Block Latency Insertion Method (Block-LIM) for Fast Transient Simulation of Tightly Coupled Transmission Lines

Tadatoshi Sekine, *Student Member, IEEE*, and Hideki Asai, *Member, IEEE*

Abstract—This paper describes a block latency insertion method (LIM) for the fast transient simulation of the large networks with many coupling elements. First, the basic formulation of LIM is reviewed. Next, the block-LIM formulation for the network with many coupling elements, such as the mutual inductance, the mutual capacitance, and controlled sources is described. Then, the block-LIM algorithm is applied to the tightly coupled transmission lines which are connected to each other by a number of mutual inductors and capacitors. Finally, some numerical results are shown, and it is confirmed that the proposed technique is useful and efficient for the simulation of the tightly coupled transmission lines.

Index Terms—latency insertion method, block-latency insertion method, coupling element, mutual inductance, mutual capacitance, controlled source, tightly coupled transmission lines.

I. INTRODUCTION

AS the high-speed and high-density electronic circuit design of the latest large-scale integration circuit progresses, various effects which depend on the high-frequency characteristics of the signals are induced on the chips and packages. Because these effects cause unexpected behavior of chips and packages on a printed circuit board, it becomes important to verify electronic circuit behavior including these effects. From a circuit simulation point of view, a net-list includes an enormous number of parasitic elements in order to verify the exact behavior of chips and packages; it is mainly provided by an extractor which extracts circuit element parameters from the object to be analyzed and outputs them as the net-lists. This fact causes a large amount of simulation time for a SPICE-like simulator based on the matrix solver. Therefore, fast simulation methods different from SPICE-like ones are strongly needed. Our recent research indicates that the latency insertion method (LIM) in [1] could be one of the important methods for the fast simulation of large networks [2]–[4]. Ultimately, a method which circumvents the matrix operation such as LIM and the relaxation-based method in [5] are imperative for the simulation of recent extremely high-speed and high-density circuits.

There exist a number of coupling elements in the equivalent circuit of chips and packages. This is because the components, including the signal conductor lines, in these high-density circuits are very close to each other, and therefore, they are connected each other electromagnetically, even though they are not connected physically or electrically. Most of the well-established commercially-based extractors derive an equivalent circuit which consists not only of resistance, inductance, and

grounded capacitance, but also of mutual inductance and mutual capacitance.

There are some techniques to expand the LIM algorithm to a circuit which includes those coupling elements [1], [6]. In [1], mutual inductance and mutual capacitance are incorporated in the LIM formulation using equivalent circuit topologies for those elements. A coupled inductor system can be transformed to a circuit composed of two equivalent current sources using a simple 2×2 matrix inversion. However, this technique deals with only one mutual inductance between two different self inductances. On the other hand, mutual capacitance is transformed to the equivalent branch topology using the companion model of a capacitor associated with an integration algorithm [7], which is composed of the series connected equivalent resistance and equivalent current source. In this case, a small inductance is inserted as a latency to the equivalent branch topology in order to derive the updating formula. However, it is evidenced in [6] that the branch which consists of the companion model and the inserted latency is not compatible with LIM which is based on the interlaced finite-difference time-domain (FDTD) algorithm. Although another modeling technique to apply the LIM algorithm to mutual capacitance is proposed in [6], it also deals with only one mutual capacitance between two nodes. In this paper, we describe application techniques of the LIM to circuit simulations with more than two coupling elements, especially to tightly coupled transmission lines. As a result, our locally implicit scheme can deal with the coupling elements while retaining the advantage of the LIM algorithm.

The remainder of the paper is organized as follows. In Section II, the formulation of the basic LIM is introduced. In Section III, we propose the block-LIM formulation which is the matrix version of the LIM formulation for coupling elements such as mutual inductance, mutual capacitance, and controlled sources. Additionally, we also describe the application technique of the block-LIM to tightly coupled transmission lines in Section IV. Section V shows some numerical results, followed by conclusions in Section VI.

II. BASIC LIM FORMULATION

In this section, we review the fundamental algorithm of LIM which is the fast transient analysis method of large networks [1]. The method is one of the finite difference method using the leapfrog scheme and its algorithm is analogous to the Yee's algorithm developed in the FDTD method. In fact, while

Fig. 1. Required circuit topologies for LIM algorithm. (a) Branch topology. (b) Node topology.

the FDTD method is well known as the form of the finite difference equations of the Maxwell's equations in the field of electromagnetics, LIM can be regarded as the application of the FDTD method to circuit simulation.

In the LIM algorithm, the circuit to be analyzed is required to be composed of a certain type of topology, namely, the branch and the node topology. The branch topology consists of the series connected resistance, inductance, and voltage source from a node a to a node b , as shown in Fig. 1(a). And the node topology consists of the parallel connected conductance, capacitance, and current source at a node a to the ground as shown in Fig. 1(b). In addition, a topology of the network has to satisfy with the following requirements: Each branch in the network must contain an inductance and each node in the network must provide a capacitive path to the ground. Otherwise, a relatively small inductor or shunt capacitor is inserted into the corresponding branch or node to generate the latency, respectively. In order to generate the updating formulas of LIM for a linear network, applying the Kirchhoff's voltage law (KVL) to the branch shown in Fig. 1(a) and applying the Kirchhoff's current law (KCL) to the node shown in Fig. 1(b) with the finite difference method lead to

$$v_a^{n+\frac{1}{2}} - v_b^{n+\frac{1}{2}} = L_{ab} \frac{i_{ab}^{n+1} - i_{ab}^n}{\Delta t} + R_{ab} i_{ab}^n - E_{ab}^{n+\frac{1}{2}}, \quad (1)$$

$$-\sum_{k=1}^{M_a} i_{ak}^n = C_a \frac{v_a^{n+\frac{1}{2}} - v_a^{n-\frac{1}{2}}}{\Delta t} + G_a v_a^{n+\frac{1}{2}} - H_a^n, \quad (2)$$

where M_a is the number of branches connected to the node a , n is the time step, and Δt is the time step size. In this formulation, the time points of the branch currents and the node voltages are collocated in half time steps, similar to the FDTD method. Then, solving (1) for the branch current i_{ab}^{n+1} and solving (2) for the node voltage $v_a^{n+1/2}$ leads to the following updating formulas:

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

$$v_a^{n+\frac{1}{2}} = \frac{C_a}{C_a + \Delta t G_a} v_a^{n-\frac{1}{2}} + \frac{\Delta t}{C_a + \Delta t G_a} \left(-\sum_{k=1}^{M_a} i_{ak}^n + H_a^n \right). \quad (4)$$

Then, all of the branch currents and the node voltages in

the network are calculated using (3) and (4) alternately as time progresses. By this formulation, if a fixed time step size is used, the calculation amount at each time step results in $\mathcal{O}(N_b + N_n)$, where N_b and N_n are the numbers of the branches and the nodes in the network, respectively. Therefore, because its linearly-increasing CPU time of LIM, it can reduce calculation costs of the transient simulation compared to methods based on a SPICE-like algorithm which need a matrix solver; the calculation amount of the algorithm based on a direct method is typically $\mathcal{O}(N^2)$ at each time step, where N is the number of the variables, with the additional calculation amount $\mathcal{O}(N^3/3)$ for the factorization step before the iteration of the transient simulation. Basically, the LIM enables one to do the fast simulation of the large linear networks tens or hundreds of times faster than with conventional simulators. The LIM algorithm is quite simple and has the capability to be extended to more general circuit simulations.

III. BLOCK-LIM FORMULATION

Needless to say, we can derive only one solution from each of the updating formulas (3) and (4) at each time step. This is because the difference equations (1) and (2) include respectively only one variable, and therefore, these equations are not solved implicitly by the matrix solver but solved explicitly and result in the explicit forms of the updating formulas. This scheme is analogous to the relaxation-based method for the circuit simulations, which separates the simultaneous equation into several individual equations and solves them individually [5]. In fact, it is noted in [8] that the basic LIM formulation introduced in II is a scalar version of the FDTD-like formulation. However, because the special elements such as mutual inductance, mutual capacitance, and controlled sources induce more than one current or voltage variable from other branches or nodes to the KVL (1) and the KCL (2) equations, the basic LIM algorithm can not be applied directly to those elements. Although there are application techniques of LIM to circuits including mutual inductance and mutual capacitance, they deal with only one mutual inductance between two self inductances and one mutual capacitance connected between two nodes as described above. We describe the way to formulate for more than one mutual inductance and mutual capacitance using the LIM algorithm below.

A. Formulation for Branch Topologies Coupled by Mutual Inductance

One mutual inductance between two self inductances is incorporated in the LIM formulation using the equivalent transformation [1]. In order to expand the LIM algorithm to a circuit with more than one mutual inductance, we consider n_b branch topologies in a circuit each of which consists of the series connected resistance R_m , self inductance L_m , and voltage source E_m , where $m = 1, 2, \dots, n_b$. The currents flowing through the branches are defined as i_m and the voltages at the both sides of the branches are v_{am} and v_{bm} . In addition, each of the self inductances L_p in these branches is coupled with self inductances L_q in any other branches by mutual inductances $L_{p,q}$, where $p, q = 1, 2, \dots, n_b$ and $p < q$,

Fig. 2. Required circuit topologies for block-LIM algorithm. (a) Branch block topology, in which n_b branch topologies are coupled by mutual inductances. (b) Node block topology, in which n_n node topologies are coupled by mutual capacitances.

as shown in Fig. 2(a). In this case, the number of mutual inductances is equal to $(n_b^2 - n_b)/2$. Then, applying KVL to each branch leads to the simultaneous equations in a vector-matrix form

$$\mathbf{v}_{ab} = \mathbf{L}_{ab} \cdot \frac{d}{dt} \mathbf{i}_{ab} + \mathbf{R}_{ab} \cdot \mathbf{i}_{ab} - \mathbf{e}_{ab}, \quad (5)$$

where

$$\mathbf{L}_{ab} = \begin{bmatrix} L_1 & L_{1,2} & \dots & L_{1,n_b} \\ L_{1,2} & L_2 & \dots & L_{2,n_b} \\ \vdots & \vdots & \ddots & \vdots \\ L_{1,n_b} & L_{2,n_b} & \dots & L_{n_b} \end{bmatrix},$$

$$\mathbf{R}_{ab} = \begin{bmatrix} R_1 & 0 & \dots & 0 \\ 0 & R_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & R_{n_b} \end{bmatrix},$$

$$\mathbf{i}_{ab} = \begin{bmatrix} i_1 \\ i_2 \\ \vdots \\ i_{n_b} \end{bmatrix}, \quad \mathbf{v}_{ab} = \begin{bmatrix} v_{a1} - v_{b1} \\ v_{a2} - v_{b2} \\ \vdots \\ v_{an_b} - v_{bn_b} \end{bmatrix}$$

and

$$\mathbf{e}_{ab} = \begin{bmatrix} E_1 \\ E_2 \\ \vdots \\ E_{n_b} \end{bmatrix}$$

are the block matrices and vectors dependent on a branch block topology shown in Fig. 2(a). Applying the finite difference method to (5) in a manner similar to the basic LIM formulation leads to the block KVL equation in the discrete time domain

$$\mathbf{v}_{ab}^{n+\frac{1}{2}} = \frac{1}{\Delta t} \mathbf{L}_{ab} \cdot (\mathbf{i}_{ab}^{n+1} - \mathbf{i}_{ab}^n) + \mathbf{R}_{ab} \cdot \mathbf{i}_{ab}^n - \mathbf{e}_{ab}^{n+\frac{1}{2}}. \quad (6)$$

Then, combining like terms and transposing the term related to the unknown variable vector $\mathbf{i}_{ab}^{n+1}$ to the left hand side to

simplify the equation lead to

$$\frac{1}{\Delta t} \mathbf{L}_{ab} \cdot \mathbf{i}_{ab}^{n+1} = \left(\frac{1}{\Delta t} \mathbf{L}_{ab} - \mathbf{R}_{ab} \right) \cdot \mathbf{i}_{ab}^n + \mathbf{v}_{ab}^{n+\frac{1}{2}} + \mathbf{e}_{ab}^{n+\frac{1}{2}}. \quad (7)$$

We can derive the values of all currents flowing through each branch in a branch block at the $(n+1)$ -th time point by solving (7). In fact, during the transient analysis of a whole circuit, the current variables associated with the branch block are updated using (7) while the other currents are updated by using (3). Thus, (7) can be regarded as an updating formula for the currents in a branch block. Note that if there is no mutual inductance between the branches, basically all $L_{p,q}$ are equal to zero and the matrix $\mathbf{L}_{ab}$ becomes a diagonal matrix, each of the equations in (6) can be regarded as the KVL equation (1) appearing in the basic LIM formulation in Section II.

Equation (7) is a linear algebraic equation and its coefficient matrix $(1/\Delta t)\mathbf{L}_{ab}$ is typically completely dense. If the coefficient matrix is a diagonally dominant matrix, we may apply several iterative methods such as the Gauss-Seidel method, the successive over-relaxation (SOR) method, and the conjugate gradient (CG) method to solve the equation. This is valid in the case that the coupling strength of two metal conductors, for example, attenuates naturally as the distance between them increases, and the values of the off-diagonal elements of the coefficient matrix decrease with distance from diagonal parts. However, some extraction techniques or extractors provide coupling elements for which the extracted values are irregular compared to the above case [9]. The coefficient matrix composed of them no longer has diagonally dominance. Under such circumstances, we should adopt a direct method such as the LU decomposition method to solve (7). Thus, our technique suffers from a factorization cost of which the calculation amount is $\mathcal{O}(n_b^3/3)$ for a branch block. In addition, the computational complexity of the forward and backward substitutions at each time step is $\mathcal{O}(n_b^2)$. However, the order of each updating formula of a branch block is smaller than that of the whole circuit, and thus, the computation time is expected to be effectively reduced.

B. Formulation for Node Topologies Coupled by Mutual Capacitance

A mutual capacitance is formulated in a similar manner as for the mutual inductance case. In order to deal with the more than one mutual capacitance, we consider n_n node topologies in a circuit each of which consists of parallel connected conductance G_m , capacitance C_m , and current source H_m to the ground, where $m = 1, 2, \dots, n_n$. The voltages at the nodes are defined as v_m and the currents flowing out from, or into, the node m are i_{mk} , where $k = 1, 2, \dots, M_m$ and M_m is the number of those currents. In addition, each of the nodes is connected with any other node by mutual capacitances $C_{p,q}$, where $p, q = 1, 2, \dots, n_n$ and $p < q$, as shown in Fig. 2(b). In this case, the number of mutual capacitances can be also counted by $(n_n^2 - n_n)/2$. Then, applying KCL to each node leads to the simultaneous equations in a vector-matrix form

$$-\mathbf{i}_a = \mathbf{C}_a \cdot \frac{d}{dt} \mathbf{v}_a + \mathbf{G}_a \cdot \mathbf{v}_a - \mathbf{h}_a, \quad (8)$$

where

$$\mathbf{C}_a = \begin{bmatrix} C_{1,1} & -C_{1,2} & \dots & -C_{1,n_n} \\ -C_{1,2} & C_{2,2} & \dots & -C_{2,n_n} \\ \vdots & \vdots & \ddots & \vdots \\ -C_{1,n_n} & -C_{2,n_n} & \dots & C_{n_n,n_n} \end{bmatrix},$$

$$\mathbf{G}_a = \begin{bmatrix} G_1 & 0 & \dots & 0 \\ 0 & G_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & G_{n_n} \end{bmatrix},$$

$$\mathbf{v}_a = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_{n_n} \end{bmatrix}, \quad \mathbf{i}_a = \begin{bmatrix} \tilde{i}_1 \\ \tilde{i}_2 \\ \vdots \\ \tilde{i}_{n_n} \end{bmatrix}$$

and

$$\mathbf{h}_a = \begin{bmatrix} H_1 \\ H_2 \\ \vdots \\ H_{n_n} \end{bmatrix}$$

are the block matrices and vectors dependent on a node block topology shown in Fig. 2(b). In addition, $C_{m,m}$ and $\tilde{i}_m$ are defined as

$$C_{m,m} = \sum_{k=1}^{m-1} C_{k,m} + C_m + \sum_{k=m+1}^{n_n} C_{m,k},$$

$$\tilde{i}_m = \sum_{k=1}^{M_m} i_{mk},$$

$$(m = 1, 2, \dots, n_n).$$

Applying the finite difference method to (8) in a similar manner to the basic LIM formulation leads to the block KCL equation in the discrete time domain

$$-\mathbf{i}_a^n = \mathbf{G}_a \cdot \mathbf{v}_a^{n+\frac{1}{2}} + \frac{1}{\Delta t} \mathbf{C}_a \cdot (\mathbf{v}_a^{n+\frac{1}{2}} - \mathbf{v}_a^{n-\frac{1}{2}}) - \mathbf{h}_a^n. \quad (9)$$

Then, combining like terms and transposing the term related to the unknown variable vector $\mathbf{v}_a^{n+1/2}$ to the left hand side to simplify the equation lead to

$$\left(\frac{1}{\Delta t} \mathbf{C}_a + \mathbf{G}_a \right) \cdot \mathbf{v}_a^{n+\frac{1}{2}} = \frac{1}{\Delta t} \mathbf{C}_a \cdot \mathbf{v}_a^{n-\frac{1}{2}} - \mathbf{i}_a^n + \mathbf{h}_a^n. \quad (10)$$

The values of the all voltages at the $(n+1/2)$ -th time point at each node in a node block are derived by solving (10). Thus, (10) can be regarded as an updating formula for the voltages in a node block. Note that if there is no mutual capacitance between the branches, basically all $C_{p,q}$ are equal to zero, each of the equations in (9) can be regarded as the KCL equation (2) appearing in Section II. For a similar reason in the branch block case, we adopt the direct method to solve (10) at each time step.

C. Formulation for Controlled Sources

Some kinds of extractors provide controlled sources to represent electromagnetic effects caused by the fluctuation of voltages and currents in the network. Those sources induce the effects of the voltages and the currents in other parts directly as the mutual elements described above. Fig. 3(a) shows the branch block in which a current controlled current source (CCCS) is connected to each resistor R_m in parallel. In this case, it is assumed that since the CCCS is controlled by the currents flowing through all other branch topologies in the branch block, there are actually $n_b - 1$ CCCSs connected to the resistor in parallel. In Fig. 3(a), $n_b - 1$ CCCSs connected to each branch topology are simplified and illustrated as one CCCS as shown in Fig. 3(b). Basically, the value of the CCCS, I_{Fm} in Fig. 3(b), is defined as

$$I_{Fm} = \sum_{k=1, k \neq m}^{n_b} (F_{m,k} \cdot i_k) \quad (m = 1, 2, \dots, n_b), \quad (11)$$

where $F_{m,k}$ is the value of the current gain and i_k is the current through the k -th resistor in the branch block. In order to formulate the branch block shown in Fig. 3(a), the parallel connected resistor and CCCS in the branch topology is transformed to the Thevenin's equivalent circuit as shown in Fig. 3(c). In this case, the value of the equivalent voltage source V_{Fm} can be obtained by $V_{Fm} = R_m \cdot I_{Fm}$. The topology of the transformed branch is the one completely required for the block-LIM formulation. Since the value of the CCCS is defined as (11) and multiplied by R_m to transform to the voltage source, the nonzero elements associated with the CCCS are stamped in the off-diagonal parts of the resistance matrix $\mathbf{R}_{ab}$ in (5). Actually, the updating formula of the currents in the branch block with CCCSs is derived by replacing $\mathbf{R}_{ab}$ in (5) with the resistance matrix

$$\mathbf{R}_{Fab} = \begin{bmatrix} R_1 & -R_1 F_{1,2} & \dots & -R_1 F_{1,n_b} \\ -R_2 F_{2,1} & R_2 & \dots & -R_2 F_{2,n_b} \\ \vdots & \vdots & \ddots & \vdots \\ -R_{n_b} F_{n_b,1} & -R_{n_b} F_{n_b,2} & \dots & R_{n_b} \end{bmatrix},$$

and the same formulation techniques in Section III-A. Note that the signs of the off-diagonal elements are all negative

Fig. 3. Branch block with controlled sources. (a) Branch block with CCCSs. Each CCCS is connected to a resistor in parallel. (b) Detailed description of CCCS in (a). (c) Thevenin's equivalent circuit of branch topology with CCCSs.

and they correspond to the direction of the equivalent voltage source V_{Fm} .

D. Block Matrix Formulation Toward RLCG-MNA Method

To clarify the formulation of the basic LIM and the block-LIM, the circuit which consists of linear RLC elements, coupling elements and current and voltage sources are formulated by using the RLCG-MNA method [10], [11]. The circuit equation derived by the RLCG-MNA method is written as

$$\begin{bmatrix} \mathbf{C} & \mathbf{0} \\ \mathbf{0} & \mathbf{L} \end{bmatrix} \cdot \frac{d}{dt} \begin{bmatrix} \mathbf{v} \\ \mathbf{i} \end{bmatrix} + \begin{bmatrix} \mathbf{G} & \mathbf{A} \\ -\mathbf{A}^T & \mathbf{R} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{v} \\ \mathbf{i} \end{bmatrix} = \begin{bmatrix} \mathbf{h} \\ \mathbf{e} \end{bmatrix}, \quad (12)$$

where $\mathbf{v}$ and $\mathbf{i}$ are the vectors of the voltage and the current variables in the whole circuit, $\mathbf{h}$ and $\mathbf{e}$ are the current source and the voltage source vectors, and $\mathbf{A}$ is the incidence matrix. The matrices $\mathbf{C}$, $\mathbf{G}$, $\mathbf{L}$, and $\mathbf{R}$ are the capacitance, the conductance, the inductance, and the resistance matrices associated with $\mathbf{v}$ and $\mathbf{i}$, respectively. The RLCG-MNA method is based on the conventional modified nodal analysis (MNA) method but different in the definitions of the voltage and the current variables in the network. In other words, it is assumed that the currents flowing through series connected elements, such as R_{ab} , L_{ab} , and E_{ab} in the branch topology in Fig. 1(a), are equal to each other, and the node voltage variables of the intermediate nodes between those elements are removed. These modifications appear as the stamps of $\mathbf{R}$, $\mathbf{L}$, and $\mathbf{e}$ in (12), which are all stamped in the same rows of the current variable vector $\mathbf{i}$. Performing the block matrix operations for (12) leads to the vector-matrix version of the KVL and KCL equations of the whole circuit and is written as:

$$\mathbf{A}^T \cdot \mathbf{v} = \mathbf{L} \cdot \frac{d}{dt} \mathbf{i} + \mathbf{R} \cdot \mathbf{i} - \mathbf{e}, \quad (13)$$

$$-\mathbf{A} \cdot \mathbf{i} = \mathbf{C} \cdot \frac{d}{dt} \mathbf{v} + \mathbf{G} \cdot \mathbf{v} - \mathbf{h}, \quad (14)$$

where the matrices and the vectors are same as those in (12). The matrices in (13) and (14) have dense parts at intervals around the diagonal elements. This is because the matrices $\mathbf{L}$, $\mathbf{R}$, $\mathbf{C}$, and $\mathbf{G}$ are composed of not only nonzero entries of the diagonal elements, which are the coefficients in the KVL and KCL equations in the basic LIM formulation, but also those of several $\mathbf{L}_{ab}$, $\mathbf{R}_{ab}$, $\mathbf{C}_a$, and $\mathbf{G}_a$ in (5) and (8) in the block-LIM formulation. In the block-LIM, the variables corresponding to each dense part are solved simultaneously, namely, the implicit

scheme is generated. In this case, although the dense parts are solved implicitly by using a direct method, those parts can be separated from each other and also divided from the non-dense parts where the coupling element does not exist. Actually, this locally implicit scheme is efficient because the matrix solver is applied only to each dense part separately, unlike the conventional direct methods which solve the whole circuit in a lump. Additionally, such a locally implicit scheme can still use different time step sizes for the different blocks by using the methods in [2] and [12] which are based on the partitioning of the network and the multirate behavior of the network.

As analogous to the basic LIM, the block-LIM seems to have a stability condition which restricts the time step size Δt . This is because the block-LIM can be regarded as a locally implicit scheme, which is based on not only an implicit difference method but also an explicit difference method, namely the leapfrog scheme. An updating calculation in the block is performed implicitly, and an updating process between the blocks is performed explicitly in the leapfrog manner. The discussion about the numerical stability of the locally implicit scheme is difficult because the scheme is based on both implicit and explicit difference methods, and includes matrix operations. At this time, we can determine the suitable time step size for the block-LIM by using a technique discussed in [8]. The technique is available for even the locally implicit scheme, instead of the conventional Von Neumann stability analysis in [1] and another method discussed in [13].

E. Simulation Algorithm

The simulation algorithm of the block-LIM is shown in Fig. 4. In Fig. 4, the numbers of the branch topologies and the node topologies, except those included in the node block or the branch block, are defined as N_b and N_n , respectively. In addition, the numbers of the branch blocks and the node blocks are defined as N_{bb} and N_{nb} , respectively. The simulation time interval is $0 \leq t \leq T$, and the total number of the time steps for a transient simulation is $n_t (= \lceil T/\Delta t \rceil)$. In this case, if the time step size Δt does not change during the simulation, the coefficient matrices $(1/\Delta t)\mathbf{L}_{ab}$ and $(1/\Delta t)\mathbf{C}_a + \mathbf{G}_a$ do not vary. Thus, the LU decomposition for each coefficient matrix of every node block and every branch block is performed only once before the loop of the transient simulation so that only

Fig. 4. Simulation algorithm of block-LIM.

the forward and the backward substitutions are performed at each time step.

IV. APPLICATION OF BLOCK-LIM TO TIGHTLY COUPLED TRANSMISSION LINES

A number of conductor lines are placed everywhere in chips and packages on printed circuit boards. These conductor lines are modeled by the tightly coupled transmission lines as shown in Fig. 5. In the circuit shown in Fig. 5, the branch blocks and the node blocks are alternately connected in series. Thus, we can separate the coupled transmission lines into block segments by using a vertical partitioning along the boundaries between the branch and the node blocks. Therefore, we can apply the block updating formulas (7) and (10) to each block individually in order to update the branch currents in the branch blocks and the node voltages in the node blocks. For example, in the case of tightly coupled transmission lines which consist of 16 transmission lines divided into 5 segments so that there exist 5 branch blocks and 6 node blocks, the coefficient matrix derived by the RLCG-MNA method can be illustrated as shown in Fig. 6. The size of the matrix is 176×176 . It is confirmed that there are 5 dense parts (in gray) corresponding to the branch blocks and 6 dense parts (in black) corresponding to the node blocks. The incidence matrix $\mathbf{A}$ is represented as the banded nonzero entries of two

Fig. 5. Topology of tightly coupled transmission lines which are composed of series connected branch blocks and node blocks.

Fig. 6. Graphical representation of coefficient matrix in RLCG-MNA formulation for tightly coupled transmission lines. The nonzero entries are shown in black corresponding to node blocks and in gray corresponding to branch blocks.

lines in the upper right part, and the distance between the two bands is equal to the number of transmission lines, namely, 16. The transposed matrix of $-\mathbf{A}$ is also stamped in the lower left part of the coefficient matrix. Note that there is no entry in the off-diagonal block parts of the upper left and the lower right matrices except the dense parts. This is because the tightly coupled transmission lines are modeled by a circuit in which there is not any mutual coupling between the blocks as shown in Fig. 5. This modeling is natural for parallel conductor lines because if two segments of the conductor lines are parallel and next to each other, they are coupled tightly, however, if they are misaligned, the coupling strength of them is sufficiently weak to be ignored. Therefore, the locally implicit scheme developed in the proposed method works effectively for the transient simulations of the tightly coupled transmission lines.

The separation of the variables described above is the most important difference from conventional methods. The domain decomposition method [14], the hierarchical technique [15], [16], the NA-approach [9] and the RLP-algorithm [17] are

TABLE I
COMPARISON OF CPU TIME.

# of lines	# of nodes	# of elements	CPU time (seconds)	
			HSPICE	block-LIM
16	177	1,689	3.44	0.23
32	513	6,161	29.90	0.48
64	1,025	23,585	331.60	1.53
128	1,409	92,353	1746.41	7.13
256	2,817	364,929	13502.44	24.00

existing techniques to reduce the costs of a fully implicit scheme in terms of computations and memory requirement. They are basically based on a circuit separation, and reduce the cost by reducing a large problem to small and low-cost problems. However, they still incur a high-cost computation such as a matrix inversion to separate a circuit to be analyzed because they are inherently fully implicit methods. On the other hand, in the block-LIM, we categorize portions of the circuit as the branch or node block. By using this separation and applying the leapfrog scheme, the block-LIM becomes a type of locally implicit scheme. As a result, our method can reduce the large problem to small problems without inducing the high-cost separating process.

V. NUMERICAL RESULTS

In this section, we apply the proposed technique to example circuits to estimate the applicability of our approach. In order to estimate the accuracy and efficiency of the block-LIM for a circuit with a number of coupling elements, several tightly coupled transmission lines are simulated. First, a pulse signal generator is appended to the leftmost node of the top line of the transmission lines shown in Fig. 5. The stimulus is a trapezoidal pulse of which the amplitude is 1 V, the rise and fall times are 0.1 ns, the pulse width is 1.0 ns, and the period is 2.2 ns, respectively. Then, we verify four cases of the transient simulations so that the numbers of the transmission lines are 16, 32, 64 and 128 in each simulation. In these cases, we assume that the transmission lines are separated vertically into 5 segments and the number of the segments does not change; there exist 5 branch blocks and 6 node blocks in every simulation. Note that the size of the coefficient matrix of each block is equal to the square of the number of lines. The values of a resistance and an inductance in each branch topology in the branch block are 0.1 Ω and 1.0 nH, respectively. The value of a grounded capacitance in each node topology in the node block is 0.01 pF. In addition, a mutual inductance between self inductances in the branch block is equal to 0.03 nH, and a mutual capacitance between nodes in the node block is equal to 1.0 fF. We also append 50.0 Ω resistances to the near end of all lines, and set the time step size 1.0 ps and the number of the total steps 4,500. This time step size is determined as described in III-D, and is sufficiently small so that the updating scheme becomes numerically stable. We observe the waveform results of the voltages at the near end and far end of the top line as well as those of the second line as a victim line.

The waveform results for the 128 transmission lines obtained by the block-LIM and HSPICE are shown in Fig. 7. In Fig. 7, there are two lines, one of which, a solid line, is for the proposed method and the other, a dashed line, is for HSPICE, respectively. It is confirmed that the waveforms from the proposed method almost completely agree with those from HSPICE. Therefore, the block-LIM is available for the simulation of the tightly coupled transmission lines for practical use. Table I shows the comparison between the block-LIM and HSPICE in terms of CPU times and the number of elements. From Table I, the block-LIM can simulate the tightly coupled transmission lines 245 times faster for 128 lines and 563 times faster for 256 lines than with HSPICE. As a result, our technique is extremely effective for fast simulations of the tightly coupled transmission lines.

VI. CONCLUSIONS

In this paper, a block matrix formulation of latency insertion method for the fast transient simulation of tightly coupled transmission lines is described. First, the basic formulation of LIM is shown. Next, the block-LIM formulation for the network with mutual inductance and mutual capacitance is described. Then, the block-LIM algorithm is applied to tightly coupled transmission lines. Finally, some numerical results are shown using example circuits. As a result, it has been confirmed that the block-LIM is useful and efficient for the simulation of tightly coupled transmission lines. For the practical use of the block-LIM, we have already built the proposed method in our simulator named *simulation program for signal and power integrity designer* (SPIDER). SPIDER can read the net-lists provided by Q3D Extractor, and perform transient simulations more than 100 times faster than HSPICE.

ACKNOWLEDGMENT

This work is partially supported by NEDO, and the authors would like to thank NEDO for the support in making this work possible.

REFERENCES

- [1] J. E. Schutt-Ainé, "Latency insertion method (LIM) for the fast transient simulation of large networks," *IEEE Trans. Circuits Syst. I*, vol. 48, no. 1, pp. 81–89, Jan. 2001.
- [2] H. Asai and N. Tsuboi, "Multi-rate latency insertion method with RLCG-MNA formulation for fast transient simulation of large-scale interconnect and plane networks," in *Proc. IEEE ECTC 2007*, Reno, NV, May 2007, pp. 1667–1672.
- [3] 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, pp. 388–397, Feb. 2007.
- [4] T. Sekine and H. Asai, "CMOS circuit simulation using latency insertion method," in *Proc. IEEE EPEP 2008*, San Jose, CA, Oct. 2008, pp. 55–58.
- [5] T. Sekine, Y. Tanji, and H. Asai, "Matrix order reduction by nodal analysis formulation and relaxation-based fast simulation for power/ground plane," *IEICE Trans. Fundamentals*, vol. E91-A, no. 9, pp. 2450–2455, Sep. 2008.
- [6] J. Choi, M. Swaminathan, N. Do, and R. Master, "Modeling of power supply noise in large chips using the circuit-based finite-difference time-domain method," *IEEE Trans. Electromagn. Compat.*, vol. 47, no. 3, pp. 424–439, Aug. 2005.

Fig. 7. Waveform results for 256 lines. Solid line: block-LIM, dashed line: HSPICE. (a) Near end of driving line. (b) Far end of driving line. (c) Near end of victim line. (d) Far end of victim line.

- [7] L. O. Chua and P. M. Lin, *Computer-Aided Analysis of Electronic Circuits: Algorithms & Computational Techniques*. New Jersey: Prentice-Hall, 1975.
- [8] J. E. Schutt-Ainé, "Stability analysis of the latency insertion method using a block matrix formulation," in *Proc. IEEE EDAPS 2008*, Seoul, Korea, Dec. 2008, pp. 155–158.
- [9] T.-H. Chen, C. Luk, and C. C.-P. Chen, "INDUCTWISE: Inductance-wise interconnect simulator and extractor," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 22, no. 7, pp. 884–894, Jul. 2003.
- [10] Y. Tanji, T. Watanabe, H. Kubota, and H. Asai, "Large scale RLC circuit analysis using RLCG-MNA formulation," in *Proc. DATE'06*, Munich, Germany, Mar. 2006, pp. 45–46.
- [11] —, "Quasi-one-step Gauss-Jacobi method for large-scale interconnect analysis via RLCG-MNA formulation," in *Proc. ISQED 2006*, San Jose, CA, Mar. 2006, pp. 395–400.
- [12] R. Gao and J. E. Schutt-Ainé, "Improved latency insertion method for simulation of large networks with low latency," in *Proc. IEEE EPEP 2002*, Monterey, CA, Oct. 2002, pp. 37–41.
- [13] S. N. Lalgudi, M. Swaminathan, and Y. Kretschmer, "On-chip power-grid simulation using latency insertion method," *IEEE Trans. Circuits Syst. I*, vol. 55, no. 3, pp. 914–931, Apr. 2008.
- [14] Q. Zhou, K. Sun, K. Mohanram, and D. C. Sorensen, "Large power grid analysis using domain decomposition," in *Proc. IEEE DATE 2006*, Mar. 2006, pp. 1–6.
- [15] M. Zhao, V. Panda, S. S. Spatnekar, and D. Blaauw, "Hierarchical analysis of power distribution networks," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 21, no. 2, pp. 159–168, Feb. 2002.
- [16] Y.-M. Lee, Y. Cao, T.-H. Chen, J. M. Wang, and C. C.-P. Chen, "Hiprime: Hierarchical and passivity preserved interconnect macromodeling engine for rlc power delivery," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 24, no. 6, pp. 797–806, Jun. 2005.
- [17] J. Jain, C.-K. Koh, and V. Balakrishnan, "Fast simulation of vlsi interconnects," in *Proc. IEEE/ACM ICCAD 2004*, Jul. 2004, pp. 93–98.

Tadatashi 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.

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.