33(9):765 – 770, sep 1985. 41

[88] OpenMP. The openmp api specification for parallel programming. http://openmp.org/wp/, october 2012. 15

[89] OpenMPI. http://www.open-mpi.org/, october 2012. 15

[90] P.J. Parsons, S.R. Jaques, S.H. Pulko, and F.A. Rabhi. Tlm modeling using distributed computing. *Microwave and Guided Wave Letters, IEEE*, 6(3):141 –142, mar 1996. 61

[91] L. Pierantoni, C. Tomassoni, and T. Rozzi. A new termination condition for the application of the tlm method to discontinuity problems in closed homogeneous waveguide. *IEEE Trans. on Microwave Theory and Tech.*, 50(11):2513–2518, nov 2002. 42

[92] PlanetLab. http://www.planet-lab.org/, october 2012. 17

[93] PRACE. Partnership for advanced computing in europe. http://www.prace-ri.eu/, october 2012. 17

[94] P. P. Puschner and A. Burns. Guest editorial: A review of worst-case execution-time analysis. *Real-Time Systems*, 18(2/3):115–128, 2000. 24

[95] S.M. Rao. *Time Domain Electromagnetics*. Academic Press, London, 1999. 10

[96] RENATER. French research and education network. http://www.renater.fr/, october 2012. 19

[97] M. Righi, J.L. Herring, and W.J.R. Hoefer. Efficient hybrid tlm/mode-matching analysis of packaged components. *IEEE Trans. on Microwave Theory and Tech.*, 45(10):1715–1724, oct 1997. 42

[98] M. Righi and W.J.R. Hoefer. Efficient 3d-scn-tlm diakoptics for waveguide components. In *Microwave Symposium Digest, 1994., IEEE MTT-S International*, pages 27–30 vol.1, may 1994. 42

[99] M. Righi, M. Mongiardo, R. Sorrentino, and W.J.R. Hoefer. Efficient tlm diakoptics for separable structures. In *Microwave Symposium Digest, 1993., IEEE MTT-S International*, pages 425–428 vol.1, 1993. 41, 42

[100] Peter Russer, Mauro Mongiardo, and Leopold B. Felsen. Electromagnetic field representations and computations in complex structures iii: network representations of the connection and subdomain circuits. *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields*, 15(1):127–145, 2002. 45

[101] P. Saguet. *Analyse des milieux guides: la methode MTLM*. Institut national polytechnique de Grenoble, 1985. 28

[102] P. Saguet and E. Pic. An improvement for the t.l.m. method. *Electronics Letters*, 16(7):247 –248, 27 1980. 40

[103] P. Saguet and E. Pic. Utilisation d'un nouveau type de noeud dans la méthode tlm en 3 dimensions (usage of a new type of node in the 3d tlm method). *Electronics Letters*, 18(11):478–480, 1982. 36

[104] T.K. Sarkar. The conjugate-gradient technique as applied to electromagnetic problems. *IEEE AP-S Newsletter*, 28(4):5–14, 1986. 10

[105] R. Scaramuzza and A.J. Lowery. Hybrid symmetrical condensed node for the tlm method. *Electronics Letters*, 26(23):1947 –1949, nov. 1990. 36

[106] T. Senior. Impedance boundary conditions for imperfectly conducting surfaces. *Applied Scientific Research, Section B*, 8:418–436, 1960. 10.1007/BF02920074. 10

[107] T. Senior. Approximate boundary conditions. *Antennas and Propagation, IEEE Transactions on*, 29(5):826 – 829, sep 1981. 10

[108] seti@home. http://setiathome.berkeley.edu/, october 2012. 17

[109] J.R. Shewchuk. An introduction to the conjugate gradient method without the agonizing pain. *School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213*, 1994. 10

[110] Yi-Chi Shih. Design of waveguide e-plane filters with all-metal inserts. *Microwave Theory and Techniques, IEEE Transactions on*, 32(7):695 – 704, jul 1984. 41

[111] W. Smith, I. Foster, and V. Taylor. Predicting application run times using historical information, 1997. 23

[112] W. Smith, V. E. Taylor, and I. T. Foster. Using run-time predictions to estimate queue wait times and improve scheduler performance. In *Proceedings of the Job Scheduling Strategies for Parallel Processing*, IPPS/SPDP '99/JSSPP '99, pages 202–219, London, UK, UK, 1999. Springer-Verlag. 23

[113] P. P. M. So, C. Eswarappa, and W. J. R. Hoefer. Parallel and distributed tlm computation with singal processing for electromagnetic field modelling. *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields*, 8(3-4):169–185, 1995. 61

[114] F. Stappert, Ermedahl A., and J. Engblom. Efficient longest executable path search for programs with complex flows and pipeline effects, 2001. 24

[115] A. Taflove and S. C. Hagness. *Computational Electrodynamics: The Finite-Difference Time-Domain Method*. Artech House, Norwood, 2005. 8

[116] L. Tarricone and A. Esposito, editors. *Grid computing for electromagnetics*. Artech House, Inc., 685 Canton Street, Norwood, MA 02062, 2004. 15

[117] TeraGrid. https://www.xsede.org/tg-archives, october 2012. 16

[118] M.E. Thomadakis. The architecture of the nehalem processor and nehalem-ep smp platforms. Technical report, Texas A&M University, 2011. 22

[119] V. Trenkic, C. Christopoulos, and T.M. Benson. A graded symmetrical super-condensed node for the tlm method. In *Antennas and Propagation Society International Symposium, 1994. AP-S. Digest*, volume 2, pages 1106 –1109 vol.2, jun 1994. 36

[120] V. Trenkic, C. Christopoulos, and T.M. Benson. New symmetrical super-condensed node for the tlm method. *Electronics Letters*, 30(4):329 –330, feb 1994. 36

[121] V. Trenkic, C. Christopoulos, and T.M. Benson. Development of a general symmetrical condensed node for the tlm method. *Microwave Theory and Techniques, IEEE Transactions on*, 44(12):2129 –2135, dec 1996. 36

[122] Unicore. Uniform interface to computing resources. http://www.unicore.eu/, october 2012. 18

[123] Valgrind. Cachegrind: a cache and branch-prediction profiler. http://valgrind.org/docs/manual/cg-manual.html, october 2012. 73

[124] A. Vernois, P. Vicat-Blanc, F. Desprez, F. Hernandez, and C. Blanchet. Gripps project: Protein pattern scanning in a grid context. In *First European Health Grid Conference (HealthGrid'03)*, 2003. 17

[125] Vis5D. Overview of vis5d. http://vis5d.sourceforge.net/doc/ch01.html, october 2012. 98

[126] Vishwa. Dynamic configuration of p2p grid middleware. http://dos.iitm.ac.in/vishwanew/, october 2012. 18

[127] A. Wexler. Solution of waveguide discontinuities by modal analysis. *Microwave Theory and Techniques, IEEE Transactions on*, 15(9):508 –517, september 1967. 41

[128] R. Wilhelm, J. Engblom, A. Ermedahl, N. Holsti, S. Thesing, D. Whalley, G. Bernat, C. Ferdinand, R. Heckmann, T. Mitra, F. Mueller, I. Puaut, P. Puschner, J. Staschulat, and P. Stenström. The worst-case execution-time problem - overview of methods and survey of tools. *ACM Trans. Embed. Comput. Syst.*, 7(3):36:1–36:53, 2008. 24

[129] D. R. Wilson and T. R Martinez. Improved heterogeneous distance functions. *Journal of Artificial Intelligence Research*, 6:1–34, 1997. 23

[130] I. Wolff, G. Kompa, and R. Mehran. Calculation method for microstrip discontinuities and t junctions. *Electronics Letters*, 8(7):177 –179, 6 1972. 41

[131] E. Wolfgang. On the definition of speedup. In Costas Halatsis, Dimitrios Maritsas, George Philokyprou, and Sergios Theodoridis, editors, *PARLE'94 Parallel Architectures and Languages Europe*, volume 817 of *Lecture Notes in Computer Science*, pages 289–300. Springer Berlin Heidelberg, 1994. 22

[132] XSEDE. Extreme science and engineering discovery environment. https://www.xsede.org/web/guest/overview, october 2012. 17

[133] E. Yamashita. *Analysis Methods for Electromagnetic Wave Problems*. Artech House, Norwood, 1995. 10

[134] H. Youssef, H. Elmokdad, F. Ndagijimana, J. Jomaah, and M. Zoater. A three-dimensional transmission line matrix method (tlm) in cylindrical coordinates. In *Information and Communication Technologies, 2006. ICTTA '06. 2nd*, volume 2, pages 2207 –2212, 0-0 2006. 40

[135] M. Youssef and L. Vahala. Effects of passengers and internal components on electromagnetic propagation prediction inside boeing aircrafts. In *Antennas and Propagation Society International Symposium 2006, IEEE*, pages 2161 –2164, july 2006. 88

[136] M. Youssef, L. Vahala, and J.H. Beggs. Electromagnetic propagation of wireless networks in aircraft cabins. In *Wireless Communications and Applied Computational Electromagnetics, 2005. IEEE/ACES International Conference on*, pages 832 – 836, april 2005. 88

[137] W. Yu, R. Mittra, T. Su, Y. Liu, and X Yang. *Parallel Finite-Difference Time-Domain Method*. Artech House Electromagnetic Analysis Series. Artech House, 2006. 14

[138] M.O. ZYALIN, F. AKLEMAN, and L. SEVGI. Comparison of tlm and fdtd techniques in rcs simulation and antenna modeling. *Proc. Of AP 2000 (ICAP + JINA) Millenium Conference on Antennas and Propagation*, april 2000. 11

PUBLICATIONS

International Conference Publications

- M. Alexandru, T. Monteil, F. Coccetti, P. Lorenz and H. Aubert, Transmission-Line Modeling Computational Electromagnetics On Grids, Workshop on Present challenges in computational electromagnetics: complexity management, multi-scales, multi-physics, uncertainty management, statistics, St Malo, France, 2-3 dcembre 2010.
- M. Alexandru, T. Monteil, P. Lorenz, F. Coccetti and H. Aubert, Efficient Large Electromagnetic Problem Solving by Hybrid TLM and Modal Approach on Grid Computing, International Microwave Symposium, Montréal, Canada, 17-22 june 2012.
- M. Alexandru, T. Monteil, P. Lorenz, F. Coccetti and H. Aubert, Large Electromagnetic Problem on Large Scale Parallel Computing Systems, International Conference on High Performance Computing & Simulation, Madrid, Spain, 02-06 July 2012.
- M. Alexandru, T. Monteil, P. Lorenz, F. Coccetti and H. Aubert, Electromagnetic Modeling of Complex Structures by TLM/Modal Hybrid Approach with Efficient Parallel Computing, European Microwave Conference, Amsterdam, Netherlands, 28 October - 02 November 2012.