



HAL
open science

Large electromagnetic problem on large scale parallel computing systems

Mihai Alexandru, Thierry Monteil, P Lorenz, Fabio Coccetti, Hervé Aubert

► **To cite this version:**

Mihai Alexandru, Thierry Monteil, P Lorenz, Fabio Coccetti, Hervé Aubert. Large electromagnetic problem on large scale parallel computing systems. International Conference on High Performance Computing and Simulation (HPCS 2012), Jul 2012, Madrid, Spain. 7p., 10.1109/HPC-Sim.2012.6266968 . hal-01228320

HAL Id: hal-01228320

<https://hal.science/hal-01228320v1>

Submitted on 12 Nov 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Large Electromagnetic Problem on Large Scale Parallel Computing Systems

M. Alexandru^{1,2}, T. Monteil^{1,3}, P. Lorenz⁵, F. Coccetti^{1,4} and H. Aubert^{1,2}

¹CNRS, LAAS, 7 avenue du colonel Roche, F-31400 Toulouse, France

²Univ de Toulouse, INP, LAAS, F-31400 Toulouse, France

³Univ de Toulouse, INSA, LAAS, F-31400 Toulouse, France

⁴Univ de Toulouse, LAAS, F-31400 Toulouse, France

⁵Lorenz Solutions, 83661 Lenggries, Germany

mihai.alexandru@laas.fr

Abstract—This paper deals with the electromagnetic modeling of large and complex electrical structures by means of large scale parallel systems, such as Grid Computing (GC) and super-computer. Transmission-Line Matrix (TLM) modeling method is applied to homogeneous volumes. The planar structures are modelled with the mode matching approach. The results prove the benefits of the GC and supercomputer environments to solve electrically large structures. A prediction model for computing performances on grid, based on a hybrid approach that combines a historic-based prediction and an application profile-based prediction, has been developed. The predicted values are in good agreement with the measured values.

Index Terms—Transmission line matrix methods, mode matching methods, grid computing, high performance computing, large-scale systems, prediction model.

I. INTRODUCTION

One of the biggest problem that science faces today is the huge amount of data to be processed very often in a entangled interdisciplinary context. Mathematics, physics, medicine, informatics or the other areas derived from them, are all dependent on usage of numerical resources. Therefore, 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, 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 or communication research are HPC applications and require large amounts of computational power that even supercomputers cannot fulfill. Here, we are interested to show the benefits and the performances of GC infrastructure applied to computing electromagnetics, a scientific domain that

matches the continuous Maxwell's equations to the numerical resources, a fundamental approach in the design of antennas, radars, satellites, computer chips, optical fiber system, mobile phone systems, etc.

This work deals with the large scale architectures numerical computation performance assessment of the electromagnetic larger structures. The approach is performed with a rigorous TLM/modal hybrid calculation of the electromagnetic field scattering phenomena inside larger structures, such as tunnels, airplane fuselage, on different computing environments: super-computer, cluster and grid. A prediction model for the computing time is presented. The results obtained prove the suitability of distributed memory architectures to handle the computing of electromagnetics applications. The optimal exploitation of the number of resources and bandwidth, for a given type of problem, demonstrates the advantage of tailoring given the scientific problem to available architectural opportunities.

In Section II, works related to this are illustrated. The hybrid numerical method approach between TLM and Mode Matching involved in our work is defined in Section III. Moreover the parallelization of the hybrid approach to be executed on parallel computing platforms is outlined in Section IV. Section V presents a prediction model for the computing time. The simulations done and the results derived are outlined in Section VI.

II. RELATED WORK

In [1], 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 [2]. The effects on computing performance when TLM calculation is distributed across a network of workstations using parallel virtual machine (PVM), are analysed in [3]. In order to illustrate the improvements in computational electromagnetics that are achievable by massively parallel processing, a parallel finite element code is used to model a low frequency magnetics problem [4]. 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 environment, based on method of moment [5]. A guide [6] introducing CEM researchers in the field of computational grid, demonstrates that grid computing is a viable environment for parallel computing using a 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 [7] based on Transmission-Line Matrix (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.

A hybrid numerical tool, similar to that presented in this paper, modeling the scattering phenomena by diakoptics procedure, is given in [8] and [9], where the discontinuity is discretized upon TLM and the homogeneous space by modal approach, given the low computing resources. In [10], the compromise done is reducing the TLM discretized space by setting the modal absorbing boundaries a few cells away from the discontinuities and considering only the first 4 or 5 modes. A numerical approach, based on an asymptotic code and a PWB method, to evaluate the high frequency coupling in a complex oversized structure is described in [11].

III. TLM/MODAL HYBRID APPROACH

Computing the electromagnetic field inside a three-dimensional domain means solving Maxwell's equations. Computational electromagnetics are positioned because of the limitation of the continuous field functions from Maxwell's equations to only some samples defined at discrete points.

The calculation of electromagnetic structures with three dimensional large geometries inside and very fine detail discontinuities is a big challenge for manufacturers today and for scientists too. The space and time modeling of such structures with the TLM, 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, given the amount of memory resources needed.

The realization of a hybrid algorithm coupling the time-domain TLM method and a modal approach, represents a solution to this problem. Modeling a structure with two numerical methods, one temporal and one modal, means decomposing the problem into two sub-structures. A subdomain discretized with TLM and the other modeled with the modal approach. This decomposition is known in the literature as time-domain diakoptic method. Subdomains are computed independently and connected together by convolution product at each time step between the impulse response of subdomain discretized with the TLM and numerical Green functions for the subdomain calculated analytically with modal approach. The TLM/modal hybrid approach presented in this work discretises the

electromagnetic field inside volumes with TLM and models the field on multiscale planar structures by modal approach.

Transmission-Line Matrix (TLM) numerical method, [12]-[13], realizes the discrete model for a finite numbers of discretization points in time, of the physical domain, by filling the space with a network of transmission-lines crossed by electrical signals whose voltage and current correspond to the electric and magnetic fields. The intersection of these lines, that have the free-space impedance, is modelled with Symmetrical Condensed Node (SCN) [14]) scheme, whose scattering matrix is derived directly from the behavior of the fields. The SCN node is formed by six branches consisting of two uncoupled transmission lines arranged in space quadrature, representing two polarizations. So, each discrete node is represented by 12 values.

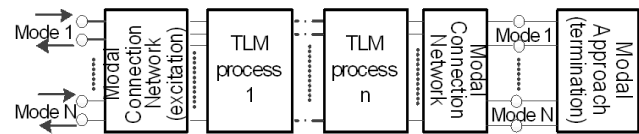


Fig. 1. Schematic view of the TLM/modal hybrid modeling approach.

The TLM numerical method discretize the space in which the electromagnetic field propagates, along the three axes, O_x , O_y , O_z . Small parallelepipeds resulting from the division of space are called TLM-SCN (Symmetrical Condensed Node) cells. The tangential planes between neighboring cells define the gates. The transmission lines join the adjacent cells by coupling the centers of the cells by gates. The center of a TLM cell is known as a scattering center. SCN with *stubs* models nonhomogeneous media.

This field modeling propagating in a certain environment becomes possible because of the equivalence between the electric and magnetic fields and the voltages and currents respectively, in a network of transmission lines.

In order to reduce the numerical dispersion when dealing with the propagation wave modeling, the mesh step has to be as small as possible. But, considering also the limit of the computational power available, the mesh step is chosen equal at least to the wavelength divided by ten. The space modeled with TLM is a discrete system whose solutions approach the solutions of a continuous system gradually when the discretization step tends to zero.

In one calculation, the TLM simulation of a structure provides much information: the impulse response of the structure, the response of the structure for any excitation, the characteristics of propagating modes by Fourier transform. The TLM method requires much computing resources, but its algorithm has the advantage of being parallelizable, which makes it possible to simulate oversized structures on multiple machines.

Mode-Matching method [13] 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, amputation of the series is a major issue in the validity of results.

A general view over the TLM/modal hybrid approach can be observed in Fig. 1. The unterminated boundaries of the discretized TLM domain are the excitation/termination (input/output) reference planes. These reference planes are connected with a Modal Connection Network (MCN) to the modal representation of the EM field. The MCN converts the modal EM representation to the TLM representation, i.e., the modal form of the EM field to the voltage pulses propagating on TLM link lines having the free space impedance.

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 MCN can be represented by a transformer network [15], or equivalently, by a coupling matrix that converts the EM modal field to the TLM representation. The free space impedance has been used as the reference impedance of all ports of the MCN network.

The transformer network presented in [15] 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 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 (such a termination would be 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. This network can be seen as a generalized Wilkinson power divider.

The hybrid approach presented in this paper is validated by the electromagnetic simulation of the TE_{10} mode propagation inside a lossless rectangular waveguide. The termination of the waveguide is modeled by the amplitude of the reflected mode obtained as a convolution product between the modal impulse response of the discontinuity and the amplitude of the incident mode.

IV. PARALLELIZATION OF THE APPLICATION

The performance of a parallel system can be illustrated by several indicators [16]. The speedup is a parameter used to evaluate the gain in terms of simulation time with N parallel computing processors compared to the same calculation on a single processor. The speedup is defined as the ratio between the time of the sequential and the parallel execution. The total simulation time equals the summation of the communication time and computing time of the longest task. The computational efficiency is the ratio between the speedup and the number of processes.

In order to avoid a heavy TLM calculation, the discretized domain can be divided into several sub-domains that will be computed in parallel on multiple CPUs that communicate between them to achieve the job. Architectures with distributed memory make possible the connection between CPUs by a communication network.

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.

MPI (Message-Passing Interface) 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.

The parallel hybrid approach, based on message passing standard, is designed for SPMD (Single Program Multiple Data) programming model. Thus, all tasks run the same program on different data. The user specifies the number of sub-domains the structure will be decomposed, i.e. the number of processes MPI. Each sub-domain 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 in a synchronous mode. 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. In order to simulate as fast as possible, only two processes are executed on the computing node, each on a different processor.

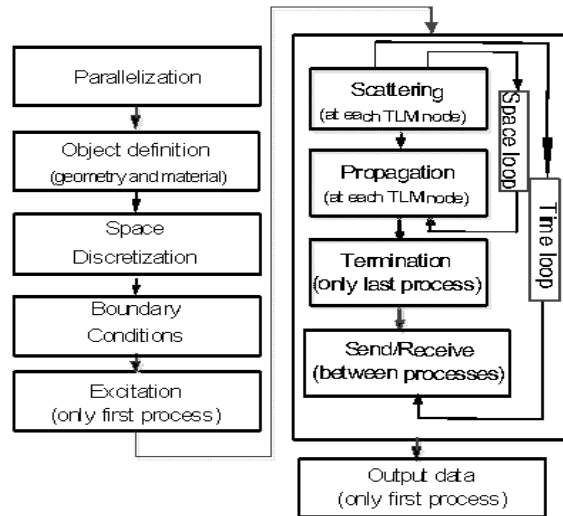


Fig. 2. Schematic view of the parallel hybrid approach implementation.

A schematic view of the parallel hybrid approach implementation can be seen in Fig. 2. On each process, the sub-structures 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 *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.

During the simulation, the processor passes more than 90% of computation time on the TLM's core. In *Scattering* block, the scattering phenomena occurred 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.

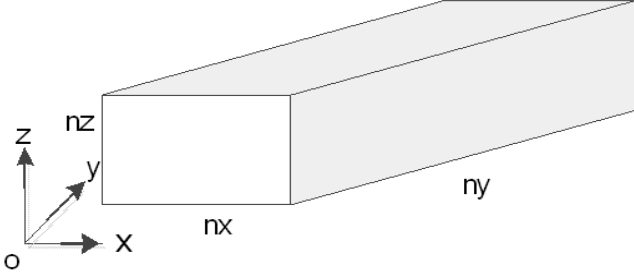


Fig. 3. 3D view of the structure (n_x , n_y , n_z - TLM cell number on the three cartesian directions).

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.

Listing 1. Propagation algorithm

```
connect()
{
  //declaration of variables
  nxy = nx*ny;
  for (int k = 0; k < nz; k++) {
    for (int j = 0; j < ny; j++) {
      for (int i = 0; i < nx; i++) {
        ind = _IND3D_(i, j, k);
        // i + j*nx + k*nx*ny
        if (i < nx-1) {
          ind_ip = ind + 1;
          // ... connection on X axis
        }
        if (j < ny-1) {
          ind_jp = ind + nx;
          // ... connection on Y axis
        }
        if (k < nz-1) {
          ind_kp = ind + nxy;
          // ... connection on Z axis
        }
      }
    }
  }
}
```

```
}
}
```

Let's take a look on *Propagation* block implementation, outlined in Listing 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. 3. 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. 4. Let's consider the discrete coordinate of the current cell as *ind*. The jumpers required to make the connections of the node with index *ind* with his neighbors are: *ind_ip*, *ind_jp*, *ind_kp*.

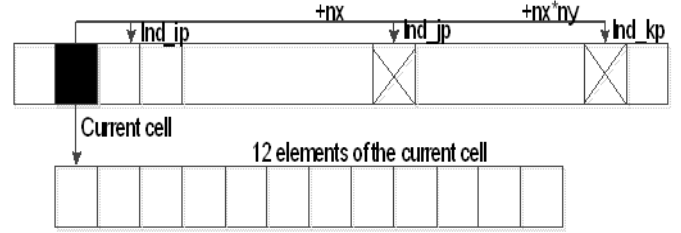


Fig. 4. cell structure.

In order to not charge twice the same data in 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: $2 * n_x * n_y * 12 * 8$ 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); n_x , n_y are the number of TLM cells on the X and Y axes; 12 represents the values characterizing the field polarizations on each cell; 8 is the number of bytes required to stock each value in memory.

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 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.

V. PREDICTION MODEL

The techniques for time prediction fall into two families: the historic-based prediction and the profile-based prediction. The final object of prediction techniques is to estimate the execution time of an application before it is launched. The historic-based prediction technique is used to predict the sequential or parallel time of the applications to be launched on clusters or grid computing. This technique estimates the execution time of an application according to the context of the execution, such as the architecture to run on and the input data, and to the past experiments. The profile-based prediction technique is used to determine the worst case execution time (WCET)

of an application. This technique estimates the WCET based on the analysis of the application code.

In order to reserve resources on the grid computing to launch simulations with TLM/modal hybrid application, a computation time prediction model based on a hybrid approach has been developed. The prediction model presented in this work 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 Grid5000 platform. To measure only the computation time for simulations that are included in historic database, they were executed sequentially to avoid the communication time.

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

$$T_{cal} = c_1 + n_x n_y n_z t c_2, \quad (1)$$

where n_x, n_y, n_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 corresponding to different blocks in the code of the application. The coefficients are determined by a linear programming formulation of a system of historic experiments.

To have a good agreement between the measured and predicted values for the computation time, the problem of the cache misses presented in Section IV 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 time estimation for structures whose space required to store $2 * n_x * n_y * 12 * 8$ bytes does not exceed the processor cache size. In this case the number of cache misses has not an important variation. Database forming simulations of this model keep the same condition. The second model estimates the computation time for structures whose space required to store $2 * n_x * n_y * 12 * 8$ bytes exceeds the processor cache size. This model takes into account the cache misses phenomena during the simulation of large structures. Database forming simulations 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 $2 * n_x * n_y * 12 * 8$ bytes is limited to 6MB.

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:

$$T_{com} = lat + msg/deb \quad (2)$$

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:

$$T_n = c_1 + n_x n_y (n_z/n) t c_2 + 4t T_{com} \quad (3)$$

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

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

$$n \leq \frac{c_1^A + n_x n_y n_z t (c_2^A - e c_2^B)}{e (c_1^B + 4t T_{com})}, \quad (4)$$

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

VI. RESULTS

Grid5000 platform [17] - the French national research infrastructure for large scale parallel and distributed computing, can be used to run large TLM applications. The objective of this platform is to provide an experimental grid configurable, controllable and geographically distributed on 9 sites in France. These sites are connected by RENATER, the French network for research and teaching. This platform allows researchers to test algorithms for parallel programming.

The computing resources of Grid5000 can be managed by two tools: OAR and Kadeploy. OAR 2 is a batch management system that allows users to reserve computing nodes on clusters of Grid'5000. On each cluster from each site, there is a NFS server with a OAR server, which is responsible for the resource management. OAR Grid is a meta-scheduler that allows users to submit applications distributed across multiple sites. Kadeploy is a tool used to deploy a custom image for Linux kernel containing the application to address on the computing nodes. The computing nodes chosen for simulations in this work are placed on four different clusters of Grid5000 (*Griffon, Chinqchint, Paradent, Parapide*)[18].

Hyperion is a supercomputer which belongs to the Calmip group. This machine was ranked 223th 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 computing resources are managed through PBS scripts.

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 - Grid5000 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 displays the speedup versus the number of processes in case of a simulation with a large waveguide structure having the dimensions: 345 mm width, 173 mm height, 2432 mm length and a mesh step of 1mm. The speedup values are obtained from simulations performed on Grid5000 platform and Hyperion supercomputer. The speedup values obtained on

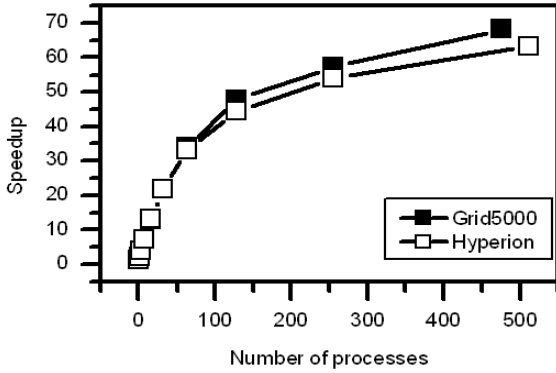


Fig. 5. Parallel computation Speedup for a waveguide modelled with 145 million of TLM cells, obtained on different environments.

grid are bigger than those from the supercomputer Hyperion because the grid processors are slower, which means that the computation time is higher.

Two structures with the dimensions 345 mm width, 173 mm height, 2432 mm length and a mesh step of 1mm, respectively 172 mm width, 86 mm height, 2432 mm length and a mesh step of 1mm have been computed on grid, using different number of processes. Fig. 6 displays the computation times versus the number of processes. The predicted values are in excellent agreement with the measured values. The measured values are obtained in sequential simulations done on cluster Griffon from Nancy site of Gid5000 platform. The two prediction models presented in Section V, of the form (1) are used to determine the estimated values for the computation times. The average error for the first structure is 0.94% while

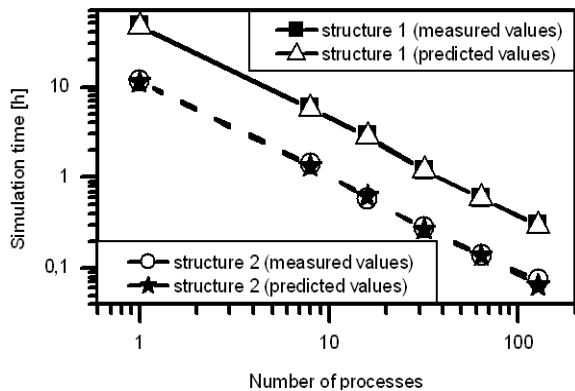


Fig. 6. Computation time on grid for different TLM cell numbers: measured values versus predicted values.

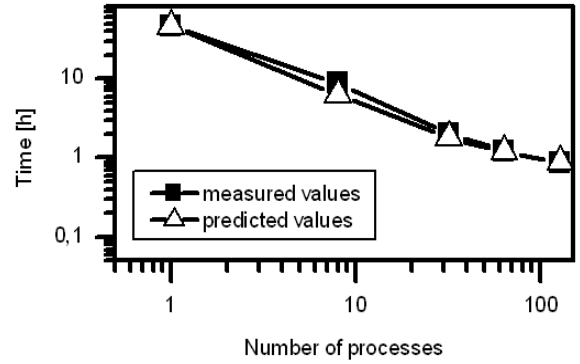


Fig. 7. Total simulation time on grid for different TLM cell numbers: measured values versus predicted values.

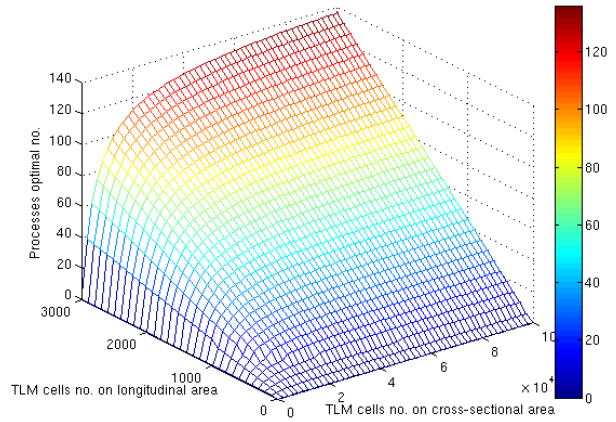


Fig. 8. Processes optimal number.

the maximum error is 1.76%. For the second structure, the average error is 2.18% and the maximum error is 6.14%.

Fig. 7 shows the simulation times on grid versus the number of processes, for various numbers of TLM cells. The predicted values are presented too. The communication time between the processes, given by (2) has been considered. The predicted values are in excellent agreement with the measured values. The average error is 9.2%.

In Fig. 8 it has been estimated the number of processes that are necessary for grid calculation of a problem with the efficiency of 43%, according to the prediction model given in 4. 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 in Section V, depending on the size structure, ie the memory space required for storing $2 * n_x * n_y * 12 * 8$ bytes. For the first structure, 107 processes are estimated, with an error of 16%, compared to the measured value, 128.

VII. CONCLUSIONS

This paper presents an original approach which combines hybrid CEM technique with grid computing in order to speed up the modeling of large electromagnetic problems. The study

has highlighted the role of parallelization scheme, grid versus supercomputer, with respect to the size of the problem and its repartition. The analysis of the simulation performances has allowed to extract practical rules for the estimation of the required resources for a given problem. The computing time prediction model outlined in this work (1), let us estimate the ressource reservation time to simulate a given structure on grid. Also, characterizing the performance of the simulations on the computing grid, rules for the estimation of the required resources have been given in (4).

REFERENCES

- [1] K. P. Allen, "Efficient parallel computing for solving linear systems of equations," *Journal of Undergraduate Research and Creative Works*, vol. 5, pp. 8–17, 2004.
- [2] 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*, vol. 8, no. 3-4, pp. 169–185, 1995. [Online]. Available: <http://dx.doi.org/10.1002/jnm.1660080304>
- [3] P. Parsons, S. Jaques, S. Pulko, and F. Rabhi, "Tlm modeling using distributed computing," *Microwave and Guided Wave Letters, IEEE*, vol. 6, no. 3, pp. 141–142, mar 1996.
- [4] M. Barton and J. Rattner, "Parallel computing and its impact on computational electromagnetics," *Magnetics, IEEE Transactions on*, vol. 28, no. 2, pp. 1690–1695, mar 1992.
- [5] J.-Y. Li and W. 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*, oct. 2009, pp. 39–43.
- [6] A. Esposito and L. Tarricone, "Grid computing for electromagnetics: a beginner's guide with applications," *Antennas and Propagation Magazine, IEEE*, vol. 45, no. 2, pp. 91–100, april 2003.
- [7] P. Lorenz, J. 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)*, nov. 2005, pp. 48–51.
- [8] M. Righi and W. Hoefer, "Efficient 3d-scen-tlm diakoptics for waveguide components," in *Microwave Symposium Digest, 1994., IEEE MTT-S International*, may 1994, pp. 27–30 vol.1.
- [9] M. Righi, J. Herring, and W. Hoefer, "Efficient hybrid tlm/mode-matching analysis of packaged components," *IEEE Trans. on Microwave Theory and Tech.*, vol. 45, no. 10, pp. 1715–1724, oct 1997.
- [10] 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.*, vol. 50, no. 11, pp. 2513–2518, nov 2002.
- [11] 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)*, sept. 2011, pp. 68–71.
- [12] W. Hoefler, "The transmission-line matrix method—theory and applications," *Microwave Theory and Techniques, IEEE Transactions on*, vol. 33, no. 10, pp. 882–893, oct 1985.
- [13] T. Itoh, *Numerical Techniques for Microwave and Millimeter-Wave Passive Structures*. New York: Wiley, 1989.
- [14] P. Johns, "A symmetrical condensed node for the tlm method," *IEEE Trans. on Microwave Theory and Tech.*, vol. 35, no. 4, pp. 370–377, apr 1987.
- [15] P. Russer, M. Mongiardo, and L. 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*, vol. 15, no. 1, pp. 127–145, 2002. [Online]. Available: <http://dx.doi.org/10.1002/jnm.435>
- [16] A. Gupta and V. Kumar, "Performance properties of large scale parallel systems," Department of Computer Science, University of Minnesota, Tech. Rep., 1993.
- [17] 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*, nov. 2005, p. 8 pp.
- [18] Grid5000. (2012) Grid5000:hardware. [Online]. Available: <https://www.grid5000.fr/mediawiki/index.php/Special:G5KHardware>