# Comprehensive Frequency Dependent Interconnect Extraction and Evaluation Methodology

Rong Jiang
Electrical and Computer Engineering
University of Wisconsin-Madison
Madison, Wisconsin, 53706-1691
jiang@cae.wisc.edu

Charlie Chung-Ping Chen
Electrical Engineering
National Taiwan University
Taipei 106, Taiwan
cchen@cc.ee.ntu.edu.tw

***Abstract*— Frequency dependent interconnect analysis is challenging since lumped equivalent circuit models extracted at different frequencies exhibit distinct time and frequency domain responses, and consequently, the analysis of a single equivalent circuit is inapplicable to evaluate the interconnect behavior at other frequencies. In this paper, we present a wide frequency range interconnect extraction and analysis methodology. First, an improved reluctance-based extraction algorithm is proposed to generate compact interconnect models at some sample frequencies. Then, DLSCF (Discrete Least Square Curve Fitting) techniques are employed to produce approximation polynomials to calculate parasitics at other frequencies. Finally, after transferring those approximation polynomials into power series of $s$ and substituting them into the MNA (Modified Nodal Analysis) formula, we develop and apply the WIFRIM (Wide Frequency Range Interconnect Moment Matching) algorithm to calculate moments of arbitrary orders. Since WIFRIM only needs to decompose a sparse conductance matrix once, it results in significant speedup while providing accuracy within 1% error.**

## I. Introduction

The irrepressible march toward smaller integrated circuit and faster operation frequency has made accurate and efficient analysis of on-chip interconnects a critical issue for guaranteeing signal integrity and satisfying design specifications. However, the increasing visibility of high frequency effects, such as skin effect, proximity effect, and substrate effect, leads to inherently frequency dependent interconnect models. Those lumped equivalent circuit models extracted at different frequencies may exhibit distinct time and frequency domain responses. Apparently, it is impractical to perform extraction and simulation at every frequency within a wide frequency range of interest. Therefore, how to efficiently analyze frequency dependent interconnects has become one of the designer's high priority concerns.

One difficulty of constructing interconnect models is to capture the long-range inductive effects. Reluctance-based methods [1]–[4] have been proved to be accurate and efficient. Those methods benefit from the great locality and shielding effect of reluctance, and hence lead to superior sparse reluctance matrices compared to the traditional PEEC-based methods [5], [6]. Furthermore, the reluctance matrix is stable and positive diagonal dominant, which makes it even more attractive for circuit simulation.

Most existing inductance and reluctance extraction algorithms utilize static formulae to calculate inductance values. One prerequisite of applying those formulae is that the current density along a conductor segment is evenly distributed. However, as frequency grows over $1gHz$, the skin depth of a conductor segment varies from $1\mu m$ to $20\mu m$, while its width may far exceed the skin depth, especially for global interconnects within the upper metal layers. Therefore, conductor segments are necessarily discretized into more delicate filaments and currents within those thin filaments are assumed uniform. Although discretization captures non-uniformly distributed currents, it also increases the model size dramatically. Furthermore, different frequencies give rise to different discretization schemes, and hence result in various equivalent circuit models.

Simulating those interconnect models is even more arduous due to their astounding sizes. Moment matching techniques [7]–[9] have been extensively studied to evaluate interconnect systems in the past decade. However, existing moment matching methods are applicable when lumped RLKC circuits are invariant and independent of frequency. Therefore, the whole frequency domain spectrum can only be obtained by extracting and simulating different frequency dependent models many times, since the simulation result of one model is exclusively valid for the particular frequency at which it is extracted.

Unsatisfied with those existing interconnect modeling and simulation problems, this paper proposes an efficient methodology to generate interconnect models and perform frequency dependent interconnect analysis. Our contributions in this paper are as follows: (1) we device a new reluctance extraction algorithm with simultaneous parallel filaments reduction. This method can generate compact circuit models without incurring additional computation time and sacrificing any accuracy; (2) we propose to simply extract circuit models at several sample frequencies and then apply DLSCF to obtain approximation polynomials to calculate parasitics at other frequencies. Advanced discrete curve fitting techniques are employed to solve numerical problems while guaranteeing the convergence. The comparison with state-of-the-art extraction tools, such as Sonnet®, shows that approximation polynomials of a reasonable order provide sufficient accuracy within $0.1\%$ error; (3) we derive and apply WIFRIM algorithm to calculate moments

of arbitrary orders with the consideration of model variation at different frequencies. The obtained moments therefore can be used to evaluate the interconnect behavior within a wide frequency range.

We have clearly supported the motivation for the new methodology. The discussion proceeds (Section II) with describing the details of the WIFRIM algorithm. Section III presents the new extraction algorithm and DLSCF techniques. Meaningful experimental results (Section IV) and a summary of our work (Section V) conclude this paper.

## II. Wide Frequency Range Moment Matching

In the introduction section, we have presented the outline of our interconnect analysis methodology:

- Extract parasitics at selected sample frequencies;
- Generate approximation polynomials by using DLSCF;
- Apply the WIFRIM algorithm to calculate moments.

Since the ultimate goal of interconnect analysis is to guarantee that it satisfies the design specifications by examining its time and frequency characteristics, the WIFRIM algorithm will be introduced before presenting other techniques.

Parasitic matrices for a given interconnect system are essentially frequency dependent. Assume for now that those parasitic matrices, including the conductance matrix $\mathcal{G}$, the capacitance matrix $\mathcal{C}$, the reluctance matrix $\mathcal{K}$, and the inductance matrix $\mathcal{L}$, can be represented in power series of $s$

$$\mathcal{G}(s)=\sum_{i=0}^{M}\mathcal{G}_i s^i,\quad \mathcal{C}(s)=\sum_{i=0}^{M}\mathcal{C}_i s^i,$$
$$\mathcal{K}(s)=\sum_{i=0}^{M}\mathcal{K}_i s^i,\quad \mathcal{L}(s)=\sum_{i=0}^{M}\mathcal{L}_i s^i. \tag{1}$$

Parasitic matrices at a particular frequency $\omega = 2\pi f$ can be produced by simply substituting $s = j\omega$ into (1). How to express thos parasitic matrices in approximation polynomials will be discussed in Section III.

### A. WIFRIM for Extracting Inductance

In the case that inductances are extracted for a general interconnect system, the equivalent circuit can be represented in terms of MNA equations in the Laplace domain as

$$\left(\begin{bmatrix} G & A_l^T \\ -A_l & 0 \end{bmatrix} + s\begin{bmatrix} C & 0 \\ 0 & \mathcal{L} \end{bmatrix}\right)\begin{bmatrix} V_n \\ I_l \end{bmatrix} = \begin{bmatrix} -A_i I_s \\ 0 \end{bmatrix}, \tag{2}$$

where $G$ and $C$ contain the stampings of conductances and capacitances respectively. $A_l$ and $A_i$ represent the adjacency matrices of inductances and independent current sources. $V_n$ and $I_l$ denote node voltages and inductance current variables. $\mathcal{L}$ is the dense inductance matrix containing self and mutual inductance information.

Based on the assumption that parasitic matrices $\mathcal{G}$, $\mathcal{C}$, and $\mathcal{L}$ can be expressed in power series of $s$ as shown in (1), it is obvious that the stamping conductance matrix $G$ and the stamping capacitance matrix $C$ in (2) can also be represented in power series of $s$

$$G=\sum_{k=0}^{M}G_k s^k,\ C=\sum_{k=0}^{M}C_k s^k, \tag{3}$$

where

$$G_k = A_g^T\mathcal{G}_k A_g,\ C_k = A_c^T\mathcal{C}_k A_c. \tag{4}$$

$A_g$ and $A_c$ denote the adjacency matrices of conductances and capacitances, respectively.

To illustrate the idea of the new moment matching method, we expand both sides of (2) into Taylor series around zero frequency $s = 0$ and by using (3), we get

$$\left(\begin{bmatrix} G_0 & A_l^T \\ -A_l & 0 \end{bmatrix} + \sum_{k=1}^{M}\begin{bmatrix} G_k & 0 \\ 0 & 0 \end{bmatrix}s^k + \sum_{k=0}^{M}\begin{bmatrix} C_k & 0 \\ 0 & \mathcal{L}_k \end{bmatrix}s^{k+1}\right)$$
$$\sum_{k=0}^{\infty}\begin{bmatrix} m_k^v \\ m_k^i \end{bmatrix}s^k = -A_i\sum_{k=0}^{\infty}\begin{bmatrix} u_k \\ 0 \end{bmatrix}s^k,$$

where $m_k^v$, $m_k^i$ and $u_k$, the coefficients of the $k^{th}$ term in the Taylor series, are known as the $k^{th}$ moment of $V_n$, $I_l$ and $I_s$ respectively. The basic idea of moment matching is to represent the finite unknown moments of the left hand side of the above equation in terms of the known moments of the right hand side, and use the obtained moments to approximate the whole frequency domain spectrum of a circuit.

Rearranging the terms on the left hand side in the above equation gives:

$$\sum_{k=0}^{M+1}P_k^L s^k \cdot \sum_{k=0}^{\infty}\begin{bmatrix} m_k^v \\ m_k^i \end{bmatrix}s^k = -A_i\sum_{k=0}^{\infty}\begin{bmatrix} u_k \\ 0 \end{bmatrix}s^k, \tag{5}$$

where

$$P_k^L = \begin{cases} \begin{bmatrix} G_0 & A_l^T \\ -A_l & 0 \end{bmatrix} & ,k=0, \\ \begin{bmatrix} G_k + C_{k-1} & 0 \\ 0 & \mathcal{L}_{k-1} \end{bmatrix} & ,1\le k\le M, \\ \begin{bmatrix} C_M & 0 \\ 0 & \mathcal{L}_M \end{bmatrix} & ,k=M+1. \end{cases} \tag{6}$$

By matching the coefficients of $s$ in (5), the WIFRIM algorithm can be expressed by the following equations

- when $k = 0$,

$$P_0^L\cdot\begin{bmatrix} m_0^v \\ m_0^i \end{bmatrix} = \begin{bmatrix} -A_i u_0 \\ 0 \end{bmatrix}, \tag{7}$$

- when $1 \le k \le M+1$,

$$P_0^L\cdot\begin{bmatrix} m_k^v \\ m_k^i \end{bmatrix} = \begin{bmatrix} -A_i u_k \\ 0 \end{bmatrix} - \sum_{j=0}^{k-1}P_{k-j}^L\begin{bmatrix} m_j^v \\ m_j^i \end{bmatrix}, \tag{8}$$

- when $k > M+1$,

$$P_0^L\cdot\begin{bmatrix} m_k^v \\ m_k^i \end{bmatrix} = \begin{bmatrix} -A_i u_k \\ 0 \end{bmatrix} - \sum_{j=1}^{M+1}P_j^L\begin{bmatrix} m_{k-j}^v \\ m_{k-j}^i \end{bmatrix}. \tag{9}$$

It can be seen that WIFRIM only needs to decompose the sparse matrix $P_0^L = \begin{bmatrix} G_0 & A_l^T \\ -A_l & 0 \end{bmatrix}$ once and the high-order moments only depend on lower-order ones, and hence can be generated iteratively.

### B. WIFRIM for Extracting Reluctance

Due to its sparsity and stability, reluctance has been adopted as an efficient method to model inductance effects. The MNA equations for reluctance is similar to (2) and the only change is that $\mathcal{L}$ is replaced by $\mathcal{K}^{-1}$

$$\left(\begin{bmatrix} G & A_l^T \\ -A_l & 0 \end{bmatrix} + s\begin{bmatrix} C & 0 \\ 0 & \mathcal{K}^{-1} \end{bmatrix}\right)\begin{bmatrix} V_n \\ I_l \end{bmatrix} = \begin{bmatrix} -A_i I_s \\ 0 \end{bmatrix}. \tag{10}$$

The reluctance matrix $\mathcal{K}$ is much sparse compared with the inductance matrix $\mathcal{L}$ and can be obtained by a new extraction algorithm given in the next section.

To take the full advantage of the sparsity of $\mathcal{K}$, we multiply both sides of (10) by $\left[\begin{smallmatrix} I & 0 \\ 0 & \mathcal{K} \end{smallmatrix}\right]$ and get

$$\Big(\left[\begin{smallmatrix} G & A_l^T \\ -\mathcal{K}A_l & 0 \end{smallmatrix}\right] + s\left[\begin{smallmatrix} C & 0 \\ 0 & I \end{smallmatrix}\right]\Big)\left[\begin{smallmatrix} V_n \\ I_l \end{smallmatrix}\right] = \left[\begin{smallmatrix} -A_i I_s \\ 0 \end{smallmatrix}\right]. \quad (11)$$

Similarly, by substituting (3) into (11) and expanding the input vector and the MNA variable vector into Taylor series about $s = 0$, we get

$$\Big(\left[\begin{smallmatrix} G_0 & A_l^T \\ -\mathcal{K}_0 A_l & 0 \end{smallmatrix}\right] + \sum_{k=1}^{M}\left[\begin{smallmatrix} G_k & 0 \\ -\mathcal{K}_k A_l & 0 \end{smallmatrix}\right] s^k + s\left[\begin{smallmatrix} C_0 & 0 \\ 0 & I \end{smallmatrix}\right]$$
$$+\sum_{k=1}^{M}\left[\begin{smallmatrix} C_k & 0 \\ 0 & 0 \end{smallmatrix}\right] s^{k+1}\Big)\sum_{k=0}^{\infty}\left[\begin{smallmatrix} m_k^v \\ m_k^i \end{smallmatrix}\right] s^k = -A_i\sum_{k=0}^{\infty}\left[\begin{smallmatrix} u_k \\ 0 \end{smallmatrix}\right] s^k. \quad (12)$$

Rearranging the terms on the left hand side in (12) gives

$$\sum_{k=0}^{M+1} P_k^K s^i \cdot \sum_{k=0}^{\infty}\left[\begin{smallmatrix} m_k^v \\ m_k^i \end{smallmatrix}\right] s^k = -A_i\sum_{k=0}^{\infty}\left[\begin{smallmatrix} u_k \\ 0 \end{smallmatrix}\right] s^k, \quad (13)$$

where

$$P_k^K = \begin{cases} \left[\begin{smallmatrix} G_0 & A_l^T \\ -\mathcal{K}_0 A_l & 0 \end{smallmatrix}\right] & , k = 0, \\ \left[\begin{smallmatrix} G_1 + C_0 & 0 \\ -\mathcal{K}_1 A_l & I \end{smallmatrix}\right] & , k = 1, \\ \left[\begin{smallmatrix} G_k + C_{k-1} & 0 \\ -\mathcal{K}_k A_l & 0 \end{smallmatrix}\right] & , 2 \le k \le M, \\ \left[\begin{smallmatrix} C_M & 0 \\ 0 & 0 \end{smallmatrix}\right] & , k = M+1. \end{cases} \quad (14)$$

By matching the coefficients of $s$ in (13), the WIFRIM algorithm of reluctance can be obtained as follows

- when $k = 0$,

$$P_0^K \cdot \left[\begin{smallmatrix} m_0^v \\ m_0^i \end{smallmatrix}\right] = \left[\begin{smallmatrix} -A_i u_0 \\ 0 \end{smallmatrix}\right], \quad (15)$$

- when $1 \le k \le M+1$,

$$P_0^K \cdot \left[\begin{smallmatrix} m_k^v \\ m_k^i \end{smallmatrix}\right] = \left[\begin{smallmatrix} -A_i u_k \\ 0 \end{smallmatrix}\right] - \sum_{j=0}^{k-1} P_{k-j}^K \left[\begin{smallmatrix} m_j^v \\ m_j^i \end{smallmatrix}\right], \quad (16)$$

- when $k > M+1$,

$$P_0^K \cdot \left[\begin{smallmatrix} m_k^v \\ m_k^i \end{smallmatrix}\right] = \left[\begin{smallmatrix} -A_i u_k \\ 0 \end{smallmatrix}\right] - \sum_{j=1}^{M+1} P_j^K \left[\begin{smallmatrix} m_{k-j}^v \\ m_{k-j}^i \end{smallmatrix}\right]. \quad (17)$$

It is clear that the WIFRIM algorithm for reluctance also only requires to decompose the sparse matrix $P_0^K = \left[\begin{smallmatrix} G_0 & A_l^T \\ -\mathcal{K}_0 A_l & 0 \end{smallmatrix}\right]$ once and the moment calculation process is the same as that of inductance.

The difference between WIFRIM and ordinary moment matching algorithm is that WIFRIM expresses parasitic matrices in power series so as to consider the parasitic variations at different frequencies. For ordinary moment matching methods, the computational time of simulating $N$ sample frequencies requires $N$ matrix decompositions, $N$ matrix-vector multiplications, and $N$ vector additions. While WIFRIM only involves one sparse matrix decomposition, at most $M+1$ matrix-vector multiplications and $M+2$ vector additions, where $M$ is the highest order of approximation polynomials. Since the dominant computational time is in matrix decomposition and we will show in the experiment section that low-order approximation polynomials are sufficient to satisfy the desired accuracy, WIFRIM is an efficient algorithm to generate moments while taking frequency dependent effects into account.

## III. Efficient Extraction and Approximation

We have presented in the previous section that WIFRIM can generate moments under the assumption that parasitic matrices can be represented in power series of $s$. In this section, we propose an improved reluctance extraction algorithm to extract parasitics at some sample frequencies. Those discrete sample values are utilized by DLSCF to construct approximation polynomials, which can be employed by WIFRIM to calculate moments of arbitrary orders.

### A. Reluctance Extraction with Parallel Filament Reduction

Generally, the first stage of interconnect analysis is to model the interconnect system as a lumped RLKC circuit. To capture skin and proximity effects, conductor segments are logarithmically discretized into filaments according to their geometries, skin depths, and the current extraction frequency. For an interconnect system containing $N_c$ conductor segments, assuming that the $k^{th}$ segment is meshed into $n_k$ filaments, the total number of filaments within the system will be

$$N_f = \sum_{k=1}^{N_c} n_k. \quad (18)$$

Obviously $N_f$ is much larger than $N_c$, especially at high frequencies.

Each filament is thin enough such that the current can be approximated uniformly distributed inside the filament. Then the current $\hat{I}_i$ and the voltage drop $\hat{V}_i$[1] on filament $i$ can be given by

$$\hat{R}_{ii}\hat{I}_i + j\omega\sum_{j=1}^{N_f}\hat{L}_{ij}\hat{I}_j \quad = \quad \hat{V}_i. \quad (19)$$

The filament DC resistance $\hat{R}_{ii}$ can be calculated from

$$\hat{R}_{ii} \quad = \quad \frac{\hat{l}_i}{\sigma\hat{a}_i}, \quad (20)$$

where $\sigma$ is the conductivity of the conductor. $\hat{l}_i$ and $\hat{a}_i$ are its length and cross-section area respectively. The mutual partial inductance between filaments, $\hat{L}_{ij}$, has the following formulation

$$\hat{L}_{ij} \quad = \quad \frac{\mu_0}{4\pi\hat{a}_i\hat{a}_j}\Big[\int_{\hat{a}_i}\int_{\hat{a}_j}\int_{\hat{l}_i}\int_{\hat{l}_j}\frac{d\vec{l}_i d\vec{l}_j}{\|\vec{r}_i - \vec{r}_j\|}d\hat{a}_i d\hat{a}_j\Big]. \quad (21)$$

Because the current inside each filament is assumed to be uniformly distributed, (21) can be accurately integrated by Grover's or Hoer's Formulae [10], [11].

(19) can be written in a matrix form:

$$(\hat{R} + j\omega\hat{L})\cdot\hat{I} \quad = \quad \hat{V}, \quad (22)$$

where $\hat{R}$ is an $N_f \times N_f$ diagonal matrix containing filament DC resistances and $\hat{L}$ is the $N_f \times N_f$ partial inductance matrix. They are known by applying (20) and (21) after conductor segments are discretized into filaments. $\hat{I}$ and $\hat{V}$ are filament current and potential drop vectors respectively. The filament impedance matrix $\hat{Z}(\omega) \in \mathcal{C}^{N_f \times N_f}$ is given by

$$\hat{Z}(\omega) \quad = \quad \hat{R} + j\omega\hat{L}. \quad (23)$$

Physically, a bundle of filaments within the same conductor segment can be treated as parallel branches. Merging parallel

[1] A little hat ˆ is used to distinguish the symbols for filaments from those for conductor segments

elements can be facilitated by using admittance instead of impedance. Let $\hat{Y}(\omega) \in \mathcal{C}^{N_f \times N_f}$ be the filament admittance matrix. Then,

$$\hat{Y}(\omega) \cdot \hat{V} \quad = \quad \hat{I}. \tag{24}$$

From (24), it is clear that the $i^{th}$ column of $\hat{Y}(\omega)$ can be obtained by setting the $i^{th}$ element in $\hat{V}$ to one and the rest elements in $\hat{V}$ to zero, and then solving (22) to obtain the current distribution vector $\hat{I}$, which is equal to the $i^{th}$ column of $\hat{Y}(\omega)$. Instead of directly inverting $\hat{Z}(\omega)$, $\hat{Y}(\omega)$, therefore, can be constructed by solving (22) $N_f$ times.

However, this mathematical treatment is physically impossible, since we cannot set the voltage drop along one filament to one while keeping all the voltage drops of its parallel filaments zero. Consequently, in order to calculate the current distribution within conductor $k$, we need to simultaneously set voltages along all its $n_k$ filaments to one.

The physical meaning of the obtained current distribution is that: the summation of all the filament currents within the $k^{th}$ conductor is its admittance, while the summation of filament currents within the $l^{th}$ conductor is its coupling with conductor $k$. These values are stamped into the conductor admittance matrix $Y(\omega) \in \mathcal{C}^{N_c \times N_c}$, which is obtained directly by solving (22) $N_c$ times.

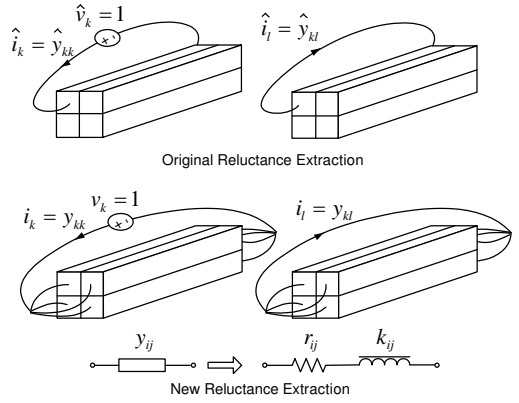


Fig. 1. Physical Explanation of Original Reluctance Extraction Method and Our Improved Reluctance Extraction Method.

Elements within $Y(\omega)$ represent admittances which are composed of two parts:

$$y_{ij} = g_{ij} + jx_{ij}, \tag{25}$$

where $g_{ij}$ is the conductance and $x_{ij}$ is the susceptance. The most straight forward way is to synthesize $y_{ij}$ as serially connected resistance and reluctance. The equivalent resistance $r_{ij}$ and reluctance $k_{ij}$ will be

$$\begin{aligned} r_{ij} &= \frac{g_{ij}}{g_{ij}^2 + x_{ij}^2}, \\ k_{ij} &= -\frac{\omega(g_{ij}^2 + x_{ij}^2)}{x_{ij}}. \end{aligned} \tag{26}$$

Those synthesized values are stamped into the conductance matrix $\mathcal{G}$ and the reluctance matrices $\mathcal{K}$ to construct equivalent circuit models. In this paper, we apply FastCap to obtain the capacitance matrix $\mathcal{C}$. The detailed extraction algorithm is summerized in Table I.

There are two reasons that this new extraction method is more efficient than the original one. First, we only need to solve (22) $N_c$ times instead of $N_f$ times. Second, the dimension of the obtained conductor admittance matrix $Y(\omega)$ is much smaller than that of the filament admittance matrix $\hat{Y}(\omega)$ and hence results in more compact models.

TABLE I

EXTRACT RELUCTANCE WITH SIMULTANEOUS FILAMENTS REDUCTION.

**INPUT:** An interconnect system including $n$ conductor segments; Extraction frequency $f$ of interest.
**OUTPUT:** Parasitic matrices $\mathcal{G}$ and $\mathcal{K}$.
**BEGIN**
For each conductor $i$ in the interconnect system, do the following:
- **a.** Search its neighboring conductors $\Upsilon_i$ by adopting one window selection algorithm, such as [3], [4];
- **b.** Discretize all the conductors within $\Upsilon_i$ into filaments;
- **c.** Calculate the filament impedance matrix $Z_f^i$;
- **d.** Set entries in the voltage vector $V_f^i$ corresponding to filaments in conductor $i$ to one while others to zero.
- **e.** Obtain the filament current distribution $I_f^i$ by solving (22);
- **f.** The self admittance of conductor $i$ equals to the sum of filament currents within conductor $i$; the summation of filament currents in conductor $j$ is the coupling between conductor $i$ and $j$;
- **g.** Synthesize admittance into serial resistance and reluctance by applying (26).
- **f.** Stamp those values into parasitic matrices $\mathcal{G}$ and $\mathcal{K}$ respectively.

**END**

### *B. Parasitics Approximation by DLSCF*

Albeit the efficiency of our improved extraction algorithm, it is still impractical to construct interconnect models at every frequency of interest. Therefore, we propose to apply DLSCF to obtain approximation polynomials, which can be used as close-form formulae to calculate parasitics at arbitrary frequencies. Furthermore, these approximation polynomials are essential to our WIFRIM algorithm discussed previously.

Assume we have extracted parasitics for a given interconnect system at $N$ sample frequencies: $\omega_1$, $\omega_2$, $\cdots$, $\omega_N$. A set of parasitic values $p_1$, $p_2$, $\cdots$, $p_N$ at those sample frequencies is to be approximated by the polynomial of degree $M$

$$y(\omega) \quad = \quad \sum_{k=0}^{M} a_k \phi_k(\omega), \tag{27}$$

where $\phi_0(\omega)$, $\phi_1(\omega)$, $\cdots$, $\phi_M(\omega)$ are appropriately chosen polynomial base functions.

The residual $r(\omega_i)$ at a particular frequency $\omega_i$ of a certain approximation is defined as

$$r(\omega_i) = p_i - y(\omega_i) = p_i - \sum_{k=0}^{M} a_k \phi_k(\omega_i). \tag{28}$$

The best approximation in the least square sense is the one for which the sum of the residuals squared over the whole domain is least

$$\sum_{i=1}^{N} [p_i - \sum_{k=0}^{M} a_k \phi_k(\omega_i)]^2 \quad = \quad minimum. \tag{29}$$

By imposing the condition in (29), it is obtained that

$$\frac{\partial}{\partial a_j}\Big\{\sum_{i=1}^{N}[p_i-\sum_{k=0}^{M}a_k\phi_k(\omega_i)]^2\Big\} = \sum_{i=1}^{N}\phi_j(\omega_i)[p_i-\sum_{k=0}^{M}a_k\phi_k(\omega_i)] = 0, \quad (30)$$

where $j=0,1,\cdots,M$. Rearranging the terms and interchanging the summation over $i$ and $k$ gives

$$\sum_{k=0}^{M}a_k\sum_{i=1}^{N}\phi_j(\omega_i)\phi_k(\omega_i)=\sum_{i=1}^{N}p_i\phi_j(\omega_i), \quad (31)$$

and hence leads to $M+1$ simultaneous linear equations, called normal equations. Unknown coefficients $a_k$ in (27) can be obtained by solving the normal equations in (31).

Although base function $\phi(x)$ in the approximation function $y(\omega)$ in (27) can be simply chosen as monomials, the formulated normal equations of the least square problem are usually ill-conditioned. Fortunately, the notorious numerical problem associate with solving a set of ill-conditioned simultaneous normal equations can be avoided by choosing $\phi(\omega)$ as mutually orthogonal polynomials.

A set of functions $g_1$, $g_2$, $\cdots$, $g_n$ is said to be mutually orthogonal if

$$\sum_{k=1}^{n}g_i(x_k)g_j(x_k) = \begin{cases} 0 & i\neq j \\ Q_i & i=j \end{cases}. \quad (32)$$

If one chooses the functions $\phi(\omega)$ as mutually orthogonal polynomials, substituting (32) into (31) gives

$$a_k\sum_{i=1}^{N}\phi_k^2(\omega_i)=\sum_{i=1}^{N}p_i\phi_k(\omega_i),\ k=0,1,\cdots,M. \quad (33)$$

Apparently, the use of orthogonal polynomials results in a set of decoupled equations, i.e. every normal equation yields one unknown. Hence, coefficients $a_k$ can be easily obtained as

$$a_k=\frac{\sum_{i=1}^{N}p_i\phi_k(\omega_i)}{\sum_{i=1}^{N}\phi_k^2(\omega_i)},\ k=0,1,\cdots,M. \quad (34)$$

In this paper, we adopt discrete Forsythe orthogonal polynomials [12] as our approximation base functions $\phi(\omega)$. The main advantage is that the orthogonal Forsythe polynomial approximation can be easily transferred into monomial approximation, and the coefficients of monomial approximation can be directly calculated without explicitly formulating those Forsythe polynomials. Therefore, parasitic matrices $\mathcal{G}$, $\mathcal{C}$, and $\mathcal{K}$ can be represented in power series of $\omega$

$$\mathcal{G}(\omega)=\sum_{i=0}^{M}\mathcal{G}_i\omega^i,\ \mathcal{C}(\omega)=\sum_{i=0}^{M}\mathcal{C}_i\omega^i,\ \mathcal{K}(\omega)=\sum_{i=0}^{M}\mathcal{K}_i\omega^i. \quad (35)$$

It is also straightforward to show that $\mathcal{G}$, $\mathcal{C}$, and $\mathcal{K}$ can be transferred into power series of $s=j\omega$

$$\mathcal{G}(s)=\sum_{i=0}^{M}\tilde{\mathcal{G}}_i s^i,\ \mathcal{C}(s)=\sum_{i=0}^{M}\tilde{\mathcal{C}}_i s^i,\ \mathcal{K}(s)=\sum_{i=0}^{M}\tilde{\mathcal{K}}_i s^i, \quad (36)$$

where for $k=0,1,2,\cdots$

$$\tilde{\mathcal{G}}_{4k}=\mathcal{G}_{4k},\quad \tilde{\mathcal{G}}_{4k+1}=-j\mathcal{G}_{4k+1},$$
$$\tilde{\mathcal{G}}_{4k+2}=-\mathcal{G}_{4k+2},\ \tilde{\mathcal{G}}_{4k+3}=j\mathcal{G}_{4k+3}. \quad (37)$$

$\tilde{\mathcal{C}}_i$'s and $\tilde{\mathcal{K}}_i$'s in (36) can be obtained similarly.

The significance of the obtained approximation polynomials in (35) and (36) is not limited to calculate parasitics at arbitrary frequencies. What makes it really attractive is that it can be incorporated into our WIFRIM algorithm to generate moments for the entire frequency range.

## IV. Experimental Results

In this section, we demonstrate the accuracy and efficiency of the proposed interconnect analysis method. All the simulations are run on an Intel® Pentium® 4 $2.4gHz$ system with $512\ MB$ memory.

To validate our frequency dependent interconnect analysis methodology, it is important to show that DLSCF can approximate real parasitic values accurately. For comparison purpose, we use one of the state-of-the-art full-wave simulation tools, Sonnet®, to capture parasitic variations with respect to frequency.

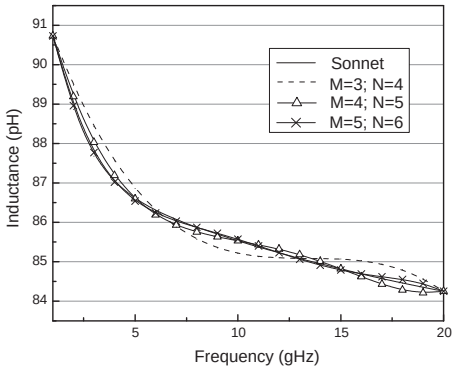


Fig. 2. DLSCF: Discrete Least Square Curve Fitting (M: the highest order of approximation polynomials; N: the number of sample points).

Figure 2 shows that low-order polynomials are sufficient to approximate real parasitics accurately. Extracting parasitics at six sample frequencies and using approximation polynomials of order five can provide accuracy within $0.1\%$ error as shown in Figure 3. The reason why DLSCF can achieve

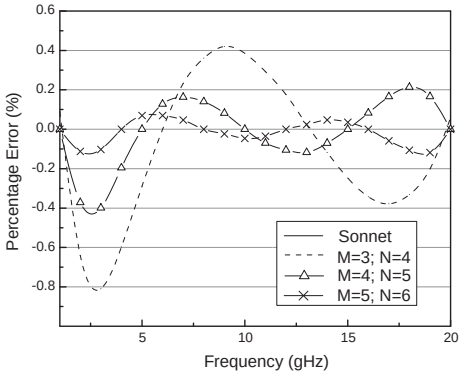


Fig. 3. DLSCF Percentage Error Curve.

high accuracy with low-order approximation polynomials is that, within $20gHz$, the variation of parasitics with respect

to frequency is basically monotone instead of fluctuant. For instance, the inductance decreases as frequency increases as shown in Figure 2. It is the simple shape of the parasitic curve that makes low-order approximation possible.

Next, we apply WIFRIM to analysis a Power/Ground network from a real IC design. Technologies in [7] are applied to obtain the final time and frequency domain information after the first twenty moments are calculated by WIFRIM. Although it is not necessary, sample frequencies are evenly distributed since one may interest in both the low and high frequency domain responses.

To perform time domain analysis, we apply a $1V$ step voltage input at one end of the power line and test the response at the other end. The waveforms with different sample points and approximation orders are shown in Figure 4. To test the

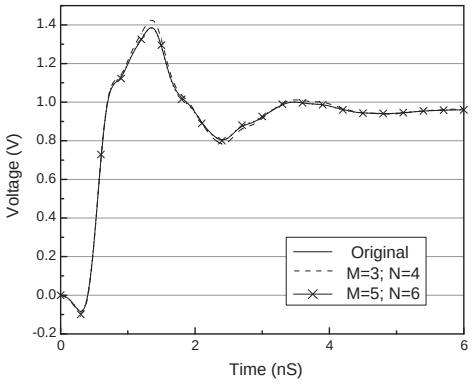


Fig. 4. Time Domain Analysis

frequency response, one current source is activated between the power line and the ground line. We first extract and analyze lumped circuit models at different frequencies and for each one, we adopt the corresponding frequency response at the specific frequency it is extracted and then combine them together to compare with WIFRIM.

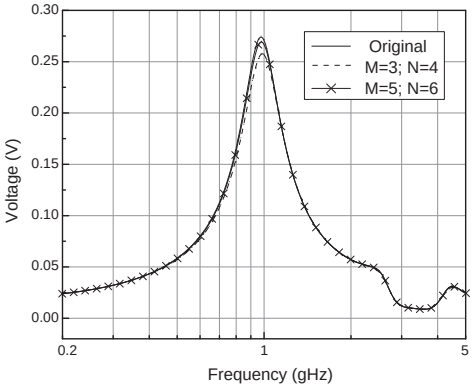


Fig. 5. Amplitude Spectrum in Frequency Domain

In current frequency range of interest, WIFRIM exhibits less than $1\%$ error by using approximation polynomials with order

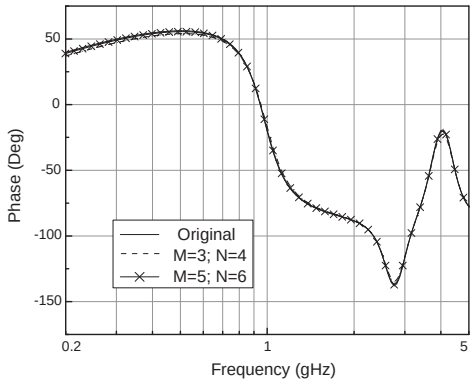


Fig. 6. Phase Spectrum in Frequency Domain

five. Especially for the frequency domain response, WIFRIM can produce almost indistinguishable results as shown in Figures 5 and 6.

## V. Conclusion

In this paper, we present a comprehensive frequency dependent interconnect analysis methodology. First, on-chip parasitics are extracted at sample frequencies by applying an improved reluctance extraction algorithm. Those extracted parasitics are utilized to construct approximation polynomials by employing discrete curve fitting techniques. The obtained approximation polynomials are used by the WIFRIM algorithm to calculate moments of arbitrary orders. Extensive experiments demonstrate that this new interconnect analysis strategy is an accurate and efficient one to provide the entire frequency domain information.

## References


[1] A. Devgan, H. Ji, and W. Dai, "How to efficiently capture on-chip inductance effects:introducing a new circuit element k," *ICCAD*, Nov 2000.

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

[3] T.-H. Chen, C. Luk, H. Kim, and C. C.-P. Chen, "Inductwise: Inductance-wise interconnect simulator and extractor," *ICCAD*, Nov 2002.

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

[5] A. E. Ruehli, "Inductance calculatioin in a complex integrated circuit environment," *IBM Journal of Research and Development*, Sep 1972.

[6] E. Rosa, "The self and mutual inductance of linear conductors," *In Bulletin of the National Bureau of Standars*, 1908.

[7] L. T. Pillage and R. A. Rohrer, "Asymptotic waveform evaluation for timing analysis," *TCAD*, vol. 9, pp. 352–366, Apr 1990.

[8] A. Odabasioglu, M. Celik, and L. T. Pileggi, "Prima: passive reduced-order interconnect macromodeling algorithm," *TCAD*, vol. 17, pp. 645–654, Aug 1998.

[9] Q. Yu, J. M. Wang, and E. S. Kuh, "Multipoint moment matching model for multiport distributed interconnect networks," *ICCAD*, Nov 1998.

[10] C. Hoer and C. Love, "Exact inductance equations for rectangular conductors with applications to more complicated geomotries," *J. Res. Nat. Bureau of Standards*, Apr 1965.

[11] F. W. Grover, *Inductance calculations: Working Formulas and Tables*. Dover Publications, 1946.

[12] G. E. Forsythe, "Generation and use of orthogonal polynomials for data fitting with a digital computer," *Journal Society Industrial and Applied Mathematics*, vol. 5, 1957.