Proceedings of ISAP2007, Niigata, Japan

Comparison of Parallel Solution of Electrically Large Structures on a Cluster and Multi-core Workstation

#Drazen S. Sumic [1], Branko M. Kolundzija [2]
[1] Research and Development, WIPL-D d.o.o.
Bul. Mihajla Pupina 10E/1, 11000 Belgrade, Serbia, drazen.sumic@wipl-d.com
[2] Department of Electrical Engineering, University of Belgrade
Bul. Kralja Aleksandra 73, 11000 Belgrade, kol@etf.bg.ac.yu

1. Introduction

Electromagnetic simulation of electrically large arbitrarily-shaped three-dimensional structures is computationally very demanding. Rigorous Method of Moments (MoM) analysis based on surface integral equations using higher order basis functions and quadrilateral mesh [1] requires only about 30 unknowns per wavelength squared for metallic structures. This efficient technique, applied in WIPL-D 3D EM solver [2], enables simulation of large and complex structures on PC computers equipped with widely-spread single-core processors. Electrical size limits imposed by the amount of RAM can be extended by application of the out-of-core solver which uses the hard drive for storage during computations. However, simulation becomes prohibitively long for extremely large problems. Therefore, a logical solution to further increase electrical size capacity is parallelization.

In this paper, we present an overview of parallelization of WIPL-D 3D EM solver on two platforms: 8-node computer cluster and a workstation equipped with a quad-core processor. Efficiency of LU decomposition is investigated on both platforms. A model of a helicopter up to 51 λ long was used as a benchmark example for illustrating parallelization efficiency. Radar cross section of a 51 λ long helicopter is calculated in 72 minutes on a 8-node cluster, while the same model requires 2.5 hours using out-of-core solver on the quad-core workstation.

A model of a 180 λ long payload fairing with a dipole in its vicinity was simulated on the quad-core workstation, using the adaptive expansion order reduction presented in [3]. The order reduction allows modeling of an electrically large antenna placement setup with several times less unknowns than in case of rigorous formulation, with only a slight loss of accuracy. The simulation was done in 10.2 hours using the out-of-core solver. On a lower frequency, the payload fairing model measured 144 λ in length, and took 97 minutes on a 8-node cluster.

2. Hardware/Software Platforms

The cluster consists of 9 computers organized in a master-node and 8 computing nodes. Both master and computing nodes have identical hardware configuration that includes:

1. CPU – Pentium 4 P4-630 at 3 GHz clock
2. RAM – 2 GB DDR2

The nodes are interconnected by a Gigabit Ethernet network. The operating system is Red Hat Enterprise Linux HPCC.

The quad-core workstation, running Windows XP x64 edition, comprises:

1. CPU – Intel Core 2 Quad at 2.66 GHz clock
2. RAM – 4 GB DDR2

ISBN: 978-4-88552-223-9 C3055©IEICE

3. Parallelization Overview

When MoM is applied in electromagnetic simulation, the starting linear operator equation is reduced to a system of linear equations. Thus, simulation time consists of the system matrix fill-in time, time needed to solve the linear system and data-post-processing time. For a large number of unknowns, matrix fill-in time and data post-processing time can be disregarded, so analysis time can be approximated with:

$$T = \frac{1}{3} BN^3 \tag{1}$$

where B is the time needed to perform one basic operation (one addition and one multiplication) while solving the linear system, and N is the number of unknowns.

By performing calculations in parallel, coefficient B from (1) is diminished, thus lowering the simulation time. Parallel simulation time can be represented as

$$T_P(N,P) = T_{calc}(N,P) + T_{I/O}(N,P) + T_{comm}(N,P) \tag{2}$$

where T_{calc} is time used for computation, $T_{I/O}$ is time used for input/output operations and T_{comm} is time used for communication among processes. Well balanced load and efficient parallel algorithm minimize the time spent for communication, thus enhancing the efficiency of the parallel program.

The computer cluster is a distributed memory system. Therefore, the data (system matrix) is distributed onto cluster nodes during the matrix fill-in phase of simulation, according to the 2D block-cyclic ditribution scheme [4]. ScaLAPACK [5] LU decomposition routines are used for solving the system of linear equations.

The workstation is a shared memory system, and parallelization is done in the form of multithreading. Each program thread is assigned to a processor core. The matrix fill-in stage is not parallelized. Threaded LAPACK routines [5] are used for solving the system of linear equations in-core and out-of-core.

4. Models

Helicopter model is shown in Figure 1. The helicopter length is 19 meters, maximum fuselage width is 3 meters and main rotor span is 15.3 meters. The model is excited with a plane wave in its cabin symmetry plane (splitting the cabin into two symmetrical parts) at an elevation angle of 45 degrees and circular polarization. Using rigorous MoM formulation the model requires 38975 unknowns at 800 MHz.

Payload fairing model previously analyzed in [3] is shown in Figure 2. Platform length is 8.66 meters, while the largest cross-section diameter is 2.9 meters. A half-wavelength dipole is placed at a quarter-wavelength distance from the platform, lying in the platform cross-section plane. One symmetry plane was applied during simulation. At 6.25 GHz the model requires 420763 unknowns with rigorous MoM formulation.

5. Numerical Results and Conclusions

Below 15000 unknowns, efficiency of the parallel LU decomposition, performed on the 8-node cluster is calculated according to [4] as

$$E(N,P) = \frac{T_\sigma(N)}{P \cdot T_\pi(N,P)} \tag{3}$$

where N is the number of unknowns, $T_\sigma(N)$ is runtime of the serial LU solver on one processor of the cluster, and $T_\pi(N,P)$ is runtime of the parallel LU solver on P processors. The results are shown in Figure 3. Above 15000 unknowns, efficiency is calculated based on the extrapolated serial solution time, due to the RAM limit of a single cluster node. Extrapolation was based on the calculated value of B from (1), assuming that (1) holds for structures above 15000 unknowns.

Efficiency of the in-core LU decomposition, performed on the workstation, is also calculated according to (1), where $T_\sigma(N)$ is runtime of the routine when only one processor core is employed, while $T_\pi(N,P)$ is runtime when P processor cores are employed. The results up to 21000 unknowns (maximum RAM capacity of the workstation) are shown in Figure 4.

From Figures 3 and 4 we can see that parallel LU decomposition is much more efficient on the shared memory workstation than on the distributed memory cluster. Hence, as the number of employed processor cores rises, the efficiency doesn't decrease. On the cluster, efficiency decreases as the number of processors increases.

Measured simulation times of the helicopter model with serial solver and parallel solver on the computer cluster are shown in Figure 5. The serial solver applies LAPACK LU decomposition routine [5], which is the serial version of the routine used in the parallel solver.

Measured simulation times of the helicopter model employing one and multiple processor cores on the workstation are shown in Figure 6.

From Figures 5 and 6 we conclude that as long as there is enough RAM to use the in-core solver, the quad-core workstation outperforms the 8-node cluster. Moreover, the price of such workstation is 4-5 times lower than the price of the 8-node cluster.

Cluster RAM capacity allows simulation of structures of up to 40000 unknowns using the in-core solver. The simulation times achieved using 4 and 8 processors are shown in Figure 7.

Structures above 20000 unknowns can be solved on the workstation only by using the out-of-core solver. The out-of-core solver writes out parts of the system matrix onto the hard drive, and then reads them in during computation. Multithreading is achieved by using multithreaded LAPACK [5] routines. The simulation times are shown in Figure 8.

When using the out-of-core solver, performance of simulation on the workstation deteriorates to a certain extent. The workstation takes 9040 seconds to solve a 38919 unknown structure, while the 8-node cluster takes 4284 seconds for the same task. However, due to price difference between the platforms, the workstation is still the favourite.

Payload fairing model was simulated at 6.25 GHz. By adaptively reducing the expansion order, the number of unknowns was decreased from 420763 down to 46684. The simulation was done in 10.2 hours on the workstation, by using the out-of-core solver. The RCS in two axial planes is given in Figures 9 and 10.

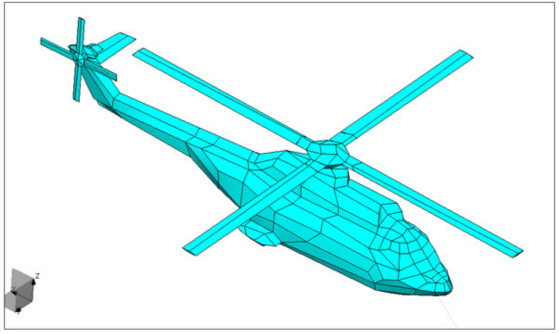

Figure 1: Helicopter model

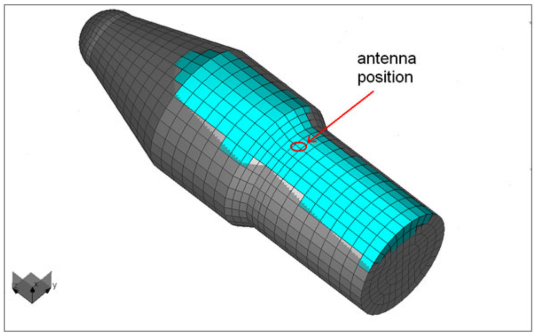

Figure 2: Payload fairing model

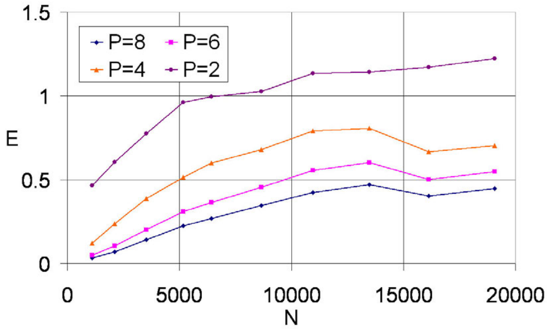

Figure 3: Efficiency of the LU decomposition on a 8-node cluster

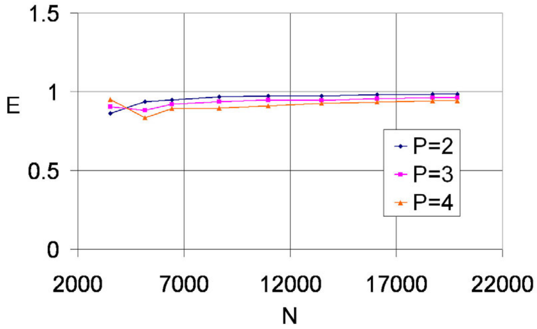

Figure 4: Efficiency of the in-core LU decomposition on the quad-core workstation

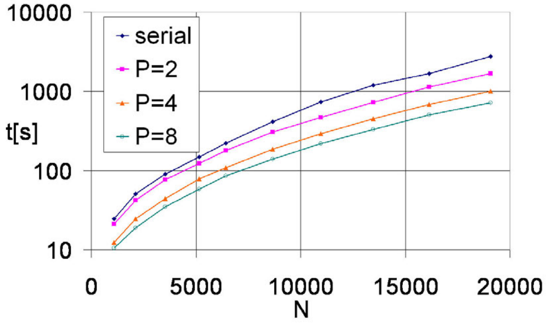

Figure 5: Simulation time for serial execution and parallel for 2, 4, and 8 processors on the cluster – under 20000 unknowns

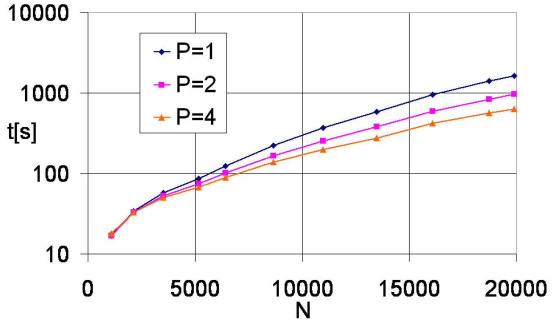

Figure 6: Simulation time for execution on 1, 2 and 4 processor cores on the workstation – in-core solution

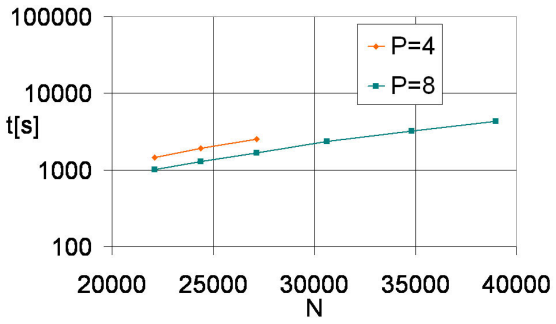

Figure 7: Simulation time for parallel execution using 4, and 8 processors on the cluster – above 20000 unknowns

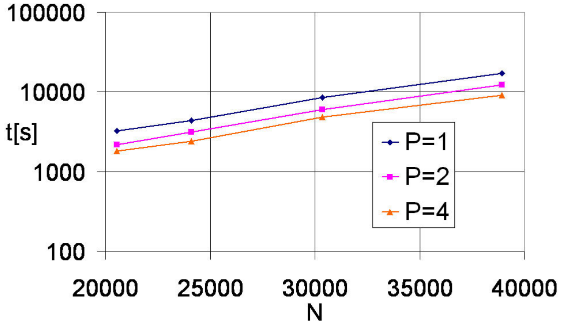

Figure 8: Simulation time for execution on 1, 2 and 4 processor cores on the workstation – out-of-core solution

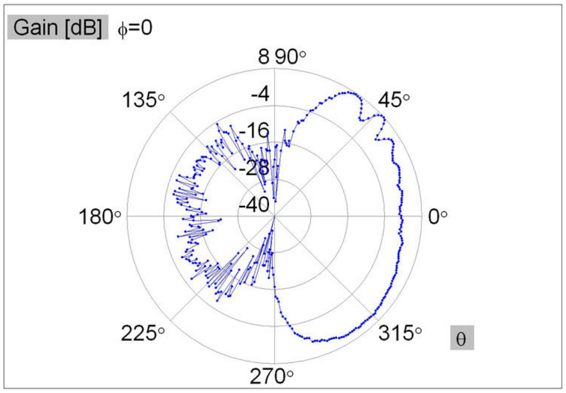

Figure 9: RCS in the plane perpendicular to dipole

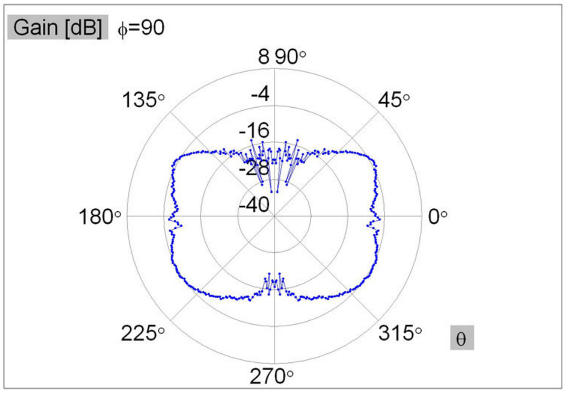

Figure 10: RCS in the plane parallel to dipole

References

[1] B. M. Kolundzija, A. M. Djordjevic, *Electromagnetic Modeling of Composite Metallic and Dielectric Structures*, Artech House, Norwood USA, 2002.

[2] WIPL-D Pro v6.1, *Software and User's Manual*, WIPL-D d.o.o., Belgrade Serbia, 2006.

[3] B. M. Kolundzija, D. S. Sumic, "Adaptive higher order modeling of antennas placed on large platforms," Proc. European Conference on Antennas and Propagation: EuCAP 2006, Nice France, session PA6, paper code 350289, 2006.

[4] P. S. Pacheco, *Parallel Programming with MPI*, Morgan Kaufmann Publishers, San Francisco USA, 1997.

[5] *Intel Math Kernel Library Reference Manual*, Intel Corporation, 2006.