INDUCTWISE: Inductance-Wise Interconnect Simulator and Extractor *

Tsung-Hao Chen, Clement Luk, Hyungsuk Kim, Charlie Chung-Ping Chen
{ tchen, lukc, hyungsuk }@cae.wisc.edu, chen@engr.wisc.edu
Electrical and Computer Engineering, University of Wisconsin-Madison

ABSTRACT

We develop a robust, efficient, and accurate tool, which integrates inductance extraction and simulation, called INDUCTWISE. This paper advances the state-of-the-art inductance extraction and simulation techniques and contains two major parts. In the first part, INDUCTWISE extractor, we discover the recently proposed inductance matrix sparsification algorithm, the K-method[1], albeit its great benefits of efficiency, has a major flaw on the stability. We provide both a counter example and a remedy for it. A window section algorithm is also presented to preserve the accuracy of the sparsification method. The second part, INDUCTWISE simulator, demonstrates great efficiency of integrating the nodal analysis formulation with the improved K-method. Experimental results show that INDUCTWISE has over 250x speedup compared to SPICE3. The proposed sparsification algorithm accelerates the simulator another 175x and speeds up the extractor 23.4x within 0.1% of error. INDUCTWISE can extract and simulate an 118K-conductor RKC circuit within 18 minutes. It has been well tested and released on the web for public usage. (http://vlsi.ece.wisc.edu/Inductwise.htm)

1. INTRODUCTION

Parasitic on-chip inductance is growing as another design concern as the VLSI technology marches toward ultra-deep sub-micron and the operation frequency approaches in the gigahertz range. Inductive coupling effect becomes more important because of the higher frequency signal contents, denser geometries, and reductions of both resistance and capacitance by copper and low-K devices. Inductance effect is present not only in IC packages but also in on-chip interconnects such as power grids, clock nets and bus structures. It causes signals overshot, undershot and oscillation, and aggravates crosstalk and power grid noises. All of these seriously impact the on-chip signal integrity. The importance and difficulty of on-chip inductance extraction and analysis are addressed in [2][3].

One major problem of inductance modeling is the long range coupling effects and the uncertainty of return paths. Since inductance is a function of a closed loop, the return path is hard to predict in advance before simulation. For this reason, A. Ruehli developed the famous Partial Element Equivalent Circuit method (PEEC) [4] model, which defines the partial self and mutual inductances with the assumption of infinity return paths. FastHenry [5] utilizes a multi-pole acceleration technique to speed up the extraction process in the frequency domain. [6] proposes to directly simulate the PEEC model in the time domain to determine the return paths, which has been shown to be accurate in a wide range of frequencies.

The PEEC model, however, leads to a large-scale dense inductance matrix since the long-range effect of inductive coupling and the uncertainty of current return. Traditional circuit simulation engines may require hours or even days for such a large-scale dense matrix simulation. To effectively reduce the mutual inductance terms, sparsification is crucial. It has been shown that direct truncation of the inductance matrix could result in instability [7]. Thus, a provable stable shift-and-truncate method was proposed by Krauter, et al [8]. This method assumes that the return path is no longer at infinity but within a shell. Other methods such as the Halo method [9] and the block diagonal method [10] also reduce the number of mutual inductances by limiting the return path to the nearest power and ground returns. Later Beattie, et al [11] develops an exponential shell return paths for further sparsfication and shows that the reachable sparsity is close to that of the K-method mentioned below.

Recently, the K-method has been presented by Hao Ji, et al [12][1]. K is the inverse of the partial inductance matrix L. Since K has higher degree of locality similar to capacitance, it is more satisfactory to sparsify on K than on L. Furthermore, [12] also shows that the K matrix is diagonally dominant and hence positive definite. The off diagonal terms are negative and can be safely deleted with sacrificing stability. Later Beattie, et al [11] also proposed to do double inversion on the inductance matrix and perform sparsification on both inductance and susceptance matrices.

However, there are several issues for the existing inductance handling flow. First, after inductance extraction, it is required to perform circuit simulation to verify signal integrity issues. Unfortunately there is a lack of effort to fundamentally speed up circuit simulation engine for inductance. Second, the traditional circuit simulation engines cannot or do not handle the K-elements directly or efficiently. Although the double inversion algorithm has been proposed, the runtime is compromised by the double inversion time. Third, when dealing with full-chip K-element extraction, the conductors are not equal-length and well aligned. Thus how to choose the window size becomes another new topic that no other previous works discussed about.

All the above issues lead to the urge for an inductance-oriented circuit extraction and simulation engine and that is exactly what we intend to provide in this paper. In this paper, we propose an efficient, accurate, and inductance-wise interconnect simulator and extractor, INDUCTWISE, which is based on the K-method. We further extend the original K-method and provide solutions for general circuit cases, which is applicable for full-chip K-element extraction, but not only for special cases such as equal-length parallel conductors. In addition, the utilization of nodal analysis formulation and the Cholesky decomposition also allow our INDUCTWISE to directly take K-element for simulation.

2. INDUCTWISE EXTRACTOR

In this section we first introduce the inductance and reluctance matrices and the K-method. Then we show the proof of stability for the K-method is unsustainable for general circuit cases, and provide a remedy algorithm. Finally we introduce

*This work is partially supported by NSF Grant CCR-0093309, Intel Corporation and Faraday Technology Corporation.

a novel window selection algorithm enabling INDUCTWISE to extract the K-elements in any circuit configuration.

2.1 Inductance and Reluctance Matrices

Given an inductance matrix $\mathcal{L}$, the $\mathcal{K}$ matrix is defined as $\mathcal{K} = \mathcal{L}^{-1}$ [12]. [11] names it susceptance. However, according to [13], susceptance is a general term for the imaginary part of the admittance matrix that can be caused by capacitance or inductance. Since the definition of reluctance is *the ratio of the total current force to the total magnetic flux in a magnetic circuit or component* and its unit is reciprocal henry (H^{-1}), we think *reluctance matrix* is more specific to the inverse inductance matrix, $\mathcal{K}$.

Each element in the partial inductance matrix is given by

$$\mathcal{L}_{ij} = \frac{\mu_0}{4\pi a_i a_j}\left[\int_{a_i}\int_{a_j}\int_{l_i}\int_{l_j}\frac{dl_i \cdot dl_j}{r_{ij}} da_i da_j\right] \quad , \qquad (1)$$

where a_i and a_j are cross-sections of segment i and j respectively, and r_{ij} is the geometric distance between two points in segment i and j. The magnetic vector potential along segment i caused by current $\mathbf{I}_j$ in segment j is defined as follows.

$$A_{ij} = \frac{\mu_0}{4\pi a_j}\left[\int_{a_j}\int_{l_j}\frac{\mathbf{I}_j}{r_{ij}} dl_j da_j\right] \qquad (2)$$

Therefore, for an $n \times n$ partial inductance matrix, the corresponding linear system equation can be written as follows.

$$\begin{bmatrix} \mathcal{L}_{11} & \mathcal{L}_{12} & \cdots \\ \mathcal{L}_{21} & \mathcal{L}_{22} & \cdots \\ \cdots & \cdots & \mathcal{L}_{nn} \end{bmatrix} \begin{bmatrix} i_1 \\ \vdots \\ i_n \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{n}(\int A_{1i} \cdot dl_1) \\ \vdots \\ \sum_{i=1}^{n}(\int A_{ni} \cdot dl_n) \end{bmatrix} \qquad (3)$$

Representing the system equation with $\mathcal{K}$, we get

$$\begin{bmatrix} \mathcal{K}_{11} & \mathcal{K}_{12} & \cdots \\ \mathcal{K}_{21} & \mathcal{K}_{22} & \cdots \\ \cdots & \cdots & \mathcal{K}_{nn} \end{bmatrix} \begin{bmatrix} \sum_{i=1}^{n}(\int A_{1i} \cdot dl_1) \\ \vdots \\ \sum_{i=1}^{n}(\int A_{ni} \cdot dl_n) \end{bmatrix} = \begin{bmatrix} i_1 \\ \vdots \\ i_n \end{bmatrix} \qquad (4)$$

2.2 Stability Issues of the K-Method

H. Ji, et. al [1] developed an advanced reluctance sparsification method called K-method. They showed that $\mathcal{K}$ has better locality than $\mathcal{L}$, and thus sparsifying on $\mathcal{K}$ actually benefits more efficiency. They proved the stability of their algorithm based on the diagonal dominance property, which is derived from the assumption that all off-diagonal terms of $\mathcal{K}$ are negative. We now show that the property does not hold for general interconnect configurations.

From Equation (4), the physical meaning of $\mathcal{K}_{ij}$ is defined by the induced current in the i^{th} conductor when the total flux for the j^{th} conductor is equal to one and those along all other conductors are set to zero. For example, as shown in Figure 1, to get the 3^{rd} row of $\mathcal{K}$, we apply flux 1 to conductor 3 and 0 to all the others. The induced currents on conductors other than the 3^{rd} are the off diagonal terms on the 3^{rd} row of $\mathcal{K}$. It is positive if the current direction is the same as the applied flux direction, or negative otherwise. [1] argues that all the induced current are negative and use this property to proof the stability of their algorithm. We find out that the off-diagonal terms are not necessary to be negative for general wire cases.

In this example, the partial inductance matrix is calculated as follows:

$$\mathcal{L} = \begin{bmatrix} 1.04 & 0.34 & 0.37 & 0.24 & 0.51 \\ 0.34 & 0.45 & 0.09 & 0.06 & 0.27 \\ 0.37 & 0.09 & 1.04 & 0.34 & 0.41 \\ 0.24 & 0.06 & 0.34 & 0.45 & 0.11 \\ 0.51 & 0.27 & 0.41 & 0.11 & 1.69 \end{bmatrix} \times 10^{-10} H \ . \qquad (5)$$

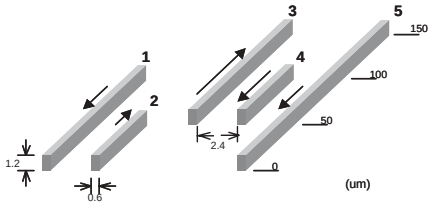


Figure 1: An example for parallel conductors with unequal lengths

By inverting $\mathcal{L}$, the $\mathcal{K}$ matrix can be obtained:

$$\mathcal{K} = \left[\begin{array}{rrrrr} 1.57 & -0.94 & -0.22 & -0.47 & -0.25 \\ -0.94 & 3.02 & 0.15 & 0.01 & -0.23 \\ -0.22 & 0.15 & 1.42 & -0.93 & -0.24 \\ \hline -0.47 & 0.01 & -0.93 & 3.12 & 0.16 \\ -0.25 & -0.23 & -0.24 & 0.16 & 0.75 \end{array}\right] \times 10^{10} H^{-1} \ . \qquad (6)$$

It is clear that some off-diagonal terms in (6) are positive. For instance, when calculating the 3^{rd} row, a unit flux is assigned on conductor 3, which demands a positive current along conductor 3 to accomplish. This current induces a positive magnetic flux along all other conductors (let's consider only conductors 1 and 2 in this explanation). To compensate this effect and make the net magnetic fluxes along 1 and 2 equal to zero, they have to carry negative currents. However, the induced current along 1 also induces another current along 2. Since the coupling effect between 1 and 2 is much stronger than that between 3 and 2, the overall effect causes conductor 2 carrying a positive current direction.

Therefore, the physical definition of $\mathcal{K}$ in [1] should be calibrated as follows: *The magnetic flux along all conductors except the j^{th} are set to zero, and that along the j^{th} conductor is set to one. To satisfy the condition, there exists some current running along each conductor. The elements $\mathcal{K}_{ij}$ is the current flowing through the i^{th} conductor. Thus their overall effect satisfies the magnetic flux assumption.* Since the result is due to the overall effect (not a single active line), negative off-diagonal elements are not guaranteed any more. This invalids the proof of the diagonal dominance property and hence the stability for the K-method becomes unsure. We will propose our remedies next.

2.3 Formal Analysis

In this section, we first present the K-method stability analysis through the duality of electrical and magnetic fields at Maxwell's equations. From the duality analysis, we shed the insights of the similarity of inductance and capacitance. Through the theorems provided in this section, we are able to propose a correction to the K-method to ensure the stability. From the Maxwell's equations, we have

$$\nabla \times \vec{E} = -\mu s \vec{H} \ , \qquad (7)$$

$$\nabla \times \vec{H} = \epsilon s \vec{E} + \vec{J} \ , \qquad (8)$$

$$\nabla \cdot \epsilon \vec{E} = \rho \ , \qquad (9)$$

$$\nabla \cdot \mu \vec{H} = 0 \ . \qquad (10)$$

The definition of the magnetic vector potential gives

$$\mu \vec{H} = \nabla \times \vec{A} \ . \qquad (11)$$

Applying (11) to Equation (7), we get

$$\nabla \times (\vec{E} + s\vec{A}) = 0 \ . \qquad (12)$$

This implies that there exists a scalar potential, V, such that

$$\vec{E} + s\vec{A} = -\nabla V \ . \qquad (13)$$

To uniquely determine $\vec{A}$, we choose the Lorentz gauge,

$$\nabla \cdot \vec{A} = -\epsilon \mu s V \ . \qquad (14)$$

By Equations (8), (11), (14), and the identity $\nabla \times (\nabla \times \vec{A}) = \nabla(\nabla \cdot \vec{A}) - \nabla^2 \vec{A}$, we can get

$$\nabla^2 \vec{A} - \mu\epsilon s^2 \vec{A} = -\mu \vec{J}. \quad (15)$$

Similarly by Equations (9)(12) and the Coulomb gauge, we get

$$\nabla^2 V - \mu\epsilon s^2 V = -\rho/\epsilon. \quad (16)$$

Equations (15) and (16) are often referred as the **nonhomogeneous Helmholtz's equations**. The solutions of Equations (15) and (16) are

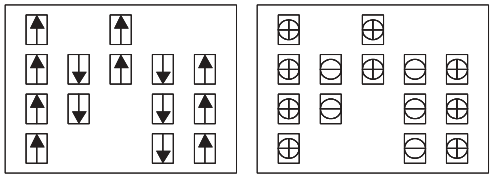
$$\vec{A}(r) = \frac{\mu}{4\pi} \int_{V'} G(r,r') \vec{J}(r') e^{s/c|r-r'|} dv' \ , \quad (17)$$

$$V(r) = \frac{1}{4\pi\epsilon} \int_{V'} G(r,r') \rho(r') e^{s/c|r-r'|} dv' \ , \quad (18)$$

in which V' is the volume of all conductors, $c = 1/\sqrt{\mu\epsilon}$, and $G(r,r') = \frac{1}{4\pi R_{ij}}$, where $R_{ij} = |r - r'|$ is the Green's function. The dual property between a magnetic and an electric problems can be observed from Equations (15)(16)(17) and (18). The major difference between $\vec{A}$ and V is that $\vec{A}$ is a directional vector and V is a scalar. There exists a transformation from a magnetic problem to a electric problem, which is described in the following lemma.

Lemma 1. *Given an uni-directional magnetic nonhomogeneous Helmholtz's equation problem, there exists a corresponding electric nonhomogeneous Helmholtz's equation problem that has the same solution.*

PROOF. Since all the magnetic sources and mediums are uni-directional, we can remove the vector natural by proper assigning the positive charge corresponding to the forward direction or negative otherwise. Hence, given current sources vector $\vec{J}$, we can create a corresponding charges $\rho = \mu\epsilon\vec{J}$ with proper signs assigned, and the solution of Equations (15) and (16) are identical. □



(a) Inductance model (b) Capacitance model

Figure 2: Dual property between inductance and capacitance problems

Figure 2 illustrates the transformation of Lemma 1. From this lemma, we can get the following theorem.

Theorem 1. *The reluctance matrix, $\mathcal{K}$, is diagonally dominant and symmetric positive definite when all the conductors are sufficiently discretized.*

PROOF. Lemma 1 shows that every uni-directional magnetic problem can be transformed into an electric one. Since it has been shown that the capacitance matrix is always diagonally dominant and symmetric positive definite for sufficiently discretized problem [4], Theorem 1 thus follows. □

Theorem 1 reveals why the diagonal dominance property of the reluctance matrix does not always hold. The answer is **discretization**. When we performing capacitance extraction, the conductors are usually well discretized. In the contrary, the conductors are always preserved as long wires while we perform inductance extraction. The length of inductance discretization is often hundred times larger than the capacitance discretization. Therefore, we come up with the remedy in the following subsection.

2.4 RBC Algorithm - Guarantee the Stability

We have already shown that finer discretization guarantees the stability of the K-method. It can also increase the accuracy. However, if we uniformly discretize conductors into small pieces, the complexity of solving this problem will become enormous, which losses the original intention of the sparsification. Therefore, we propose a cutting algorithm to obtain a stable reluctance matrix without too much prejudice to the simulation run time. This algorithm is called Recursive Bisection Cutting Algorithm (RBCA), which is based on the idea that the reluctance matrix is diagonally dominant and symmetric and positive definite (s.p.d.) when the conductors are sufficiently discretized as in capacitance problems.

From the previous discussion, the diagonal dominance of $\mathcal{K}$ is strongly related to unequal-length and misalignment cases. Theorem 1 also tells that smaller discretization is better than larger one, which implies that longer conductor actually acts a critical role in this problem. Thus, the basic idea of the RBCA is to recursively cut the longest conductor when the positive off-diagonal element occurs during the K-method procedure. In order to make sure the RBCA result in all negative off-diagonal elements in $\mathcal{K}$, we perform the K-method with a small window (will discuss how to choose this window in the later section) and check if every off-diagonal element in the small K-matrix is negative. If there exists any positive off-diagonal value, we cut the longest conductor in this window. After this cutting, back-trace those conductors that are reluctance-coupled with this cut one. If this cutting causes new positive off-diagonal value to any previously processed conductor, we recursively cut the troublemaker conductors. Iteratively repeat this until the final K-matrix has all positive off-diagonal entries. The RBCA is summarized as in Table 1.

For each conductor j
1. Choose a window W.
2. Calculate $\mathcal{K}_{ij}$, where $i \in W$, as in the K-method.
3. If $\exists \mathcal{K}_{kj} > 0$, $k \in W$ and $k \neq j$, do the following:
   a. cut the longest conductor l, $l \in W$, by half.
   b. Back-trace and reperform the small matrix inversion for the i^{th} conductor that $\mathcal{K}_{ki} \neq 0$, where $i \in W$ and $i < j$. If this cutting causes any new positive off-diagonal term, recursively perform the cutting to the troublemaker conductor.

Table 1: The Recursive Bisection Cutting Algorithm (RBCA)

Theorem 2. *The RBCA guarantees all non-positive off-diagonal elements in K, and hence the s.p.d. property validates the proof of stability in [1].*

The proof of Theorem 2 follows by the recursion and Theorem 1. We've already known that positive off-diagonal values happen when conductors have seriously mismatched lengths or mis-aligned organizations. Since all of the previous works [1][11][14] considered only equal-length parallel conductors, the exception case we show doesn't exist. However, to build a full-chip inductance (reluctance) extractor, this possibility does exist. We propose this cutting algorithm serving as the stability guard of our INDUCTWISE extractor to insure the s.p.d. of the sparse reluctance matrix. We will show how to select the window in general circuit configurations in the following section.

2.5 WS Algorithm - Capture Significant Effect

All previous works used equal-length parallel conductors as their examples to show the benefits of sparsifying on reluctance matrices. It is not clear if the K-method can work on general irregular geometries. The lack of generality limits the application of the K-method only to analysis some special configurations such as buses. However, general routing cases are more irregular, which might contain uneven-length or misaligned conductors. For these cases, it's very difficult to determine what a "window" is when performing the window-based K-method. In this section, we propose a novel algorithm to determine what should be included in the window when we extract sparse reluctance matrices. Let's first define the terminology using in the following discussion by the example circuit in Figure 3.

Aggressors and Victims: When performing the K-method and calculating one of the columns in $\mathcal{K}$, we set the magnetic flux along the corresponding conductor to one named the **aggressor**, and others to zero called **victims**. In Figure 3, we assume that conductor 1 is now the aggressor and others are victims.

ESF, ESR and ESA: Supposed the aggressor has length L, we now define the **extended search factor (ESF)**, x, such that the **effective search range (ESR)** extends both ends of the aggressor out xL (i.e. segment $\overline{ah}$). The ESR is a strip with length $(1+2x)L$. Then the **effective search area (ESA)** is defined by sweeping from left infinity to right infinity with the ESR. The ESA is marked by slash lines in Figure 3.

Shields and Shielding Level: If a victim is partially or fully in the ESA, it is called a **shield** for the aggressor. For example, in Figure 3 conductors 2, 4, 5, and 6 are shields of 1, but 3 is not. The shielding level indicates how close the victim shields the aggressor. If there exist k shields between a shield and the aggressor, the shield is with the $(k+1)^{th}$ level. For example, conductor 2 is the 1^{st} level of shield for segment $\overline{ad}$, and conductor 4 is the 2^{nd} level of shield for $\overline{bd}$. Conductor 5 contains two part. The upper part is the 3^{nd} level of shield for $\overline{bd}$ and the lower part 1^{st} level of shield for $\overline{dg}$.

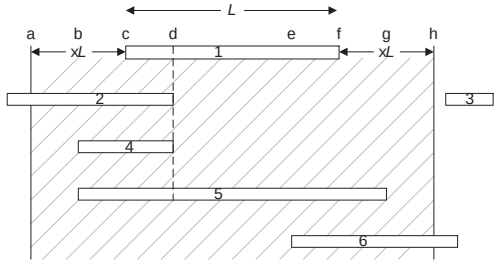


Figure 3: A example for the terminology definition

We now discuss how the K-method works. From the physical point of view, the experiment results in [1] demonstrates that the shielding effect for the mutual reluctance does exist but not for partial mutual inductance. From the numerical point of view, the K-method tries to select the most significant values on a column (row) and inverts it. The inversion causes the off-diagonal values of the K decreasing in a rapider way than L. This fact makes the reluctance element has better locality than the partial inductance. Therefore, properly selecting relatively significant couplings within the small window actually dominates the accuracy of the algorithm. However, the K-method exists some difficulties for irregular geometries. First, the strength of coupling does not strictly decrease as the distant between conductors increase, so the closer one may not be the more significant one. This means that the further conductor may have stronger coupling but is not included in the window. Second, an intuitional solution is to select the largest inductive coupling values in the small window. To find the most significant value in the L-matrix, we have to extract all the partial mutual inductance values, which makes the extraction complexity $O(n^2)$ and losses the efficiency. Moreover, this solution leads to a topologically asymmetric K-matrix and makes the later-on symmetrization process introducing more errors to the final sparse K-matrix.

In Equation (1), the inner product of $dl_i \cdot dl_j$ implies that the mutual inductance has a large value when two conductors are parallel and next to each other, and has a small value when they are misaligned. If two conductors are perpendicular, their partial mutual inductance is zero. From these observations and utilizing the shielding effect of reluctance elements, given the ESF x and the desired shielding level k, what conductors should be included in the small window are determined by our window selection algorithm (WSA) summarized in Table 2.

1. Divide all conductors into vertical and horizontal sets.
2. For the vertical set, sort by their x coordinates.
3. For every conductor from left to right, do the following:
   a. Search from the first victims next to the aggressor, and select those shields (by definition) until every segment on the ESR (i.e. $\overline{ah}$ in Figure 3) are shielded no less than k levels. This forms the right part of the window.
   b. The left part of the window is obtained from the window-selection results of previous conductors.
   c. Set the flux along the aggressor to one and others to zero, and solve the reluctance elements by inverting the small matrix corresponding to the window. The resulting elements form the column of $\mathcal{K}_{asym}$ that corresponds to the aggressor.
4. Analogically repeat steps 2 and 3 for the horizontal set.
5. Symmetrize by $\mathcal{K} = \frac{1}{2}(\mathcal{K}_{asym} + \mathcal{K}_{asym}^T)$.

Table 2: The Window Selection Algorithm (WSA)

Notice that we select the shields until every arbitrarily segment on the ESR are shielded no less than k levels to ensure we capture the significant effect. For example, if we set the level of shielding to 1 in Figure 3, the victims selected should be conductors 2, 5, and 6. Thus all points on $\overline{ah}$ are shielded at least once. In this algorithm, we only have to search the right-hand-side shields for each aggressor. The left-hand-side shields can be obtained from the previous results. It is obvious that A is B's n^{th} level of shield, then B must be A's n^{th} level of shield. This observation allows us to use the previous evaluated mutual inductance values and the window information, which can save half of the extraction time. The obtained K-matrix is topologically symmetric.

3. INDUCTWISE SIMULATOR

In this session, we present our efficient time domain RLKC INDUCTWISE simulator. We will first focus on two circuit matrix formulations MNA (Modified Nodal Analysis) and NA (Nodal Analysis). Later, the way to deal with independent source in the NA formulation and the pros and cons of these two formulations will be discussed.

3.1 MNA Approach

First, we briefly review the MNA equations. Given a linear circuit, the MNA formulation can be expressed as follows,

$$\tilde{\mathbf{G}}\mathbf{x} + \tilde{\mathbf{C}}\dot{\mathbf{x}} = \mathbf{b} \ , \tag{19}$$

$$\text{in which} \quad \begin{array}{ll} \tilde{\mathbf{G}} = \begin{bmatrix} \mathbf{G} & \mathbf{A}_l^T \\ -\mathbf{A}_l & \mathbf{0} \end{bmatrix}, & \mathbf{x} = \begin{bmatrix} \mathbf{v}_n \\ \mathbf{i}_l \end{bmatrix}, \\ \tilde{\mathbf{C}} = \begin{bmatrix} \mathbf{C} & \mathbf{0} \\ \mathbf{0} & \mathcal{L} \end{bmatrix}, & \mathbf{b} = \begin{bmatrix} -\mathbf{A}_i^T \mathbf{I}_s \\ \mathbf{0} \end{bmatrix}. \end{array} \tag{20}$$

$\mathbf{G} = \mathbf{A}_g^T \mathcal{G} \mathbf{A}_g$ and $\mathbf{C} = \mathbf{A}_c^T \mathcal{C} \mathbf{A}_c$. $\mathcal{G}$, $\mathcal{C}$ and $\mathcal{L}$ are conductance, capacitance and inductance matrices respectively. $\mathbf{A}$'s are the adjacency matrices of the circuit, whose subscriptions g, c, l and i associate with $\mathcal{G}$, $\mathcal{C}$, $\mathcal{L}$ and $\mathbf{I}_s$ respectively. $\mathbf{I}_s$, $\mathbf{i}_l$, $\mathbf{v}_n$ are vectors of current sources, inductance current variables and nodal voltage variables respectively. For transient analysis, the trapezoidal integration approximation of Equation (19) over time interval $[kh, (k+1)h]$ is given by

$$\tilde{\mathbf{G}}\left(\frac{\mathbf{x}^{k+1} + \mathbf{x}^k}{2}\right) + \tilde{\mathbf{C}}\left(\frac{\mathbf{x}^{k+1} - \mathbf{x}^k}{h}\right) = \frac{\mathbf{b}^{k+1} + \mathbf{b}^k}{2}$$

It can be rewritten as follows,

$$\left(\tilde{\mathbf{G}} + \tfrac{2}{h}\tilde{\mathbf{C}}\right)\mathbf{x}^{k+1} = \left(-\tilde{\mathbf{G}} + \tfrac{2}{h}\tilde{\mathbf{C}}\right)\mathbf{x}^k + \mathbf{b}^{k+1} + \mathbf{b}^k \tag{21}$$

The MNA approach works for ordinary inductance sparse approximations, but not for the reluctance matrix. In this paper, we use this method to solve the exact solution.

3.2 NA Approach

In this subsection, we will show that Nodal Analysis is feasible for sparse reluctance matrix, and even has more advantages than MNA methods. Substituting (20) into (21) and performing block matrix operations, we can obtain two equations as follows,

$$\begin{aligned} &\left(\mathbf{G} + \frac{2}{h}\mathbf{C}\right)\mathbf{v}_n^{k+1} + \mathbf{A}_l^T \mathbf{i}_l^{k+1} \\ &\quad = \left(-\mathbf{G} + \frac{2}{h}\mathbf{C}\right)\mathbf{v}_n^k + \mathbf{A}_l^T \mathbf{i}_l^k - \mathbf{A}_i^T\left(\mathbf{I}_s^{k+1} + \mathbf{I}_s^k\right) \end{aligned} \tag{22}$$

$$-\mathbf{A}_l \mathbf{v}_n^{k+1} + \frac{2}{h}\mathcal{L}\mathbf{i}_l^{k+1} = \mathbf{A}_l \mathbf{v}_n^k + \frac{2}{h}\mathcal{L}\mathbf{i}_l^k \tag{23}$$

Rearranging Equations (22) and (23), we can get

$$\begin{aligned} \left(\mathbf{G} + \frac{2}{h}\mathbf{C} + \frac{h}{2}\mathbf{K}\right)\mathbf{v}_n^{k+1} &= \left(-\mathbf{G} + \frac{2}{h}\mathbf{C} - \frac{h}{2}\mathbf{K}\right)\mathbf{v}_n^k \\ &\quad -2\mathbf{A}_l^T \mathbf{i}_l^k - \mathbf{A}_i^T\left(\mathbf{I}_s^{k+1} + \mathbf{I}_s^k\right) \end{aligned} \tag{24}$$

$$2\mathbf{A}_l^T \mathbf{i}_l^{k+1} = h\mathbf{K}\left(\mathbf{v}_n^{k+1} + \mathbf{v}_n^k\right) + 2\mathbf{A}_l^T \mathbf{i}_l^k \tag{25}$$

where $\mathbf{K} = \mathbf{A}_l^T \mathcal{K} \mathbf{A}_l$, where $\mathcal{K}$ is the reluctance matrix that equals to $\mathcal{L}^{-1}$. Let $\mathbf{Y} = \left(\mathbf{G} + \frac{2}{h}\mathbf{C} + \frac{h}{2}\mathbf{K}\right)$ be the admittance matrix. Since $\mathbf{G}$, $\mathbf{C}$ and $\mathbf{K}$ are all s.p.d., $\mathbf{Y}$ can be easily proven to be s.p.d. Hence the Cholesky Decomposition or the Preconditioned Conjugate Gradient iterative method is applicable for the NA formulation. Using the reluctance extraction algorithms shown in the previous section, matrix $\mathbf{K}$ is sparse, which keeps $\mathbf{Y}$ still a sparse matrix. This result is extended from our previous work [15]; please refer to it for detail derivation. [14] has similar discovery independently.

3.3 A Comparison Study

Since the MNA matrix $\left(\tilde{\mathbf{G}} + \frac{2}{h}\tilde{\mathbf{C}}\right)$ in (21) is asymmetric, LU factorization is unavoidable. On the contrary, matrix $\mathbf{Y}$ in the NA is s.p.d., which validates the Cholesky decomposition. There are several well-known benefits of the Cholesky decomposition over the LU decomposition. First, the runtime and memory requirements of the Cholesky decomposition are half as those of the LU decomposition since the former can take the advantage of the symmetricity. Second, the LU decomposition requires pivoting algorithms to enhance numerical conditions and avoid breaking down. It has been shown that the accuracy of the Cholesky decomposition is always the best regardless of the matrix ordering. Matrix reordering for the Cholesky is usually performed only for fill-in reduction and only topologically. The sparsity of the NA formulation is often slightly worse than MNA since $\mathbf{K} = \mathbf{A}_l^T \mathcal{K} \mathbf{A}_l$ introduces more matrix entries than $\mathcal{L}$. However, we believe that the additional entries are offset by the saving of symmetricity.

It is well known that the computation time of the factorization is dominated by the number of fill-ins and the matrix ordering plays a crucial role to the fill-ins. The reduction of fill-ins not only saves the runtime of the decomposition but also has tremendous benefit for later transient simulation since we have a smaller amount of matrix entries in the triangle matrices. It is also known that it is easier and more efficient to perform matrix reordering to symmetric matrices.About matrix reordering algorithms, there are just so many of them such as RCM (Reverse Cuthill-McKee), MD (Minimum Degree), ND (Nested Dissection) methods and their variants. To the authors' knowledge and experimental results, we discover that MD is one of the most efficient ways to reduce fill-ins for time-domain circuit simulators.

3.4 Handling Independent Voltage Sources

In case there are independent voltage sources in the circuit, we have to add extra current variables in the MNA equations. In fact, if we transform voltage sources into Norton equivalent currents, there is no need to add any current variables. Thus the symmetricity of the NA formulation can be preserved since the Norton equivalent currents only affect the right-hand side in Equation (24). If the voltage source connects to R or C elements, this method can be easily implemented. Norton equivalent circuits for R and C elements are available. However, coupling of inductances makes this transformation ineligible for L elements. Consider the circuit shown in Figure 4(a), which shows a voltage source connects to one terminal of two coupled inductances. Using frequency domain analysis, the current-voltage equations on its two ports are as follows,

$$\begin{aligned} I_1 &= \frac{K_{11}}{s}(V_1 - V_K) + \frac{K_{12}}{s}V_2 \\ I_2 &= \frac{K_{12}}{s}(V_1 - V_K) + \frac{K_{22}}{s}V_2 \end{aligned}$$

These two equations can be rewritten in the following format,

$$\begin{aligned} I_1 &= \frac{K_{11}}{s}V_1 + \frac{K_{12}}{s}V_2 - \frac{K_{11}}{s}V_S \\ I_2 &= \frac{K_{12}}{s}V_1 + \frac{K_{22}}{s}V_2 - \frac{K_{12}}{s}V_S \end{aligned}$$

which can be represented as the circuit shown in Figure 4(b). The voltage source is replaced by current sources. The NA approach is still feasible if reluctance elements are present.

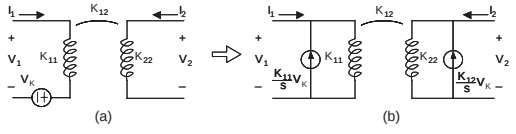


Figure 4: Norton equivalent transformation for $\mathcal{K}$

			Attacker		Faraway victim	
Shielding Level	ESF	Density	1^{st} peak error	2^{nd} peak error	1^{st} peak error	1^{st} droop error
1	0.0	4.0%	3.21%	-2.26%	-33.52%	-51.72%
1	0.5	4.6%	0.55%	0.62%	-26.46%	-37.70%
1	1.0	5.0%	0.52%	0.82%	-27.19%	-35.63%
2	0.5	8.1%	-0.33%	0.72%	-11.59%	-22.99%
3	0.5	11.4%	0.12%	1.23%	-11.30%	-22.07%
5	0.5	17.5%	0.20%	1.91%	-3.85%	-4.85%

Table 3: Accuracy comparison for different shielding levels and ESFs (154 conductor segments)

4. EXPERIMENTAL RESULT

We develops our INDUCTWISE reluctance extractor and simulator in C/C++ programming language. The extractor implements our WSA-based K-method with the RBCA stability guarantee. The simulator implements both MNA and NA solutions, which is able to simulate both inductance and reluctance cases. The simulations are run on an Intel Pentium IV 1.4GHz system.

Table 3 shows the accuracy information for a circuit with 154 conductors. Each driving end has a voltage source connected to the nearest ground wire, and each loading end has load capacitors connected to both the power and ground lines. We activate one of the driving sources, which is called attacker, with an 1-volt step function, and observe the responses of the loading ends of both the attacker and a faraway victim that is 10 conductors away from the attacker. From Table 3, we find the shielding level and the ESF affect the accuracy in the following trend. Enlarging the ESF improves the accuracy on the attacker, but not the victim. On the contrary, enlarging the shielding level helps improve the accuracy on the faraway victim, but has less effectiveness to the attacker. Figures 5(a) and (b) show the waveforms for different parameters and illustrate this trend. In this case, shielding level 1 with 0.5 ESF can already approach the exact solution very well for the active conductor, and a higher shielding level even improves the faraway accuracy more. The WSA with shielding level 5 and ESF 0.5 almost matches the exact solution.

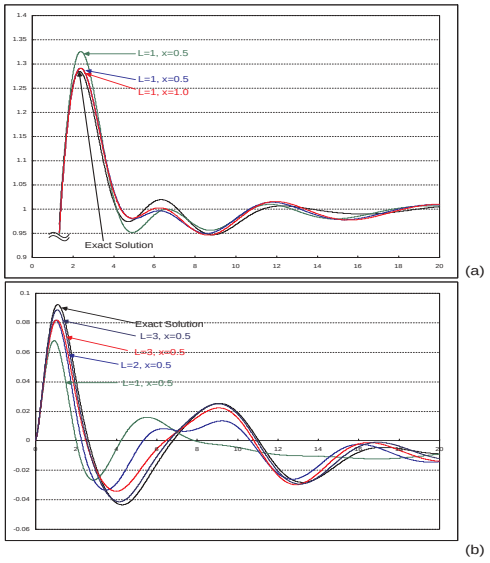


Figure 5: Waveforms of (a)Attacker @different ESFs (b) Faraway victim @different shielding levels

Table 4 shows the runtime information of INDUCTWISE. The RC setup is the same as described in the previous paragraph. We perform each simulation for 200 time steps. For the 550-conductor case, SPICE3 [16] takes 1799.1 seconds to solve the exact solution while INDUCTWISE only takes 7.2 seconds (250x speedup). Due to the superlinear dependence of solve time on matrix size, the speed up will be more dramatic for larger systems. For the sparse reluctance solution, we set the shielding level to 3 and the ESF to 0.5, and have 23.4x and 175x extraction and simulation speedup respectively for the 1862-conductor case. It can extract and simulate an 118K-conductor RKC circuit within 18 minutes.

	Full Inductance			Sparse Reluctance	
# of cond.	Extract time(s)	SPICE3 time(s)	I.W. Sim. time(s)	Extract time(s)	I.W. Sim. time(s)
154	1.6	22.7	0.4(61x)	0.6(2.6x)	0.2(2.1x)
550	21.2	1799.1	7.2(250x)	2.6(8.1x)	0.7(9.9x)
1,862	243.2	>2days	559.3	10.4(23.4x)	3.2(175x)
20,944	-	-	-	115.1	49.8
118,275	-	-	-	648.1	425.4

Table 4: Run time comparison between INDUCTWISE (RLC and RKC) and SPICE3 (RLC)

5. REFERENCES

[1] H. Ji, A. Devgan, and W. Dai. Ksim: A stable and efficient rkc simulator for capturing on-chip inductance effect. In *ASPDAC*, June 2001.

[2] M.W. Beattie and L.T. Pileggi. Inductance 101: Modeling and extraction. In *DAC*, June 2001.

[3] K. Gala, D. Blaauw, J. Wang, V. Zolotov, and M. Zhao. Inductance 101: Analysis and design issues. In *DAC*, June 2001.

[4] A.E. Ruehli. Inductance calculatioin in a complex integrated curcuit environment. *IBM Journal of Research and Development*, pages 470–481, September 1972.

[5] M. Kamon, M.J. Tsuk, and J.K. White. Fasthenry: A multipole-accelerated 3-d inductance extraction program. In *DAC*, June 1993.

[6] N. Chang, S. Lin, J. Lillis, and C.K. Cheng. *Inteconnect Analysis and Synthesis*. John Wiley & Sons Inc., 2000.

[7] Z. He, M. Celik, and L.T. Pileggi. Spie: Sparse partial inductance extraction. In *DAC*, 1997.

[8] B. Krauter and L.T. Pileggi. Generating sparse partial inductance matrices with guaranteed stability. In *ICCAD*, November 1995.

[9] K.L. Shepard and Z. Tian. Return-limited inductances: A practical approach to on-chip inductance extraction. *IEEE Trans. on Computer Aided Design of Integrated Ciurcuits and Systems*, 19(4):425–436, April 2000.

[10] K. Gala, V. Zolotov, R. Panda, B. Young, J. Wang, and D. Blaauw. On-chip inductance modeling and analysis. In *DAC*, June 2000.

[11] M.W. Beattie and L.T. Pileggi. Modeling magnetic coupling for on-chip interconnect. In *DAC*, June 2001.

[12] A. Devgan, H.Ji, and W. Dai. How to efficiently capture on-chip inductance effects: Introducing a new circuit element k. In *ICCAD*, November 2000.

[13] http://whatis.techtarget.com/definition/0,,sid9_gci531197,00.html.

[14] H. Zheng, B. Krauter, M. Beattie, and L. Pileggi. Window-based susceptance models for large-scale rlc circuit analysis. In *DATE*, 2002.

[15] T.H. Chen and C.C.P. Chen. Efficient large-scale power grid analysis based on preconditioned krylov-subspace iterative methods. In *DAC*, June 2001.

[16] L.W. Nagel. Spice2: A computer program to simulate semiconductor circuits. *U.C. Berkely*, ERL Memo ERL-M520, 1975.