Efficient Model Update for General Link-Insertion Networks*

Zhuo Feng
Department of ECE
Texas A&M University
College Station, TX 77843
fengzhuo@ece.tamu.edu

Peng Li
Department of ECE
Texas A&M University
College Station, TX 77843
pli@ece.tamu.edu

Jiang Hu
Department of ECE
Texas A&M University
College Station, TX 77843
jianghu@ece.tamu.edu

ABSTRACT

Link insertion has been proposed as a means of incremental design to improve performance robustness of linear passive networks. In clock network design, links can be inserted between subnetworks to reduce the variability of clock skews introduced by process and environmental fluctuations, thereby improving the network's immunity to PVT variations. Under these scenarios, it is desired to incrementally compute a reduced-order model for the updated network in order to efficiently evaluate the effectiveness of link insertions. In this paper, we present an efficient model update scheme for general link-insertion networks. By updating the Krylov projection subspace used in model order reduction, the proposed scheme can efficiently compute a reduced-order model for the network with inserted links. More generally, we extend the proposed approach to consider the merging of a (small) multiple-input linear network with a much larger network. We demonstrate the usage of the proposed technique for clock networks and general RLC circuits with an arbitrary number of link insertions as well as the more general case where the inserted links are in the form of a linear network.

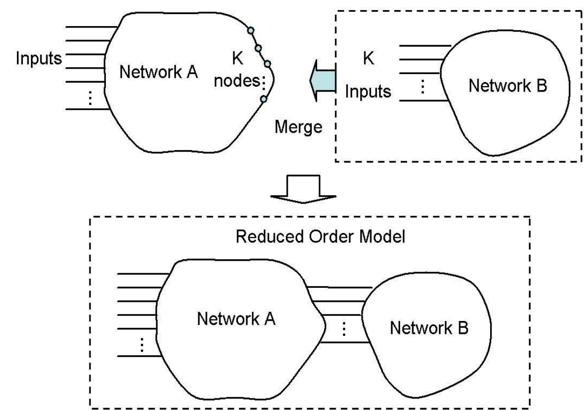

Figure 1: Model update for link-insertion networks.

1. INTRODUCTION

It has been demonstrated that non-tree clock networks can be instrumentally adopted to reduce the variability of clock skews in the presence of process variations [3]. In recent clock network synthesis works of [7, 8], links are inserted between different sub-trees/sub-networks to improve the immunity to process variations. Multiple link insertion sites are considered and the network is re-evaluated each time to determine the effectiveness of link insertion. Since many choices of link insertion need be examined in an optimization flow, it is critical to evaluate each updated clock network efficiently to reduce the cost of the overall optimization task. Although the reduced order models can be computed for linear passive networks by using any of existing model order reduction algorithms (such as [6, 1, 5]), recomputing a reduced-order model from scratch for a large network after each link insertion is computationally expensive in an optimization flow.

To reduce the computational cost for link-insertion networks, we consider the general situation shown in Fig. 1. In the figure, network A is a large linear passive network under the examination. It is assumed that K internal nodes of network A are selected and links are going to be inserted between them. Generally, we consider a K-input network, network B, is being inserted between the selected K nodes. In other words, the notation of link insertion is generalized to network merging. For our case, Network B is considered to be small, otherwise we assume that a reduced order model of network B is available. Given K selected nodes of network A and a model of network B, our goal is to efficiently compute a reduced order model for the merged network. To this end, we assume that the LU factor of the conductance matrix of network A is available and we will reuse the LU factor to obtain the reduced order model for the combined network. In the proposed approach, instead of generating a new updated model from scratch, we adopt the matrix inversion formula of Sherman and Morrison as well as the inversion formula of block matrices to efficiently compute the updated Krylov projection subspace [2]. In the past, the Sherman and Morrison inversion formula has been used in [10] for DC fault simulation and in [4] for considering driver resistance variations in interconnect analysis. The dominant computational cost of our approach is due to solving a linear system with M unknowns, where M is equal to the number of circuit nodes if multiple links are inserted or the size of the (small) linear network to be merged. For both cases, the complexity of the updated scheme is much less than computing the

*This work was support in part by SRC under contracts 2005-TJ-1298 and 2004-TJ-1205.

reduced model directly.

In the following sections, we first consider the case where only one link is inserted to a network, then extend the technique to the situation where multiple links are simultaneously inserted. Finally, we more generally consider the merging of a small network with a large one. We demonstrate the application of the proposed model update scheme on several clock and RLC networks under the context of general link insertion. The speedup for the rank one update of large RC networks is up to 140X in our experiment.

2. RANK-1 MODEL UPDATE

In this section, we present our approach for computing the updated reduced-order model after one link is inserted into the original network. We directly update the transfer function moments where the Matrix Inversion Formula [2] is used to quickly provide linear system solutions defined by the updated conductance matrix G_u. Each transfer function moment is computed efficiently by taking the advantage of the LU factorization of the original conductance matrix G. After computing these moments, an updated Krylov-subspace is obtained to generate the updated reduced model. To illustrate the proposed model update approach, let us consider a simple case where a resistor with a resistance value R_1 is inserted into a single-input linear network. For the original network, a reduced order model can be computed by projecting the full system matrices using the following Krylov-subspace [5]:

$$\{m_0, m_1, ..., m_q\} = \{G^{-1}b, G^{-1}CG^{-1}b, \dots, (G^{-1}C)^q G^{-1}b\}, \tag{1}$$

where $G, C \in \mathbb{R}^{n \times n}$ are conductance and susceptance matrices of the network, b is a vector linking the input to the network, n is the number of unknowns and $q+1$ is the reduced model order. Once a resistance R_1 is inserted between nodes i and j, the updated G matrix will become G_u and its inverse can be obtained using the Matrix Inversion Formula [2]

$$G_u^{-1} = (G + \frac{1}{R_1}\Delta_{ij})^{-1} = G^{-1} - \alpha G^{-1}\Delta_{ij}G^{-1}, \tag{2}$$

where $\alpha = \frac{1}{R_1 + c_{ij}^T G^{-1} c_{ij}}$ and $\Delta_{ij} = c_{ij}c_{ij}^T$. The connection vector

$$c_{ij} = [0, ..., +1, ..., -1, ...0]^T \tag{3}$$

is a vector with the i^{th} and j^{th} elements being positive and negative ones respectively, and all other elements being zeros. Let m_0^u, m_1^u, m_2^u,..., m_q^u be the updated moments after the insertion of R_1, then we have

$$m_0^u = G_u^{-1}b = G^{-1}b - \alpha G^{-1}\Delta_{ij}G^{-1}b. \tag{4}$$

If we denote the i^{th} element of vector v as r_i^v, it is not difficult to show the following

$$M\Delta_{ij}v = (r_i^v - r_j^v)Mc_{ij}, \tag{5}$$

where M is an arbitrary matrix with a proper dimension. Consequently we will have

$$G^{-1}\Delta_{ij}G^{-1}b = (r_i^{m_0} - r_j^{m_0})G^{-1}c_{ij}, \tag{6}$$

where $m_0 = G^{-1}b$. By defining a new vector $p_{ij} = G^{-1}c_{ij}$, (4) is modified to

$$m_0^u = m_0 - \alpha(r_i^{m_0} - r_j^{m_0})p_{ij}. \tag{7}$$

Following the similar procedure, we obtain the expression for the next moment as

$$\begin{aligned} m_1^u &= \left(G^{-1} - \alpha G^{-1}\Delta_{ij}G^{-1}\right)Cm_0^u \\ &= G^{-1}Cm_0^u - \alpha G^{-1}\Delta_{ij}G^{-1}Cm_0^u. \end{aligned} \tag{8}$$

Defining $\tilde{m}_1^u = G^{-1}Cm_0^u$, we have:

$$m_1^u = \tilde{m}_1^u - \alpha(r_i^{\tilde{m}_1^u} - r_j^{\tilde{m}_1^u})p_{ij}. \tag{9}$$

Similarly we can compute the moments of higher orders iteratively by using the result of the last step. For the l^{th} moment, it follows that:

$$m_l^u = \tilde{m}_l^u - \alpha(r_i^{\tilde{m}_l^u} - r_j^{\tilde{m}_l^u})p_{ij}, \tag{10}$$

where

$$\tilde{m}_l^u = G^{-1}Cm_{l-1}^u. \tag{11}$$

As observed, for each moment of a higher order, we only have to compute one new vector $\tilde{m}_l^u$ and linearly combine it with the vector p_{ij}. Since the LU factorization of the original conductance matrix G is already computed, the new vector $\tilde{m}_l^u$ can be efficiently computed. For a general N-port network, we can follow the similar procedure as:

$$\begin{aligned} M_1^u &= \left(G^{-1} - \alpha G^{-1}\Delta_{ij}G^{-1}\right)CM_0^u \\ &= G^{-1}CM_0^u - \alpha G^{-1}\Delta_{ij}G^{-1}CM_0^u; \end{aligned} \tag{12}$$

$$\begin{aligned} &\tilde{M}_1^u = G^{-1}C\tilde{M}_0^u \Rightarrow \\ &M_1^u = \tilde{M}_1^u - \alpha p_{ij}c_{ij}^T\tilde{M}_1^u; \\ &\vdots \\ &M_l^u = \tilde{M}_{l-1}^u - \alpha p_{ij}c_{ij}^T\tilde{M}_{l-1}^u, \end{aligned} \tag{13}$$

where $M_l^u \in \mathbb{R}^{n \times N}$ is the $(l+1)^{th}$ block moment. We make the following further observations:

- If there are extra capacitances introduced by the inserted link, we can simply replace matrix C with the updated matrix C_u in the above procedure while the rest of computation remains the same;
- In the above we have only shown how to reuse the existing LU factor of G matrix to efficiently compute the moment of the updated system explicitly at each step. It should be understood that orthogonalization steps can be incorporated to improve the numerical stability of the model update as in a standard Krylov-subspace based method;
- After obtaining the new Krylov-subspace we can use a projection-based algorithm like PRIMA [5] to generate the reduced model $\tilde{G}_r$, $\tilde{C}_r$, $\tilde{B}_r$ and $\tilde{L}_r$. It is not difficult to observe that for a network with a large number of unknowns, the computation cost of our approach is much less than the direct method in which the inversion of the updated conductance matrix has to be computed.

3. RANK-K MODEL UPDATE

To consider multiple link insertions, we need to extend the rank-1 update scheme of the previous section to a more general rank-k update scheme. Denote by $R_1, ..., R_k$ the resistances associated with these links, (i_1, j_1), (i_2, j_2), ..., (i_k, j_k) the corresponding pairs of nodes for each link insertion, , and $c_{i_1 j_1}$, $c_{i_2 j_2}, ..., c_{i_k j_k}$ the connection vectors defined as in (3), respectively. The updated conductance matrix can be written as

$$G_u = G + \sum_{t=1}^{k} \frac{1}{R_t} c_{i_t j_t} c_{i_t j_t}^T = G + S_c \Lambda_g S_c^T, \tag{14}$$

where matrices $S_c \in \mathbb{R}^{n \times k}$ and $\Lambda_g \in \mathbb{R}^{k \times k}$ are defined as

$$S_c = [c_{i_1 j_1}, c_{i_2 j_2}, \cdots, c_{i_k j_k}], \tag{15}$$

$$\Lambda_g = \begin{bmatrix} \frac{1}{R_1} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \frac{1}{R_k} \end{bmatrix}. \tag{16}$$

By the Matrix Inversion Lemma, we have the inversion of G_u as

$$(G_u)^{-1} = G^{-1} - G^{-1} S_c (\Lambda_g^{-1} + S_c^T G^{-1} S_c)^{-1} S_c^T G^{-1}. \tag{17}$$

If we define matrix $K \in \mathbb{R}^{k \times k}$ as

$$K = (\Lambda_g^{-1} + S_c^T G^{-1} S_c)^{-1}, \tag{18}$$

the updated first block moment M_0^u of an N-port network with input matrix B becomes

$$\begin{aligned} M_0^u &= (G^u)^{-1} B \\ &= G^{-1} B - G^{-1} S_c K S_c^T G^{-1} B \\ &= M_0 - G^{-1} S_c K S_c^T M_0. \end{aligned} \tag{19}$$

Set

$$p_{i_1 j_1} = G^{-1} c_{i_1 j_1},\ p_{i_2 j_2} = G^{-1} c_{i_2 j_2},\ ..., p_{i_k j_k} = G^{-1} c_{i_k j_k}, \tag{20}$$

and

$$P = [p_{i_1 j_1}, p_{i_2 j_2}, \cdots, p_{i_k j_k}]. \tag{21}$$

In a similar way, we can express the l^{th} updated block moment as

$$\begin{aligned} M_l^u &= (G_u)^{-1} C_u M_{l-1}^u \\ &= \left[G^{-1} - P K S_c^T G^{-1}\right] C_u M_{l-1}^u \\ &= \tilde{M}_l^u - P K S_c^T \tilde{M}_l^u, \end{aligned} \tag{22}$$

where

$$\tilde{M}_l^u = G^{-1} C_u M_{l-1}^u. \tag{23}$$

In terms of complexity, we need to solve k linear systems to obtain $p_{i_m j_m} (m = 1, 2, ..., k)$ in (20) rather than one linear solution as in the rank-one update. Additionally, a $k \times k$ matrix inversion is needed to compute matrix K in (18). Nevertheless when $k, N \ll n$, the computation of the updated moments takes much less CPU time than the direct method, where an $n \times n$ matrix factorization is needed. For many applications, k and N are always much smaller than n, thus our approach is very efficient.

The flow of the proposed rank-k update scheme is summarized in Algorithm 1. It should be noted again that the orthogonalization steps can be straightforwardly incorporated to improve the numerical stability.

Algorithm 1 Rank-k update of the Krylov-subspace

Input: k, link insertion sites $(i_m, j_m), m = 1, \cdots, k$, S_c, Λ_g, C_u, LU of G, M_0, q.
Output: The updated Krylov projection matrix $\mathbf{X_u}$.

```
Set P ← G^{-1} S_c; K ← (Λ_g^{-1} + S_c^T G^{-1} S_c)^{-1}; X_u, M_0^u ← M_0.
for p = 1 to q do
   M̃_p^u ←Solve G M̃_p^u = C_u M_{p-1}^u;
   for m = 1 to k do
      row m of T ← row i_m of M̃_p^u − row j_m of M̃_p^u;
   end for
   M_p^u ← M̃_p^u − PKT;
   X_u ← [X_u  M_p^u];
end for
Return the updated Krylov-subspace projection matrix: X_u.
```

4. MERGING TWO MODELS

In this section, we introduce a general model update scheme for merging a (large) network, say A, with another network, say B, as shown in Fig. 1. As mentioned before, we assume that either network B is small or a reduced order model of it is available. Nevertheless, in the following update scheme, we need not to distinguish these two cases.

We denote by $i_1, ..., i_k$ the k internal nodes of network A to which the ports of network B will be attached. We use conductance matrix G_r, susceptance matrix C_r, input and output matrices B_r, L_r, and unknown vector x_r, all with proper dimensions, to specify network B with k ports. Notice that by our convention, the inputs to the network B are k port voltages and the outputs are k port currents. Similarly, network A is described by matrices/vectors G, C, B, L, x with proper dimensions. The Modified Nodal Analysis (MNA) formulation for the merged network is given as

$$\begin{cases} (G + sC)x + E L_r^T x_r = Bu \\ (G_r + sC_r)\, x_r = B_r u' \end{cases} \tag{24}$$

where

$$u' = \begin{bmatrix} e_{i_1}^T x \\ \vdots \\ e_{i_k}^T x \end{bmatrix} = E^T x, \tag{25}$$

are k port voltages. Substituting (25) into (24), we have

$$\begin{bmatrix} G & E_1 \\ E_2 & G_r \end{bmatrix} \begin{bmatrix} x \\ x_r \end{bmatrix} + s \begin{bmatrix} C & 0 \\ 0 & C_r \end{bmatrix} \begin{bmatrix} x \\ x_r \end{bmatrix} = \begin{bmatrix} Bu \\ 0 \end{bmatrix}, \tag{26}$$

where

$$E_1 = E L_r^T, E_2 = -B_r E^T. \tag{27}$$

Applying the well-known inversion formula of block matrices it follows

$$\begin{bmatrix} G & E_1 \\ E_2 & G_r \end{bmatrix}^{-1} = \begin{bmatrix} \left(I + G^{-1} E_1 K E_2\right) G^{-1} & -G^{-1} E_1 K \\ -K E_2 G^{-1} & K \end{bmatrix}, \tag{28}$$

where $K = (G_r - E_2 G^{-1} E_1)^{-1}$ is the inverse of the Schur complement of G. We further define $T = \left(I + G^{-1} E_1 K E_2\right)$, $Q = -G^{-1} E_1 K$ and $L = -K E_2$, then it follows that the first moment of this system is given as

$$M_0^{mg} = \begin{bmatrix} G & E_1 \\ E_2 & G_r \end{bmatrix}^{-1} \begin{bmatrix} B \\ 0 \end{bmatrix} = \begin{bmatrix} T G^{-1} B \\ L G^{-1} B \end{bmatrix}. \tag{29}$$

The next moment can be written as

$$M_1^{mg} = \begin{bmatrix} T G^{-1} & Q \\ L G^{-1} & K \end{bmatrix} \begin{bmatrix} C & 0 \\ 0 & C_r \end{bmatrix} \begin{bmatrix} M_0^{mgu} \\ M_0^{mgl} \end{bmatrix}. \tag{30}$$

M_0^{mg} is partitioned into an upper part and a lower part. In the same way, we may iteratively compute the l^{th} higher order moments as

$$M_l^{mg} = \begin{bmatrix} TG^{-1}C & QC_r \\ LG^{-1}C & KC_r \end{bmatrix} \begin{bmatrix} M_{l-1}^{mgu} \\ M_{l-1}^{mgl} \end{bmatrix}. \quad (31)$$

We notice that in the above procedure, $G^{-1}CM_l^{mgu}$ can be efficiently computed using the LU of G while other matrix computations are performed directly since the dimensions of involved matrices or vectors are small. Thus, just as the rank-k update scheme, the reduced order model for the combined network can be obtained rather efficiently by avoiding factorizing the updated conductance matrix.

5. RESULTS

In this section, the results in frequency and time domain are shown to demonstrate the effectiveness of the proposed approach. We first show the accuracy of our model update scheme and then its significant speedup over the direct method.

5.1 An RC network

We consider a two-port RC network consisting of 768 circuit unknowns. A link is modelled as a π model that consists of a resistance and two grounded capacitances. For simplicity, all the inserted π models are identical. Two cases are considered: 20 and 40 link insertions. For both cases, we apply the proposed algorithm to construct an updated Krylov-subspace and then generate a reduced model of size 14 (number of states). Each model matches up to 7 moments of the transfer function. In Fig. 2 and Fig. 3, Y_{21} is compared based on four different models. Two full models are shown, one with link insertions and the other without. In these cases, the frequency characteristics of Y_{21} changes noticeably after link insertions. For the full model with link insertions, two reduced order models are considered. The first reduced order model, labelled as "Redu. Model (Old Proj.)", is built by using the projection matrix computed for the full model without links. Thus, the impact of links is not considered in the projection matrix. The second reduced order model, labelled as "Redu. Model (New Proj.)" is computed using the proposed update scheme where the impact of links is fully accounted for. As observed in both figures, the reduced model with the updated projection matrix produces almost indistinguishable results with the full model with link insertions, while the reduced model produced by using the original projection matrix can not capture the frequency response at all.

5.2 RC clock trees

In the second example, a part of a clock distribution network is considered. As shown in Fig. 4, there are 50 identical subtrees driven by 50 clock drivers. The total number of leaf clock sink nodes is 400. Totally 49 links (π models) are inserted between the leaf nodes of any pair of adjacent subtrees as shown in Fig. 4. In our experiment, we drive 49 subtrees using an identical voltage input (clock 1, 2GHz) while the 25^{th} subtree is driven by a skewed clock input (clock 2, 2GHz). To evaluate the impact of the link insertions, we examine the waveforms at two leaf nodes, node 1 and node 4, as shown in Fig. 4, and compare against the responses in the network with links.

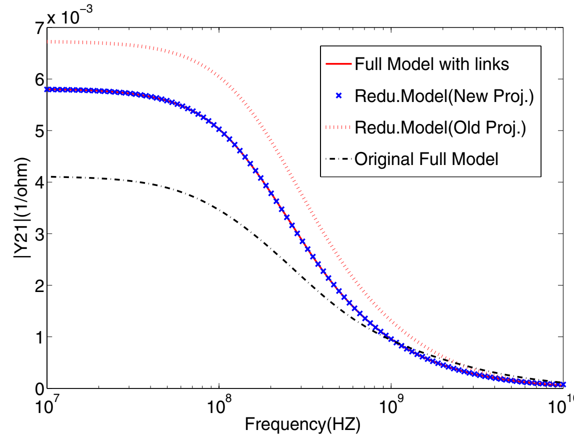

Figure 2: Y_{21} of the RC network after 20 link insertions.

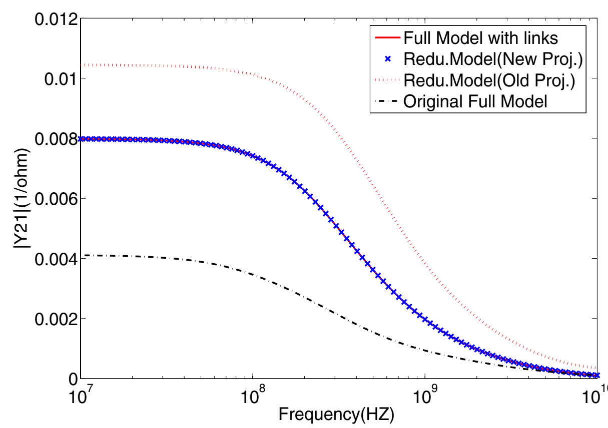

Figure 3: Y_{21} of the RC network after 40 link insertions.

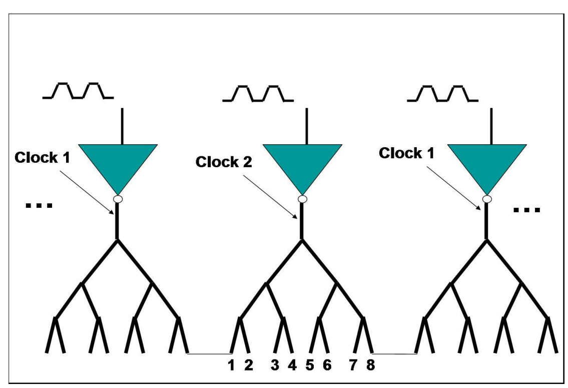

Figure 4: RC clock trees.

0-7695-2523-7/06 $20.00 © 2006 IEEE

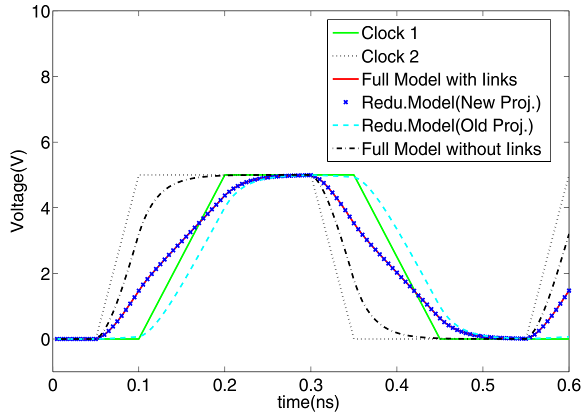

Figure 5: Voltage response at the insertion node 1 (R_{link}=0.01R_0).

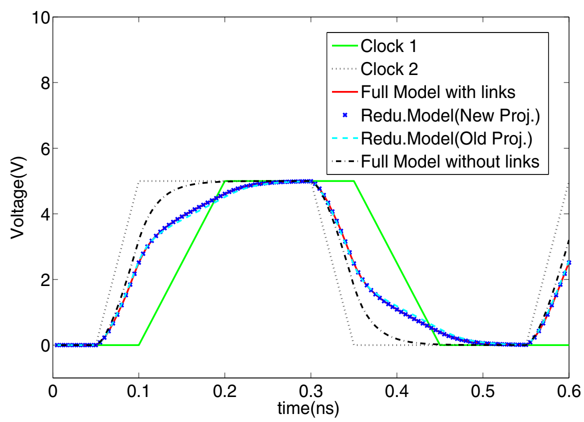

Figure 6: Voltage response at the insertion node 1 (R_{link}=10R_0).

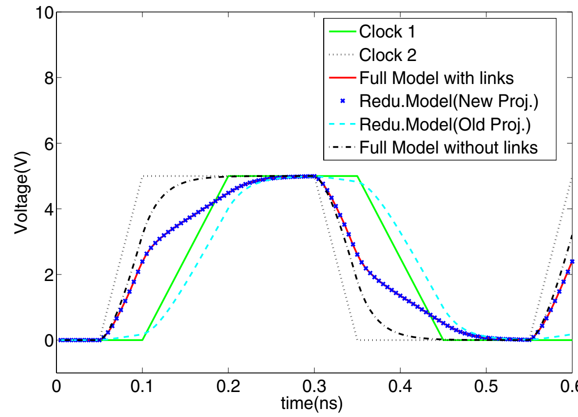

Figure 7: Voltage response at the neighboring node 4 (R_{link}=0.01R_0).

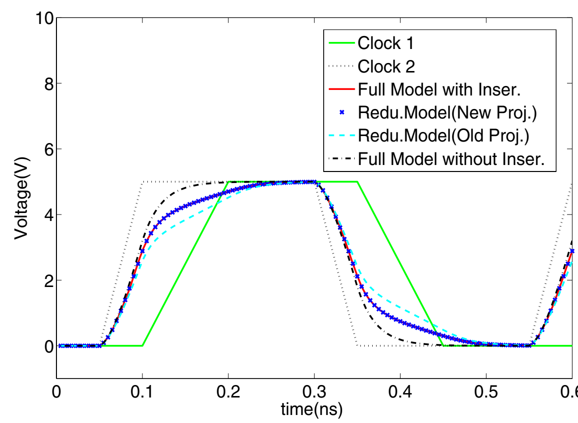

Figure 8: Voltage response at the neighboring node 4 (R_{link}=10R_0).

We select a reference resistance value R_0 and consider how the link resistance values of $0.01R_0$ and $10R_0$ impact the voltage responses at the two observation nodes. R_0 is selected to be the averaged resistance of one segment in the clock trees. Fig. 5 and Fig. 6 show that at the insertion node, the links greatly influence the waveforms of the leaf nodes unless $R_{link} > 10R_0$. It is noted that the link resistance values of $0.01R_0$ and R_0 have similar effects on the response at the insertion node according to our experiments.

Fig. 7 and Fig. 8 show the waveforms at the neighboring node 4 where the similar conclusions can be obtained. In these cases it is observed that our reduced model can accurately capture the circuit responses while the reduced model with the original projection matrix may lead to extremely large errors (Fig. 5 and Fig. 7).

5.3 Hybrid clock network

We also investigate a hybrid clock network with 12,816 unknowns. The network has a similar topology as described in [9]. It consists of a 4-level binary tree with 8 leaf-nodes and one driving node on the top of the tree. Each leaf-node

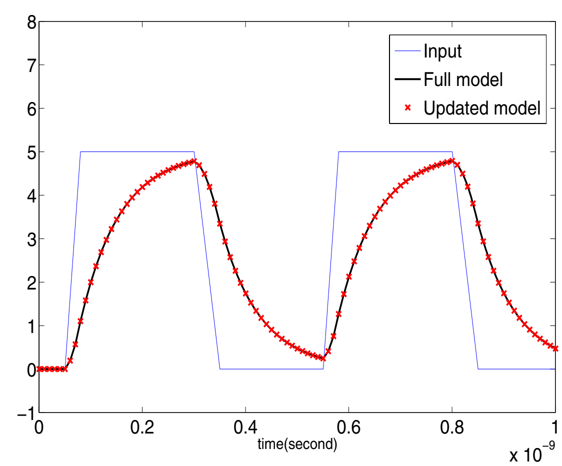

Figure 9: Node voltage of a hybrid clock network.

Proceedings of the 7th International Symposium on Quality Electronic Design (ISQED'06)
0-7695-2523-7/06 $20.00 © 2006 IEEE

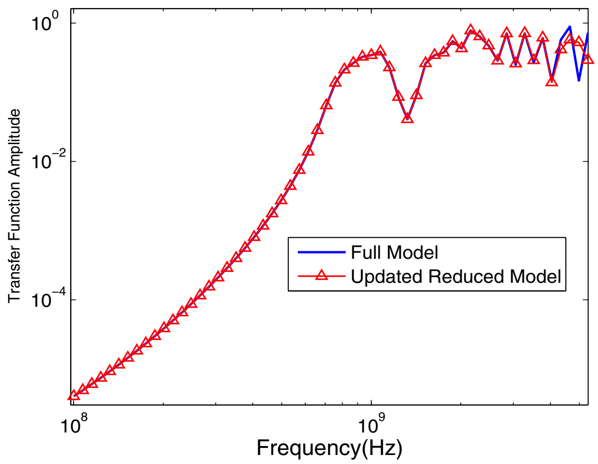

Figure 10: Model update for the two merged RLC networks.

is driving an RC mesh with a dimension of 40 by 40. Five links (π models) are inserted between several branches of the tree. We arbitrarily select one node of an RC mesh to examine the transient voltage waveform for a given clock input. As shown in Fig. 9, the results by the full model (with links) are compared with the updated reduced model (rank 5 update). The accuracy of the links update scheme is clear.

5.4 Two merged RLC networks

We demonstrate the model update for two merged RLC networks. The first network has 16 ports and $4,824$ unknowns and merged with a 2-port RLC line consisting of 31 unknowns. Two internal nodes are selected from the first network and the 2-port RLC line is inserted between these two nodes. In Fig. 10, the voltage transfer function from one port to one of the insertion nodes is plotted and the updated reduced order model is compared with the full model after the merging. The updated reduced model has a size of 160. As seen from the figure, the reduced order model is very accurate compared to the full model up to 50GHz.

5.5 Results on runtime

In this subsection, we demonstrate the high efficiency of the proposed techniques. All our experiments are conducted on a 3GHz Pentium-4 PC with 4GB memory running Linux operating system.

We first investigate the runtimes for generating reduced models for RC meshes. We compare the runtimes of the direct method with the proposed approach on the meshes with different sizes. Table 1 shows the LU factorization time (LU T.) and the total time (Tot. T.) of the direct method. The total times for computing the updated reduced models for rank 1, 10 and 20 as well as the speedups (Sp.) for rank one update are also shown in the table. The results indicate that the maximum speedup for the RC mesh with 40k unknowns can be as high as 142X for the rank one update. It is also observed that the computational cost is increased with the update rank.

In Table 2, we list the runtime results of generating reduced models for hybrid clock networks described in [9]. All the networks have the same binary tree topology (as the one in subsection 5.3) but have different sizes of RC meshes. A 15X speedup is achieved for the hybrid network with 12,816 unknowns in our experiment.

Table 1: Runtimes for RC meshes

	Direct Solve		Rank k Update			Sp.
Size	LU T.	Tot. T.	k=1	k=10	k=20	k=1
1k	0.03	0.04	<0.01	0.02	0.06	4X
2.5k	0.29	0.31	0.02	0.07	0.19	15X
6.4k	3.37	3.44	0.09	0.23	0.59	38X
10k	7.75	7.88	0.15	0.41	0.98	53X
40k	126.83	127.57	0.90	2.21	4.89	142X

Table 2: Runtimes for Hybrid clock networks

	Direct Solve		Rank 5 Update	Sp.
Size	LU T.	Tot.T.	Tot. T.	k=5
3,216	0.12s	0.14s	0.03s	5X
7,216	0.68s	0.74s	0.08s	9X
12,816	2.42s	2.55s	0.17s	15X

6. CONCLUSIONS

We have presented an efficient approach for model update of general link-insertion networks. Both the rank-1 and rank-k update schemes are shown in details. Additionally, we have extended our approach to a more general case where two networks are merged together. Compared with the direct method, the proposed technique can achieve significant speedups. Several numerical examples are presented with frequency and time domain simulations to demonstrate the saliency of this approach. The runtimes for some RC meshes and hybrid clock networks are also compared with the direct method to show the improved efficiency of the new approach.

7. REFERENCES

[1] P. Feldmann and R. Freund. Efficient linear circuit analysis by pade approximation via the lanczos process. *IEEE Trans. Computer-Aided Design*, 14(5):639–649, May 1995.

[2] G. H. Golub and C. F. V. Loan. *Matrix computations*. Johns Hopkins University Press, 3rd ed, 1996.

[3] N. Kurd, J. Barkatullah, R. Dizon, T. Fletcher, and P. Madland. A multigigahertz clocking scheme for the pentium 4 microprocessor. *IEEE J. of Solid-State Circuits*, 35(11):1647–1653, Nov. 2001.

[4] H. Levy, W. Scott, D. MacMillen, and J. White. A rank-one update method for efficient processing of interconnect parasitics in timing analysis. In *Proc. IEEE/ACM Design Automation Conf.*, June 2000.

[5] A. Odabasioglu, M. Celik, and L. Pileggi. Prima: passive reduced-order interconnect macromodeling algorithm. *IEEE Trans. Computer-Aided Design*, 17(8):645–654, Aug. 1998.

[6] L. Pillage and R. Rohrer. Asymptotic waveform evaluation for timing analysis. *IEEE Trans. Computer-Aided Design*, 9(4):352–366, Apr. 1990.

[7] A. Rajaram, J. Hu, and R. Mahapatra. Reducing clock skew variability via cross links. In *Proc. IEEE/ACM Design Automation Conf.*, pages 18–23, June 2004.

[8] A. Rajaram, D. Pan, and J. Hu. Improved algorithms for link based non-tree clock network for skew variability reduction. In *Proc. ACM Intl. Symp. on Physical Design*, pages 55–62, 2005.

[9] P. Restle, A. Ruehli, and S. Walker. Multi-ghz interconnect effects in microprocessors. In *Proc. ACM Intl. Symp. on Physical Design*, pages 93–97. April, June 2001.

[10] M. Tian and R. Shi. Nonlinear analog *DC* fault simulation by one-step relaxation. In *IEEE VLSI Test Symposium*, pages 126–131, Apr. 1998.