Fast Circuit Simulation Based on Parallel-Distributed LIM using Cloud Computing System

Yuta Inoue, Tadatoshi Sekine, Takahiro Hasegawa, and Hideki Asai

***Abstract*—This paper describes a fast circuit simulation technique using the latency insertion method (LIM) with a parallel and distributed leapfrog algorithm. The numerical simulation results on the PC cluster system that uses the cloud computing system are shown. As a result, it is confirmed that our method is very useful and practical.**

***Index Terms*—Latency insertion method, fast circuit simulation, parallel computing, cloud computing system**

I. INTRODUCTION

In these years, the high-speed and high-density electronic circuit designs have been required for the latest chips, packages and boards. With the progress of integration technology, a variety of signal and power integrity problems have become serious and important. Thus, for the efficient designs, a variety of advanced simulation techniques have been required to clarify the various effects of the high-speed signal behaviors.

LIM has been proactively proposed as one of the fast transient simulation methods applicable to large networks [1-5]. The algorithm of LIM is analogous to the relaxation-based one which does not need matrix operations and it seems that this is suitable for the parallel implementation. We have already given a parallel-distributed leapfrog algorithm [3, 4] based on the LIM by using MPI [6].

This paper shows the novel simulation results performed by the clustered cloud computing system [7, 8] with the sixteen calculation instances. In our approach, the original circuit is partitioned into several computational domains. The updating calculations in each domain are performed concurrently. In this case, the number of the domains is exactly equal to the number of processing elements (PEs). In this research, a plane circuit, which is frequently given as the model of power distribution networks, is analyzed by parallel-distributed LIM.

Manuscript received Oct. 1, 2009; revised Dec. 27, 2009.
Department of Information Science and Technology, Shizuoka University Hamamatsu, Japan
E-mail : inoue@tzasai7.sys.eng.shizuoka.ac.jp

II. LATENCY INSERTION METHOD

LIM is one of the circuit simulation methods based on the leapfrog algorithm for the fast transient analysis. Unlike the conventional SPICE-like simulators which require the time-consuming LU decomposition of large scale coefficient matrices, the LIM algorithm does not need directly the matrix operations. In fact, because of its linearly-increasing characteristic of the calculation amount of the LIM algorithm, LIM-based simulation is much faster than the conventional methods for large-scale networks [1-5].

The LIM algorithm requires the circuit to be analyzed to be composed of the combination of the certain type of the topology, namely the branch and node topologies. The branch topology is shown in Fig. 1(a), and the node

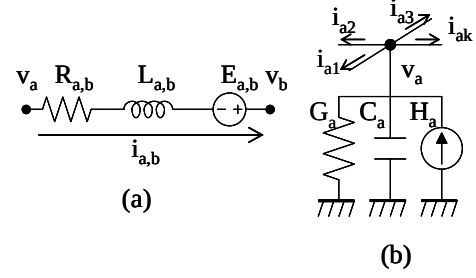

Fig. 1. Required linear circuit topologies for LIM algorithm. (a) Linear branch topology for LIM. (b) Linear node topology for LIM.

topology is shown in Fig. 1(b). The branch must consist of the series connected resistance $R_{a,b}$, inductance $L_{a,b}$ and independent voltage source $E_{a,b}$, and they are connected between arbitrary nodes a and b in the network. Similarly, each node in the circuit must consist of the parallel connected conductance G_a, capacitance C_a and independent current source H_a and they are respectively connected between an arbitrary node a and the reference node, i.e. ground. That is to say, a topology of the network has to be satisfied with the following requirements: Each branch in the network must contain an inductance and each node in the network must connect a capacitance to ground. Otherwise, a relatively small inductor or shunt capacitor is inserted into the corresponding branch or node to generate latency, respectively. Thus, in order to generate the updating formulas of LIM for a linear network, applying the Kirchhoff's voltage law (KVL) to the branch and the Kirchhoff's current law (KCL) to the node with the finite difference method leads to

$$v_a^{n+\frac{1}{2}} - v_b^{n+\frac{1}{2}} = R_{a,b} i_{a,b}^n + L_{a,b}\left(\frac{i_{a,b}^{n+1} - i_{a,b}^n}{\Delta t}\right) - E_{a,b}^n \qquad (1)$$

$$-\sum_{k=1}^{Ma} i_{a,k}^n = G_a v_a^{n+\frac{1}{2}} + C_a\left(\frac{v_a^{n+\frac{1}{2}} - v_a^{n-\frac{1}{2}}}{\Delta t}\right) - H_a^n \qquad (2)$$

where n is the time step, Δt is the time step size and M_a is the number of the branches connected to the node a. Note that the time steps of the branch current and the node voltage are collocated in half time step, which is similar to the algorithm in the FDTD (Finite Difference Time Domain) method for the electromagnetic simulation. Then, solving (1) for the branch current $i_{a,b}^{n+1}$ and (2) for the node voltage $v_a^{n+1/2}$ leads to the following updating formulas.

$$i_{a,b}^{n+1} = \frac{-\Delta t R_{a,b} + L_{a,b}}{L_{a,b}} i_{a,b}^n + \frac{\Delta t}{L_{a,b}}\left(v_a^{n+\frac{1}{2}} - v_b^{n+\frac{1}{2}} + E_{a,b}^{n+\frac{1}{2}}\right) \qquad (3)$$

$$v_a^{n+\frac{1}{2}} = \frac{C_a}{\Delta t G_a + C_a} v_a^{n-\frac{1}{2}} + \frac{\Delta t}{\Delta t G_a + C_a}\left(-\sum_{k=1}^{Ma} i_{a,k}^n + H_a^n\right) \qquad (4)$$

Since all terms in the right hand sides of the updating formulas (3) and (4) can be given at the (n+1)-th or the (n+1/2)-th time step, each variable is updated only by substituting the values at the passed time steps. Therefore, they are updated alternately and explicitly as the time progress.

III. PARALLEL-DISTRIBUTED LIM

As described above, each current and voltage variable is updated individually in the LIM algorithm, and thereby the current and voltage updating processes can be easily performed in parallel. In other words, in the case that branch currents are updated at an arbitrary time point, each branch current is updated itself without any other variables at the same time step and can refer the variables at the past time points explicitly. The same procedure is also done in the case of updating the voltage. Thus, the calculations for updating are decoupled each other, and therefore, they can be performed in parallel completely.

Here, the procedure of the parallel-distributed LIM is described for the plane circuit which consists of passive, linear and time-invariant components as shown in Fig. 2. The power/ground plane in a printed circuit board is modeled as the equivalent circuit and its topology is suitable for the LIM algorithm [3, 4]. In Fig. 2, it is assumed that the number of processing elements (PEs) is two and the plane circuit is divided into two domains along the interface node c, h, m and r. Then, one PE, named PE1, holds the values of the branch currents $i_{b,c}$, $i_{g,h}$, $i_{l,m}$, $i_{q,r}$ and the other currents and voltages in the left half plane. And another one, named PE2, holds $i_{c,d}$, $i_{h,i}$, $i_{m,n}$, $i_{r,s}$ and the other variables in the right half plane. Note that the values of the interface node voltages v_c, v_h,

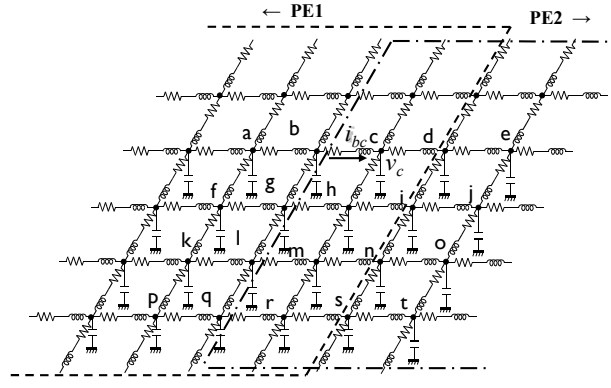

Fig. 2. Partitioning of a plane circuit.

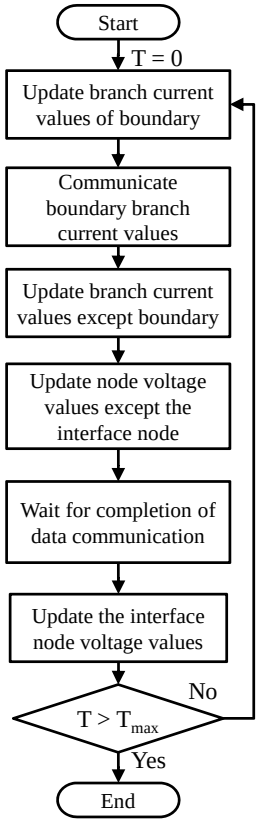

Fig. 3. Flowchart of parallel-distributed LIM.

v_m and v_r are held by both PE1 and PE2, and the updating calculations for these voltages are processed by both PEs. In the parallel-distributed LIM, each PE updates only the variables which each PE holds.

Fig. 3 shows the algorithm of parallel-distributed LIM. In the original LIM, the branch currents and the node voltages are alternately updated in each time step. On the other hand, in the parallel-distributed LIM, first the branch current values of boundary are updated. Second, the boundary branch current values are communicated with neighboring PEs. The branch current values and the node voltage values except the boundary part are calculated in each domain during data communication. Each PE has to wait for completion of data communication. Finally, the interface node voltages are updated.

IV. NUMERICAL RESULTS

In order to verify the validity of the original LIM and the parallel-distributed LIM, some example circuits were simulated. Fig. 4 shows an example plane equivalent circuit. In all of the simulations, the waveform with

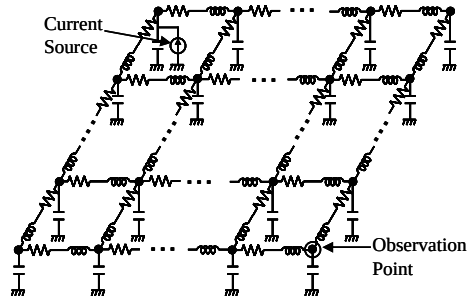

Fig. 4. An example plane equivalent circuit.

delay of 0.2 nsec, rising time of 0.1 nsec, pulse width of 1.0 nsec, and magnitude of 0.05 A was used as the input current.

First, the simulation results (transient responses) of the plane equivalent circuit composed of 400 unit cells are illustrated in Fig. 5 and Table 1 shows the execution times by HSPICE and the LIM. The simulation has been done on Sparcv9 1GHz. The waveform results, Fig. 5, show the good agreement between the LIM and HSPICE. From Table 1, it can be seen that the LIM is about 160 times faster than HSPICE in the case of 10,000 unit cells.

Next, in order to demonstrate the performance of the parallel-distributed LIM, we simulated transient responses of some plane circuits, which are modeled by 1,000,000,

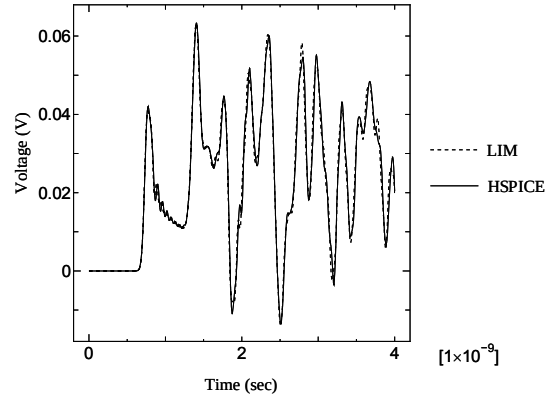

Fig. 5. Transient simulation result of the network composed of 400 Unit Cells.

Table 1. Comparing execution times by HSPICE and LIM

Number of Cells	Execution time (seconds)	
	HSPICE	LIM
400	4.68	0.39
10,000	935.88	5.78

4,000,000 and 9,000,000 unit cells.

We confirmed the performance of a clustering computer network system having two instances. A clustering computer network system is constructed by the cloud computing system provided by Amazon EC2 service [7]. The performance of two instances which correspond to 2 PCs is compared to the single PC case. Each calculation instance has two CPUs, each of which is composed of quad core. In addition, each process is performed by each core. Thus, the 16 cores are available as the maximum performance. Fig. 6 shows the relationship between the speed-up ratio and the number of processes for three kinds of network models under the condition that the number of the time steps was 1,000. In the case of the cloud computing system, the speed-up ratio is saturated around the 6 processes. We also performed a SGI Altix4700 under the same condition. This high performance computer system is composed of sixteen CPUs, each of which is Itanium 2 1.6 GHz. In addition, each process is performed by each CPU. Table 2 shows the computer environments of the SGI Altix4700 and the cloud computing system. In the case of SGI Altix4700, the speed-up ratio for all models is monotonically increasing.

We also tested the performance of cloud computing system by using sixteen instances, namely 32 CPUs.

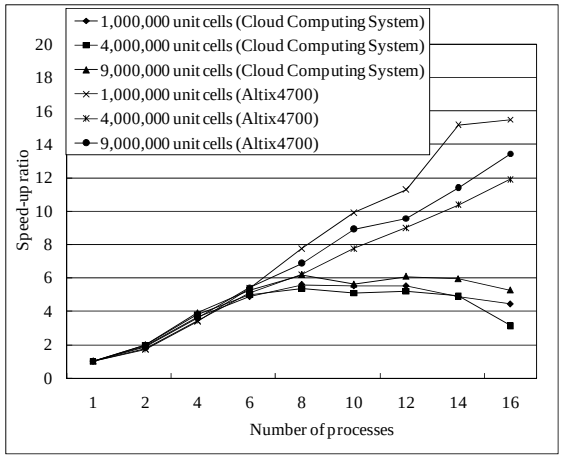

Fig. 6. Speed up ratio comparison of cloud computing system with Altix4700.

Table 2. Computer environments

	SGI Altix4700	Cloud Computing System
CPUs	**16**	**4**
Cores	**-**	**16**

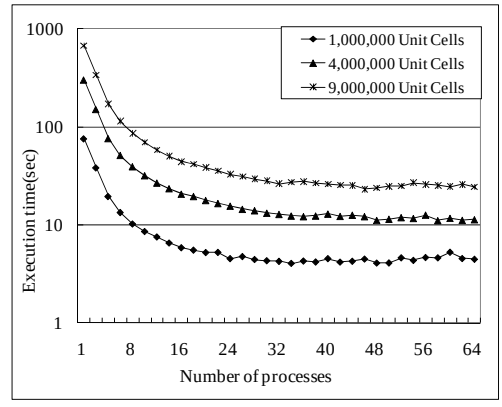

Fig. 7. Execution time vs # of process.

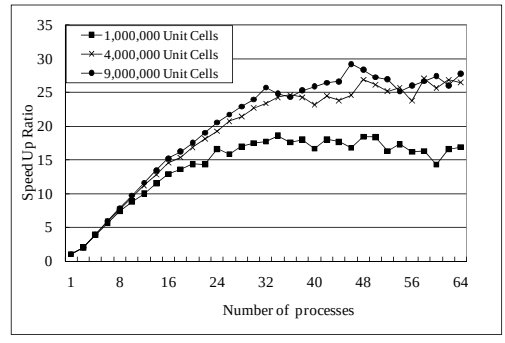

Fig. 8. Speed-up ratio.

Thus, 128 cores are available as the maximum performance. Fig. 7 shows the relationship between the execution time and the number of processes. The execution time monotonically decreases until around 32 processes. Fig. 8 shows the relationship between the speed-up ratio and the number of processes. The speed-up ratio monotonically increases until around 32 processes. These figures clearly show that the execution time of 32 processes is around 25 times faster than the execution time of 1 process. Although the execution time decreased until around 32 processes, the execution time does not decrease in the range of over 32 processes. That is to say, the speed-up ratio is saturated by the bottle neck of data transfer between CPUs and main memory. Therefore, the performance cannot be improved by increasing the number of cores. As a result, it is considered that the execution time monotonically decreases by increasing the number of CPUs.

V. CONCLUSIONS

In this paper, we described the parallel and distributed LIM-based fast simulation method for large-scale linear

networks. This method is very useful for the power distribution network analysis. First, LIM was briefly reviewed and it was referred that this method was suitable for the parallel and distributed computing. Next, the parallel-distributed LIM was constructed on the cloud computing system. Finally, it was confirmed that the parallel-distributed LIM on the cloud system was very efficient and the performance was almost ideally high according to the number of CPUs without losing accuracy.

REFERENCES

[1] J. E. Schutt-Ainé, "Latency insertion method (LIM) for the fast transient simulation of large networks," *IEEE Trans. Circuit Syst. I*, Vol.49, No.1, Jan., 2001, pp.81-89.

[2] H. Kubota, Y. Tanji, T. Watanabe and H. Asai, "Generalized Method of the Time-Domain Circuit Simulation based on LIM with MNA Formulation," *Proc. CICC 2005*, Sep., 2005, pp.289-292.

[3] T. Watanabe, Y. Tanji, H. Kubota and H. Asai, "Parallel-Distributed Time-Domain Circuit Simulation of Power Distribution Networks with Frequency-Dependent Parameters," *Proc. ASP-DAC 2006*, Jan., 2006, pp.832-837.

[4] T. Watanabe, Y. Tanji, H. Kubota and H. Asai, "Fast Transient Simulation of Power Distribution Networks Containing Dispersion Based on Parallel -Distributed Leapfrog Algorithm," *IEICE Trans. Fundamentals*, Vol.E90-A, No.2, Feb., 2007, pp.388-397.

[5] H. Asai and N. Tsuboi, "Multi-Rate Latency Insertion Method with RLGC-MNA Formulation for Fast Transient Simulation of Large-Scale Interconnect and Plane Networks," *Proc. ECTC2007*, June., 2007, pp.1667-1672.

[6] http://www.mpi-forum.org/

[7] http://aws.amazon.com/ec2/

[8] Y. Inoue, T. Sekine, T. Hasegawa and H. Asai, "Fast Circuit Simulation Based on Parallel-Distributed LIM using Cloud Computing System," *Proc. ITC-CSCC2009*, Jul., 2009, pp.845-846.

Yuta Inoue received the B.E. and M.E. degrees in system engineering from Shizuoka University, Hamamatsu, Japan, in 2005 and 2007, respectively. Currently, he is working toward the Ph.D. degree in information science and technology at Shizuoka University. His research interests are in the fast circuit simulation of the large interconnects and the power distribution networks (PDNs) of the chips and packages.

Tadatoshi Sekine received the B.E. and M.E. degrees in system engineering from Shizuoka University, Hamamatsu, Japan, in 2007 and 2009, respectively. Currently, he is working toward the Ph.D. degree in information science and technology at Shizuoka University. His research interests are in the fast circuit simulation of the large interconnects and the power distribution networks (PDNs) of the chips and packages.

Takahiro Hasegawa received the Ph.D. degrees in information engineering from Kyushu Institute of Technology, Fukuoka, Japan, in 1997. Since 1997, he has been with Shizuoka University, Hamamatsu, Japan, where he is currently an Associate Professor involved with information infrastructure for the campus network and its security system including high performance computers and a cloud computing.

Hideki Asai received the B.E., M.E., and Ph.D. degrees in electrical engineering from Keio University, Yokohama, Japan, in 1980, 1982, and 1985, respectively. In 1985, he was with the Department of Electrical and Electronics Engineering, Sophia University, Tokyo, Japan. He was a Visiting Professor at Carleton University, Ottawa, ON, Canada, and Santa Clara University, Santa Clara, CA (1999-2000). Since 1986, he has been with Shizuoka University, Hamamatsu, Japan, where he is currently a Professor involved with VLSI-CAD and electrical design automation (EDA), analog circuit design, and neural networks. He is an author of the books, “Exercise Notes of Digital Circuits, CORONA PUBLISHING. CO., LTD., 2001” and “Electronic Circuit Simulation Techniques, SCI TECHS PRESS, 2003” Dr. Asai is a member of the IEEE Nonlinear Circuits and Systems Technical Committee. He was secretary for the IEEE Circuits and Systems Society Tokyo Chapter (1994-1995), and secretary of the Technical Group on Nonlinear Problems of the Institute of Electronics, Information and Communication Engineers (IEICE) (1997-1999). He was a chairman of the Technical Group on Nonlinear Problems of the IEICE (2007-2008) and a chairman of the Technical Group on System Packaging CAE of JIEP (2007-2009), and is now an executive board member of JIEP. He was the recipient of the Research Encouragement Awards on the occasion of the Takayanagi anniversary, the 50th anniversary of the founding of the IEICE Tokai branch, on the occasion of the Saitoh anniversary, and Prize for Science and Technology (Research Category) awarded by Minister of Education, Culture, Sports, Science and Technology in 1988, 1989, 1993, and 2009, respectively.