

大规模 RC 网络的优化消去模型降阶算法

程 康¹⁾, 叶佐昌²⁾, 喻文健^{1)*}

¹⁾(清华大学计算机科学与技术系 北京 100084)

²⁾(清华大学微电子学研究所 北京 100084)

(yu-wj@tsinghua.edu.cn)

摘 要: 针对电路后仿中大规模、多端口 RC 矩阵在降阶时非零元会显著增加, 导致降阶系统处理时间大幅上升的问题, 提出了寻找最优剩余节点数, 使降阶后的网络求解速度达到最优的算法。首先采用统计建模的方法得到 RC 网络求解时间与节点数、非零元数的近似关系; 然后在对实际电路进行模型降阶时用节点消去法快速得到剩余节点数与非零元的关系, 从中选出最优的剩余网络节点数并根据它实施模型降阶。实验结果表明, 采用该策略得到的降阶网络不造成精度损失, 且使电路求解速度比已有方法快 2~5 倍。

关键词: 电路后仿; 模型降阶; 数据拟合

中图法分类号: TN391.9

Model Order Reduction for Large Scale RC Networks Based on Optimal Elimination

Cheng Kang¹⁾, Ye Zuochang²⁾, and Yu Wenjian^{1)*}

¹⁾(Department of Computer Science and Technology, Tsinghua University, Beijing 100084)

²⁾(Institute of Microelectronics, Tsinghua University, Beijing 100084)

Abstract: Model order reduction (MOR) has been widely used in modeling and simulation of integrated circuits. MOR for large scale RC networks resulting from post-layout circuit analysis is still a difficult problem. Elimination based MOR methods has attracted focus. However, in practice, the number of elements in a network may not necessarily reduce during the elimination due to the degradation of sparsity. Given elimination order, in theory there is an optimal elimination, with which the resulting network has minimal elements. In this paper, we propose to find this optimal elimination based on symbolic analysis and statistical analysis. Experimental results show that with such approach the solution time of reduced network can be reduced by a factor of 2~5.

Key words: post simulation; model order reduction; data fitting

随着集成电路复杂度以及电路规模的提高, 电路仿真验证遇到极大挑战。即使对于中等规模的电路, 电路后仿, 也就是在寄生参数提取之后的电路仿真, 需要求解的电路节点数也动辄会达到百万以上的量级。而对典型的混合信号电路进行整体后仿, 规模往往超过千万甚至上亿节点, 这对仿真器而言是

个极大的挑战。

因此, 人们希望通过模型降阶来减小电路的规模, 同时又希望由此带来对精度的牺牲尽量小。模型降阶在数字电路设计中已经被广泛采用, 标准的模型降阶方法, 如 PRIMA^[1], Poor Man's TBR^[2] 等, 可以将几千甚至上万节点的 RC 网络降阶到只有几

收稿日期: 2011-12-22; 修回日期: 2012-05-29。基金项目: 国家自然科学基金(61106031, 61076034); 清华大学自主科研计划。程 康(1983 年—), 男, 硕士, 主要研究方向为数值计算方法和超大规模集成电路计算机辅助设计算法; 叶佐昌(1979 年—), 男, 博士, 讲师, 主要研究方向为集成电路计算机辅助设计; 喻文健(1977 年—), 男, 博士, 副教授, CCF 高级会员, 论文通讯作者, 主要研究方向为数值计算方法和超大规模集成电路计算机辅助设计算法。

个元件,同时可以保证电路分析所需的精度.然而同样的方法对寄生参数提取后的模拟与混合信号电路却完全无能为力;其原因在于,这些传统降阶方法只能处理端口(也就是对外连接的节点)数很小的 RC 系统.典型的数字电路中需要关心的互连电路,通常其只有很少的几个端口,用于描述互连的输入、输出、串扰等.对于这样的电路,传统的模型降阶方法具有很好的效果;而当电路的端口数增大时,传统降阶方法的弊端就显现出来.由于传统降阶方法得到的系统往往为稠密系统,当端口数增大时,其效率急剧下降的同时,降阶后的模型复杂度也迅速上升;并且“降阶”后的模型复杂度往往远远大于原始电路的复杂度,虽然其电路节点数小于原始电路,这使得“降阶”变得毫无意义.

近年来,研究者针对上述 2 方面的问题,提出了一系列的解决方法^[3-8].其中,由于具有能够同时解决以上 2 个问题的潜力,基于节点消去的方法^[6-8]一般被认为是针对大规模 RC 系统唯一可能有效的一类方法.现有的基于节点消去的模型降阶方法基本思想是逐个消去网络中的节点,在消去每个节点的同时,对该节点的近邻节点两两之间建立一个连接,连接的 RC 支路的数值根据矩匹配的规则进行确定.基于节点消去的方法,得到的系统往往还具有一定的稀疏性.

相对于传统的降阶方法,虽然基于节点消去的方法得到的模型更为稀疏,然而随着节点的消去,一般情况下网络仍然会变得更加稠密.盲目消去节点往往会导致“降阶”后的电路比原始电路更加复杂,因此节点数的降低与网络复杂度(可以以矩阵非零元数目作为衡量)之间存在一个矛盾,势必在节点数与网络复杂度之间找到某种折中,使得降阶后的电路仿真速度达到最高.

本文提出一种基于优化消去的模型降阶算法(optimal elimination reduction, OER).首先快速得到大规模 RC 网络的剩余节点数与非零元的关系;其次,采用统计建模的方法得到网络求解时间与节点数和非零元的关系,并通过这 2 个关系可以直接得到最优的节点消去策略.实验表明,采用该算法可使得降阶网络的求解速度提高 2~5 倍,同时它也可以与其他的降阶方法相结合,以降低降阶复杂度.

1 网络消去在电路后仿中的应用

在混合信号电路后仿的过程中,不但要考虑线

网的电阻(R)以及自电容(C),还需考虑线网之间的耦合电容(CC),我们称之为“考虑 C+CC+R”的后仿真.图 1 显示了一个考虑 C+CC+R 情况的电路的例子,其节点数(问题规模)大大增加,不仅增加了寄生电阻,而且寄生电容数目也会极大的增加,以至于大规模的模拟与混合信号电路无法进行直接仿真,所以必须对模型进行降阶处理,对于大型 RC 网络系统就是网络约减.其目的是输入大规模 RC 网络系统,输出端口特性近似输入网络,且保持电路无源性的小规模 RC 网络系统.

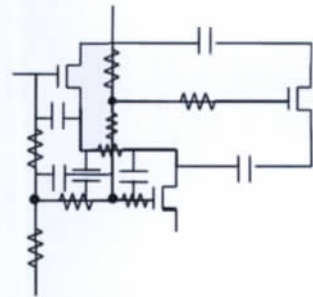


图 1 考虑 C+CC+R 情况的 RC 网络

1.1 投影法

将大规模 RC 网络系统经过标准的节点分析,可以将其写成下面的形式:

$$\begin{cases} \mathbf{E}\mathbf{u}(t) = \mathbf{C}\dot{\mathbf{x}}(t) + \mathbf{G}\mathbf{x}(t) \\ \mathbf{y}(t) = \mathbf{E}^T \mathbf{x}(t) \end{cases} \quad (1)$$

其中 \mathbf{C} 是 RC 网络系统的电容矩阵, \mathbf{G} 是 RC 网络系统的电导矩阵.

通过 Krylov 子空间等投影方法可以对式(1)进行降阶.通过构造不同的子空间,可以得到不同的变换矩阵 \mathbf{V} 和 $\mathbf{W} \in \mathbf{E}^{n \times r}$.根据变换矩阵 \mathbf{V} 和 \mathbf{W} ,就可以得到原始 RC 网络系统的降阶系统为

$$\begin{cases} \tilde{\mathbf{E}}\tilde{\mathbf{u}}(t) = \tilde{\mathbf{C}}\tilde{\dot{\mathbf{x}}}(t) + \tilde{\mathbf{G}}\tilde{\mathbf{x}}(t) \\ \tilde{\mathbf{y}}(t) = \tilde{\mathbf{E}}^T \tilde{\mathbf{x}}(t) \end{cases};$$

其中 $\tilde{\mathbf{x}}(t), \tilde{\dot{\mathbf{x}}}(t) \in \mathbf{R}^r$, $\tilde{\mathbf{E}} = \mathbf{W}^T \mathbf{E} \mathbf{V}$, $\tilde{\mathbf{C}} = \mathbf{W}^T \mathbf{C} \mathbf{V}$, $\tilde{\mathbf{G}} = \mathbf{W}^T \mathbf{G} \mathbf{V}$,此系统即为式(1)的降阶系统.投影法的优点在于精度高,但是它会破坏系统的原有特性,如矩阵的稀疏性以及形状分布的特点无法得到保持.

1.2 节点消去法

对大型 RC 网络进行消去的节点消去法,通过消去部分选定的节点以降低问题的维度.从图 2 可以看出,消去一个节点,器件之间的关联需要进行修改,与被消去节点相连的节点要两两连接.当消去节点 N 时,同时消去了边 AN, BN 和 CN ,在新的图中要将节点 A, B, C 连接起来,产生了新的边 AB, AC 和 BC .

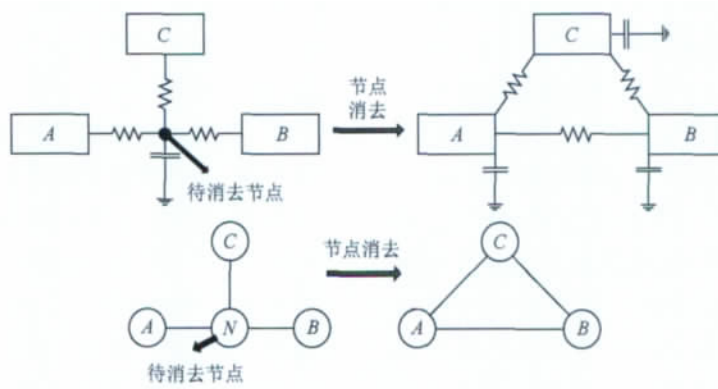


图 2 RC 网络节点消去的物理模型和图的表示

图 3 所示为用矩阵形式表示节点消去的过程：将各个节点作为矩阵的对角元，其中的 6 个黑色方块表示边 NA, NB 和 NC 。当消去节点 N 时，矩阵变为图 3 右侧所示，表示边 NA, NB 和 NC 的黑色方块随着节点 N 消去，但是新产生了表示边 AB, AC 和 BC 的黑色方块。使用矩阵同样可以很好地表示电路的拓扑结构，而且可以提高实际约减过程的速度和可操作性。

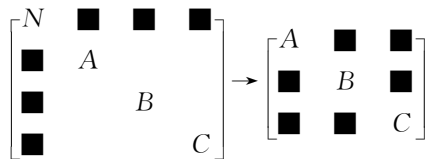


图 3 RC 网络节点消去的矩阵表示

设 G 和 C 分别为 RC 网络的电导、电容矩阵，两者皆为对称正定矩阵，对它们进行电路节点消去的过程与对矩阵执行高斯消去法是完全等价的。以 G 为例，令 $G^{(0)} = G = \begin{bmatrix} a_{11} & b_0 \\ b_0^T & B^{(0)} \end{bmatrix}$ ，消去一个节点，矩阵记为 $G^{(1)}$ ，消去 i 个节点，矩阵记为 $G^{(i)}$ 。

定义第一个消去阵为

$$L^{(0)} = \begin{bmatrix} \sqrt{a_{11}} & \\ -\frac{1}{a_{11}}b_0^T & I_{n-1} \end{bmatrix},$$

那么 $G^{(1)}$ 计算公式为

$$G^{(1)} = L^{(0)} G^{(0)} (L^{(0)})^T = \begin{bmatrix} 1 & 0 \\ 0 & B^{(0)} - \left(-\frac{1}{a_{11}}\right)b_0 b_0^T \end{bmatrix}.$$

同理可得到

$$L^{(i)} = \begin{bmatrix} I & \\ & \sqrt{a_{ii}} \\ & -\frac{1}{a_{ii}}b_i^T & I_{n-i} \end{bmatrix},$$

$$G^{(i)} = \begin{bmatrix} I_{i-1} & & \\ & a_{ii} & b_i \\ & b_i^T & B^{(i)} \end{bmatrix} \quad (2)$$

$$G^{(i+1)} = L^{(i)} G^{(i)} (L^{(i)})^T = \begin{bmatrix} I_{i-1} & & \\ & 1 & 0 \\ & 0 & B^{(i)} - \left(-\frac{1}{a_{ii}}\right)b_i b_i^T \end{bmatrix} \quad (3)$$

那么消去 $i+1$ 个节点后的降阶矩阵为

$$B^{(i)} - \left(-\frac{1}{a_{ii}}\right)b_i b_i^T.$$

从式(2)到式(3)又称为计算一步的 Schur 补 (Schur complement)，持续进行这样的运算可以将矩阵的节点按照其排序的顺序进行消去。现有的节点消去法在一定程度上能够保证模型的稀疏性，但是由于每消去一个节点都要对整个矩阵进行重新计算，运算时间过长，不能很好地满足实际问题的需要。

2 基于优化消去的模型降阶方法

2.1 实际问题的特点和处理的重点

RC 网络对应的矩阵是对称稀疏矩阵，只需要处理矩阵的上三角或下三角部分，而且非零元很少，处理过程中可以采取具有针对性的方法来提高运算效率，减少运算时间。本文算法只需考虑其中元素的数目和分布，不用做实际的数值计算，可以大大地减少运算量。

对 RC 网络进行节点消去，目的是在精度允许的范围内降低模型的阶数，减少电路后仿的时间。RC 矩阵在消去的过程中会产生数量不定的非零元，有可能当节点消去到一定程度后，因为新增连接太多反而使得运算时间大大增加。因此，必须要在 RC 网络保留的节点数和增加的连接数之间做一个评估，以确定一个最佳的消去程度，使得消去得到的

RC 网络具有最小的处理时间.

2.2 快速遍历统计非零元数量

图 4 说明了节点消去过程中如何确定非零元的位置,红色的点是消去过程中产生的填入元.可以理解为 $A(6,1)$ 与 $A(7,1)$ 产生了填入元 $A(7,6)$, $A(3,2)$ 与 $A(8,2)$ 产生了填入元 $A(8,3)$ 那么,当消去第

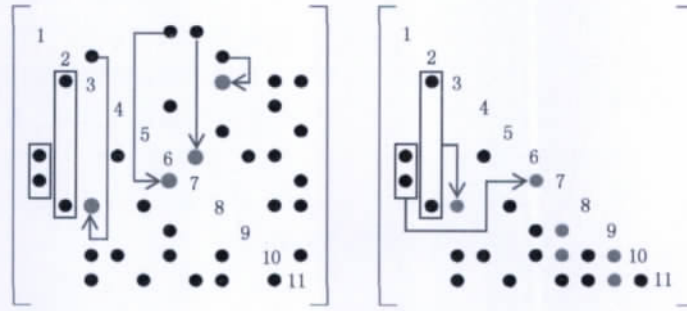


图 4 RC 网络矩阵的节点消去过程

当对大规模的 RC 网络系统进行处理时,可以将其表示为一个对称的稀疏矩阵,使用压缩稀疏行(CSR)的格式存储该矩阵,用 A_p, A_i 和 A_x 3 个向量来存储维度为 $n \times n$,非零元数目为 n_z 的矩阵 A . A_p 是长度为 $n+1$ 的向量, $A_p(i)$ 表示矩阵第 1 至 $i-1$ 行的非零元数目之和, $A_p(1)=0$. A_i 及 A_x 是长度为 n_z 的向量, A_x 中存储的是从第一行开始,列号由小至大的非零元的数值, $A_i(i)$ 中存储的是 $A_x(i)$ 的列号.采用这样的存储方法,可以减少存储空间,矩阵 A 所占用的空间只有 $2 \times n_z + n$.同时也可以方便我们进行消去,遍历查找时可以减少循环次数,进而提高运算速度.

按照以上的存储方法,在实际的运算中存在一个问题,在消去元素和产生填入元的过程中,需要不断地改变向量 A_i 和 A_x 的长度来进行消去和插入,这样使得运算的效率很低.因此,应当在建立向量的同时预测在整个过程中可能产生的填入元的数目,提前申请空间,在计算的过程中只要修改对应的数值即可,不用改变向量的长度,从而减少运算时间.可以对 RC 矩阵进行一次 Cholesky 分解,得到的上三角阵 U 即为填入元最多的情况,将 U 用 CSR 格式存储即可.

消去的主要过程是先对矩阵进行一次强制最小度排序(Camd)^[9],再做一次 Cholesky 分解,然后按照 CSR 格式存储 Cholesky 分解生成的上三角阵 U ,逐行对矩阵进行消去.通过一次遍历,可以得到矩阵消去到不同维度时非零元的确切数目,从而大幅度地降低了运行时间.

三列时,由于该列非零元位于第 8,10,11 列,那么产生的填入元将位于 $A(10,8)$, $A(11,8)$ 和 $A(11,10)$,但是由于这 3 个位置已有元素,所以不产生填入.消去某列元素时,根据该列元素的行号即可确定填入元的位置,产生的填入元在消去其所在列时也可能产生新的填入元.

2.3 根据消去结果预测矩阵处理时间

我们最终的目的是得到一个新的矩阵,使得处理这个矩阵的时间比原始矩阵时间要少.但是,在矩阵消去的过程中,非零元数目的变化没有规律,不同的矩阵消去过程中非零元数目变化差别很大,如图 5 所示,只有通过实际的消去才能得到矩阵非零元数量的变化情况.

图 5 展示了 3 个 RC 矩阵消去过程中非零元数量(y 轴)、维度(x 轴)的变化过程和最优消去的位置.可以看出,最优消去的位置并不全是非零元最少或者维度最小的点,所以在得到了矩阵消去过程中维度与非零元的确切关系后也不能主观地得到最优消去点.一个矩阵在消去的过程中,假设维度降到初始的十分之一,但是非零元的数目增加到了初始的 3 倍,那么选择哪个做仿真速度更快一些是不确定的.

我们需要知道的是处理不同维度、不同非零元数目的矩阵所需要的时间,以及时间、维度、非零元三者之间的关系.对消去程序加以改进,将每消去一个节点后产生的新矩阵输出,用它进行相同的操作,如 Cholesky 分解,统计出对应的运算时间.通过大规模的实验,用经验公式 $f = \alpha + \beta * n_z + \gamma * n$ 进行最小二乘拟合,解出 α, β, γ 的值.通过这 3 个参数,在得到了矩阵维度和非零元的关系后,我们就可以预测出矩阵在消去到不同程度时进行仿真所需要的运行时间,从而确定最优消去点,降低仿真用时.

下面给出本文提出的基于优化消去的模型降阶算法.

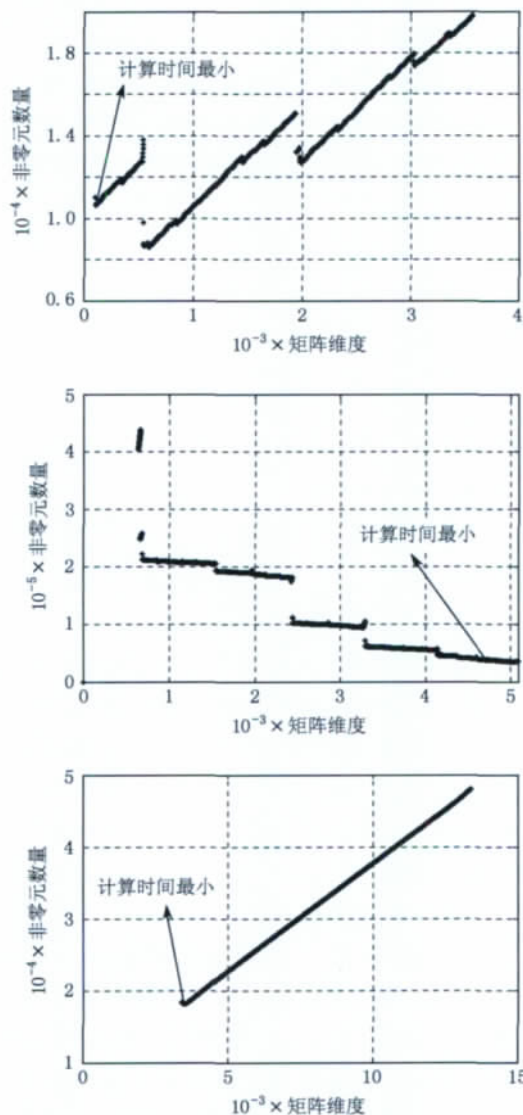


图 5 矩阵消去过程中非零元数量变化

算法 1. 基于优化消去的模型降阶算法

- Step1. 将矩阵 A 做 Camd 排序, 把不可消去的节点置于矩阵对角线的右下角, 数量为 s , 取矩阵上三角部分为 U_1 .
- Step2. 对更新后的矩阵 A 做 Cholesky 分解, 得到上三角阵 U_2 .
- Step3. 将 U_1, U_2 中所有元素置为 1, $U=U_1+U_2$.
- Step4. 将 U 转化为 CSR 格式, 生成向量 A_p, A_i, A_r .
- Step5. For $m=1; \text{size}(A_p, 1)-1-s$
- Step6. $A'_i = A_i[A_p(m)+1; A_p(m+1)]$.
- Step7. 将 A'_i 中的元素 (即记录的该行非零元的列号) 两两组合记为 (j, k) , 提取其中第一个数字较小 $j < k$ 的组合方式.
- Step8. 到 $A_i[A_p(j)+1; A_p(j+1)]$ 中去查找列号为 k 的位置 l , 即 $A_i(l)=k$.
- Step9. 如果 $A_r(l)=1$, 则产生填入元.
- Step10. 将 A_r 中对应 A_i 中遍历过元素的值为 1.

Step11. End

Step12. 将更新完成后的向量 A_p, A_i, A_r 转换为矩阵 U_3 .

Step13. $\bar{U} = U_3 - U_1$, 取 \bar{A} 为 $\bar{U} + \bar{U}^T$ 的右下 $(n-m) \times (n-m)$ 部分, n 为 \bar{U} 维度.

Step14. 将获得的维度与非零元用参数进行计算, 确定运算时间最少的消去程度.

由于 RC 网络进行仿真时, 需要考虑到电容矩阵 C 和电导矩阵 G , 所以进行优化消去时也应当同时考虑到两者的影响. 算法 1 中的矩阵 $A = G + C$, 对 A 进行了一定的处理, 舍去了部分极小的数值, 这样得到的经验公式同时反映了电导和电容的影响.

3 数值实验结果

本文实验均运行于 8 核 Intel(R) Xeon(R) CPU 的 Linux 工作站, CPU 的主频为 2.4 GHz. 程序由 Matlab 和 C 语言混合编程实现.

矩阵 T_1 是维度 93 377、非零元数目 567 297、不可消去的节点数为 8 609 的 RC 矩阵; T_2 是维度 446 951、非零元数目 2 456 719、不可消去的节点数为 63 764 的 RC 矩阵. 按照不同的拆分方式将 T_1 分解成 89, 45, 34 个子矩阵, 称为 T_{11}, T_{12}, T_{13} ; T_2 拆分为 61 个子矩阵, 称为 T_{21} .

为了作为对比, 我们在消去每个节点时先对矩阵进行一次 Camd 排序, 并在排序的过程中利用约束将所有的端口节点排列到最后, 然后再计算其一步的 Schur 补, 最终只保留全部的端口节点, 此方法简称为 C&S 方法. C&S 方法与 OER 方法进行预处理时间及加速比如表 1 所示.

表 1 不同 RC 矩阵用 C&S 及 OER 方法做预处理的时间对比

RC 矩阵	运行时间/s		加速比	
	C&S	OER		
T_1	T_{11}	460	3.9	117X
	T_{12}	1 536	7.95	193X
	T_{13}	3 349	13.29	252X
T_2	T_{21}	失败	288.3	—

表 2 中的 T_3 是维度 22 851、非零元 116 629、不可消去的节点数 3 404 的 RC 矩阵. 依次选择不做处理 (none), C&S, OER 3 种算法处理后, 进行 10 000 次 Cholesky 分解所需要的时间.

表 2 不同预处理方法得到的矩阵做仿真的时间对比

RC 矩阵	时间/s			加速比	
	none	C&S	OER		
T_{11}	2 620	1 959	1 340	1.96X	
T_1	T_{12}	3 233	4 627	1 343	2.41X
	T_{13}	3 129	7 136	1 526	2.05X
T_2	T_{21}	12 869	13 976	9 339	1.38X
T_3		2 415	29 664	445	5.43X

根据拟合的结果, $\alpha = -5.5665 \times 10^{-4}$, $\beta = 2.0945 \times 10^{-7}$, $\gamma = 2.2567 \times 10^{-6}$, 对预处理得到的矩阵维度和非零元计算可以得到仿真时间最少的矩

阵消去程度. 矩阵 T_2 分割出来的 61 个子矩阵的消去过程大部分和图 5 第 2 个矩阵相似, 未对其进行预处理已具有比较好的运算性能, 所以和用矩阵直接运算对比, 速度提升比较有限. T_3 则不同, 矩阵非零元数量经过消去后会减少, 但是在最后会呈现极快的增加, 如果采用 C&S 方法, 得到的矩阵将是一个满阵, 求解时间反而会大幅增加; 与 C&S 方法相比, OER 方法更好地确定了消去的程度, 获得了 5 倍以上的加速比. 图 6 显示的是一个真实的 RC 矩阵使用 OER 方法得到的预测运算时间(蓝色数据点)和实际测得的运算时间的对比, 图 6 a 是三维视图, 图 6 b, 图 6 c 分别是 2 个不同方向的侧视图.

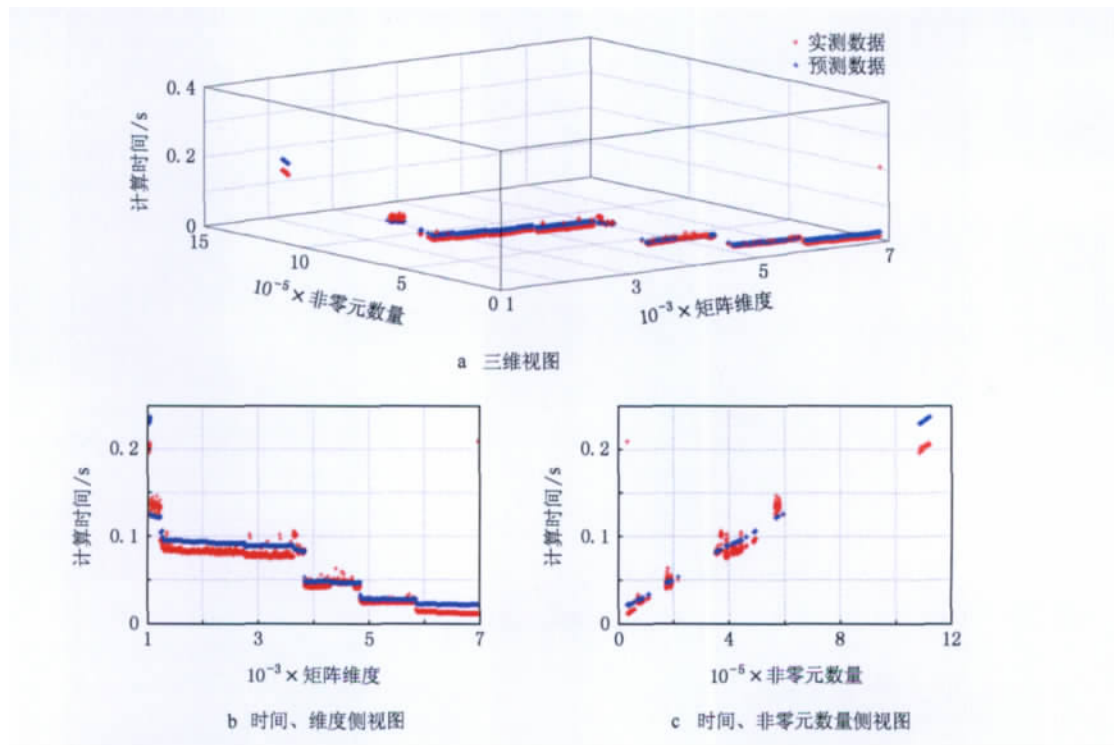


图 6 矩阵消去到不同程度做 Cholesky 分解所需时间的预测结果

如果将矩阵预处理的时间与矩阵处理完毕后进行一次运算的时间相加, 这个时间可能会比用原矩阵直接运算要长, 但是在数模混合电路的后仿中, 同样的一个矩阵往往要求解很多次, 甚至多达上万次, 因此预处理所需的时间开销是值得的, 它能使总的仿真时间显著缩短.

OER 方法的核心依然是节点消去法, 没有方法误差, 它和其他的基于节点消去方法得到的非零元数量及分布是相同的, 降阶精度也是相同的. 使用通过统计建模得到的拟合参数对矩阵处理时间进行预测, 平均误差在 10% 左右, 但是能准确地找到最佳的消去节点数, 也就是说能较好地反映矩阵处理时间随着矩阵维度及非零元数量变化的趋势.

4 结 论

本文针对大规模网络模型降阶, 提出优化的节点消去算法. 该算法首先对于大规模的 RC 网络进行节点消去, 快速得到剩余节点数与非零元的关系; 其次采用统计建模的方法得到网络求解时间与节点数和非零元的关系; 并通过这 2 个关系可以直接得到最优的节点消去算法. 此算法可与 SIP 方法^[8] 结合得到复杂度最优的降阶网络. 采用实际 IC 设置实验表明, 采用该算法可使得降阶网络的求解速度提高 2~5 倍.

参考文献 (References):

- [1] Odabasioglu A, Celik M, Pileggi L. PRIMA: passive reduced-order interconnect macromodeling algorithm [C] // Proceedings of IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. Los Alamitos: IEEE Computer Society Press, 1997: 58-65
- [2] Phillips J R, Silveira L M. Poor man's TBR: a simple model reduction scheme [C] // Proceedings of IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. Los Alamitos: IEEE Computer Society Press, 2005: 43-55
- [3] Feldmann P. Model order reduction techniques for linear systems with large numbers of terminals [C] // Proceedings of the Conference on Design, Automation and Test in Europe. Washington D C: IEEE Computer Society Press, 2004: 944-947
- [4] Feldmann P, Liu F. Sparse and efficient reduced order modeling of linear subcircuits with large number of terminals [C] // Proceedings of International Conference on Computer Aided Design. Los Alamitos: IEEE Computer Society Press, 2004: 88-92
- [5] Li P, Shi W P. Model order reduction of linear networks with massive ports via frequency-dependent port packing [C] // Proceedings of the 43rd annual conference on Design automation. New York: ACM Press, 2006: 267-272
- [6] Kerns K J, Yang A T. Preservation of passivity during RLC network reduction via split congruence transformations [C] // Proceedings of the 34th Design Automation Conference. Los Alamitos: IEEE Computer Society Press, 1997: 34-39
- [7] Sheehan B N. TICER: realizable reduction of extracted RC circuits [C] // Proceedings of International Conference on Computer Aided Design. Los Alamitos: IEEE Computer Society Press, 1999: 200-203
- [8] Ye Z C, Vasilyev D, Zhu Z H, *et al.* Sparse implicit projection (SIP) for reduction of general many-terminal networks [C] // Proceedings of IEEE/ACM International Conference on Computer Aided Design. Los Alamitos: IEEE Computer Society Press, 2008: 736-743
- [9] Amestoy P, Davis T A, Duff I S. Algorithm 837: AMD, an approximate minimum degree ordering algorithm [C] // Proceedings of ACM Transactions on Mathematical Software. New York: ACM Press, 2004: 381-388



(上接第 1505 页)

- [8] Li Xin, Wu Fuli, Chen Weifeng, *et al.* Visualizing tetrahedral volume datasets with quick adaptive multiway sorting [J]. Journal of Computer-Aided Design & Computer Graphics, 2011, 23(12): 2025-2032 (in Chinese)
(李昕, 吴福理, 陈伟锋, 等. 基于快速自适应多路排序的四面体可视化. 计算机辅助设计与图形学学报, 2011, 23(12): 2025-2032)
- [9] Williams P L. Visibility ordering meshed polyhedral [J]. ACM Transactions on Graphics, 1992, 11(2): 103-126
- [10] Silva C T, Mitchell J S B, Williams P L. An exact interactive time visibility ordering algorithm for polyhedral cell complexes [C] // Proceedings of ACM Symposium on Volume Visualization. New York: ACM Press, 1998: 87-94
- [11] Comba J, Klosowski J, Max N, *et al.* Fast polyhedral cell sorting for interactive rendering of unstructured grids [J]. Computer Graphics Forum, 1999, 18(3): 369-376
- [12] Cook R, Max N, Silva C T, *et al.* Image-space visibility ordering for cell projection volume rendering of unstructured data [J]. IEEE Transactions on Visualization and Computer Graphics, 2004, 10(6): 695-707
- [13] Farias R, Mitchell J, Silva C. ZSWEEP: an efficient and exact projection algorithm for unstructured volume rendering [C] // Proceedings of IEEE Volume Visualization and Graphics Symposium. New York: ACM Press, 2000: 91-99
- [14] Crow F C. Summed-area tables for texture mapping [C] // Proceedings of the 11th Annual Conference on Computer Graphics and Interactive Techniques. New York: ACM Press, 1984: 207-212
- [15] Hensley J, Scheuermann T, Coombe G, *et al.* Fast summed-area table generation and its applications [J]. Computer Graphics Forum, 2005, 24(3): 547-555