

考虑温度/功耗/热导之间相互作用的 单循环迭代热分析算法

潘月斗^{1,2}, 王嘉琪^{1,2}, 唐 亮³, 骆祖莹³

(1. 北京科技大学自动化学院, 北京 100083; 2. 北京科技大学钢铁流程先进控制教育部重点实验室, 北京 100083;
3. 北京师范大学信息科学与技术学院, 北京 100875)

摘 要: 随着纳米工艺的不断改进, 温度对漏电流功耗和热导的影响日益显著. 考虑温度/功耗/热导相互作用的 3D 芯片热分析需要采用迭代方法对温度进行精确求解, 即先用功耗密度向量和热导矩阵来求解温度向量, 再用求解出来的温度向量来刷新功耗密度向量和热导矩阵. 为了提高 3D 芯片热分析的效率, 本文以一个设定温度值下的均匀热导矩阵作为预条件, 先提出了一种双循环、内循环低迭代次数的高效求解算法 TPG-FTCG. 鉴于 TPG-FTCG 具有超快的内循环收敛速度, 本文省去了 TPG-FTCG 算法的内循环部分, 提出了一种单循环、低迭代次数的 TPG 求解算法 TPG-Sli. 基于 GPU (Graphics Processing Unit) 并行加速技术, 本文编写并改进了 TPG-Sli 的 GPU 加速算法. 实验数据表明: 与采用经典高效的 ICCG 算法进行 3D 芯片热分析的 TPG-ICCG 算法相比, 在足够小的误差范围内, TPG-Sli 的 GPU 加速算法可以获得 120 倍的速度提升.

关键词: 算法; 热分析; 快速傅里叶变换; GPU 并行

中图分类号: TP393 **文献标识码:** A **文章编号:** 0372-2112 (2016)06-1300-07

电子学报 URL: <http://www.ejournal.org.cn>

DOI: 10.3969/j.issn.0372-2112.2016.06.006

TPG-Sli: Single-Loop Iterative Thermal Analysis Algorithm Considering Interactions Among Temperature, Power and Heat Conductance

PAN Yue-dou^{1,2}, WANG Jia-qi^{1,2}, TANG Liang³, LUO Zu-ying³

(1. School of Automation, University of Science and Technology Beijing, Beijing 100083, China;
2. Key Laboratory of Advanced Control of Iron and Steel Process (Ministry of Education), Beijing 100083, China;
3. College of Information Science and Technology, Beijing Normal University, Beijing 100875, China)

Abstract: With the improvement of the nanometer technology, the influences among temperature, leakage power and heat conductance become increasingly significant and it should be taken into account in 3D chip comprehensive thermal analysis to solve the accurate temperature based on the iterative solution. The comprehensive thermal analysis method uses the nodal power density vector and the heat conductance matrix to solve the nodal temperature vector, and then, refreshes power density and heat conductance with the obtained nodal temperature. In order to improve the efficiency of 3D chip comprehensive thermal analysis, this work uses the heat conductance matrix as the precondition under a setting temperature. Then it proposes an efficient algorithm TPG-FTCG (CG with the Fast Transform-based Preconditioner) which has double-loop and lower inner-loop iterations. According to TPG-FTCG's fast inner-loop convergence rate, this work removes TPG-FTCG's inner-loop part then proposes a more efficient TPG solving algorithm TPG-Sli (Single-loop iterative), which only has single-loop iterative and fewer iterations. Based on the GPU parallel computing, this work compiles and refines TPG-Sli's GPU-parallel-computing algorithm. Experimental results demonstrate that: On the premise of precision losing, the TPG-Sli's GPU algorithm can achieve about 120X speedup compared with the TPG-ICCG algorithm, which uses the classical and efficient ICCG to deal with the 3D chip comprehensive thermal analysis.

Key words: algorithm; thermal analysis; Fast Fourier transform; GPU parallel computing

1 引言

随着集成电路 (Integrated Circuit, IC) 纳米工艺的不断改进, 3D 芯片已成为事实, 提高了 IC 集成度与功耗,

但不可避免地带来了发热问题, 过高的工作温度可能导致芯片无法正常工作、甚至损坏芯片. 因此在芯片设计中要反复进行温度优化, 研究精确高效的 3D 芯片热分析方法已成为了一个 EDA (Electronic Design Automa-

tion) 领域的研究热点。

在纳米 IC 热分析中,温度/功耗/热导参数之间存在着明显的相互影响.一方面随着衬底温度的升高,晶体管漏电流功耗显著增加,推高最终的芯片温度^[1].另一方面随着温度的升高,硅材料的热导率会出现明显的下降,使得热阻升高,也会影响最终的温度分布^[2].本文综合考虑了温度/功耗/热导($T/P/G$)三方面相互影响的关系,对考虑 $T/P/G$ 相互影响的 TPG 热分析方法及其快速求解算法进行了系统地研究.

为了综合考虑 $T/P/G$ 之间的相互影响,TPG 采用迭代求解方法逼近精确解,即先用功耗密度向量 P 和热导矩阵 G 来求解温度向量 T ,再用求出的 T 来刷新 P 和 G .由于在迭代过程中热导矩阵 G 是变化的,尽管现有的不完全乔莱斯基分解共轭梯度算法 ICCG^[3]能够根据 P 和 G 来迭代求解 T ,如果采用 ICCG 进行 TPG 求解,就构成了内外双循环迭代求解算法 TPG-ICCG,但其收敛速度较慢,每次求解 T 都需要很多次迭代,TPG-ICCG 算法具有外循环迭代次数少和内循环迭代次数多的特点,过多的迭代次数降低了 TPG 的分析效率.

为了提高 TPG 的分析效率,本文采用 FTCCG 算法^[4](Precondition Conjugate-Gradient algorithm with the Fast Transform-based Preconditioner)来构建一种双循环、内循环低迭代次数的高效求解算法 TPG-FTCCG,该算法以一个设定温度值下的均匀热导矩阵 G_A 作为预条件,每次求解 T 仅需要极少次数的迭代.鉴于 TPG-FTCCG 具有超快的内循环收敛速度,本文直接省去其内循环,提出了一种单循环、低迭代次数的高效 TPG 求解算法 TPG-Sli (Single-loop iterative),该算法先以 G_A 为预条件、采用双重快速傅里叶变换(FFT)的共轭梯度方法直接根据 P 和 G 求解出含微量残差的 T ,再用这个含微量残差的 T 来刷新 P 和 G ,最后收敛于一个具有较高精度的求解结果.基于 GPU 并行计算技术,本文编写了 GPU 并行算法 TPG-Sli_{-GPU},并利用系数矩阵预存储技术编写其优化算法 TPG-Sli_{-iGPU}.

2 相关工作与本文贡献

在芯片设计的各种阶段都需要精度与速度各异的 热分析算法来针对芯片温度进行设计优化,目前热分析算法包括:结构级热分析^[4,13]、全芯片三维热分析^[5-7]、统计热分析^[8]等算法.全芯片三维热分析又分为稳态与瞬态热分析两种^[5-7],其中稳态热分析是其它热分析的研究基础,一直得到广泛的研究,以适应不同应用的需求^[8,13-15].

全芯片三维稳态热分析一般会对散热系统进行离散化以获得热导矩阵 G ,然后根据输入的功耗密度向量 P ,通过求解热分析方程 $GT = P$,来获取芯片的温度分

布向量 T ^[5,6].目前采用逐次逼近的迭代方法进行求解的热分析算法主要有 ICCG^[3]、格林函数算法^[9,10]、变向隐含算法(Alternating Direction Implicit, ADI)^[7]、连续过松弛算法(Successive Over Relaxation, SOR)^[8,15]、多层网格算法(Multigrid, MG)^[6,14]等.其中 ICCG 是具有强鲁棒性的经典算法,被广泛地用做热分析算法研究的对比算法.对于迭代算法而言,只要能够降低其迭代的次数就能够降低其算法复杂度、提高其算法效率.

在热分析中,一般都假设芯片具有绝热四壁^[7],如果假设与器件层平行的 X - Y 平面内的导热介质具有同质导热性能,那么芯片进行均匀离散化后,就可以采用基于 FFT 变换的热分析算法 FT^[4,11,12]进行直接求解,由于不需要迭代,FT 算法具有高效的求解效率^[11].对于 X - Y 平面具有非同质导热性能的应用,可以基于同质结构来构造一个预条件热导矩阵 G_A ,采用预条件共轭梯度算法(PCG)或残差补偿方法进行迭代求解,如果 G_A 越近似于实际热导矩阵 G ,则热分析需要的迭代次数就越少,热分析的求解效率就越高^[4].

随着纳米工艺的不断改进,已有热分析研究不仅考虑温度升高对漏电流功耗的影响^[1,8],也开始考虑温度升高对硅衬底热导的影响^[2].由于在热分析中考虑 T 对 G 影响的研究才起步,目前还没有看到关于 TPG 热分析的研究成果.

与现有热分析算法研究相比,本文工作的贡献体现在如下方面:

(1) 基于文献[4,12]所给出的 FTCCG 算法,本文提出了 TPG-FTCCG 算法,先以环境温度下的热导矩阵 G_A 作为 G 的预条件矩阵,便于 FT 算法进行快速的预条件求解.与平均需要 410 次迭代才能求解出 T 的 TPG-ICCG 算法相比,TPG-FTCCG 算法平均只需要 4~5 次迭代就能够求解出 T ,在可以忽略的误差范围内,TPG-FTCCG 可以获得 20 倍左右的加速.

(2) 为了进一步提高分析效率,本文提出了一种全新的单循环、高效 TPG 求解算法 TPG-Sli.由于省却了内循环,TPG-Sli 算法求解效率比 TPG-FTCCG 提高了 3 倍.

(3) 本文编写了 GPU 加速算法 TPG-Sli_{-GPU} 和改进算法 TPG-Sli_{-iGPU} 算法.在足够小的误差范围内,TPG-Sli_{-iGPU} 算法求解效率比 TPG-FTCCG 提高了 6 倍,比 TPG-ICCG 提高了 120 倍.

3 研究基础

3.1 3D 集成芯片的全芯片三维热分析模型

如图 1 所示,3D 集成芯片主要对多个硅器件层进行垂直堆叠,3D 芯片上下分别接着主散热通道和辅散热通道,鉴于主散热通道的散热能力是辅散热通道的 20 多倍,本文在 3D 芯片热分析中,忽略了辅散热通道、

只考虑主散热通道.

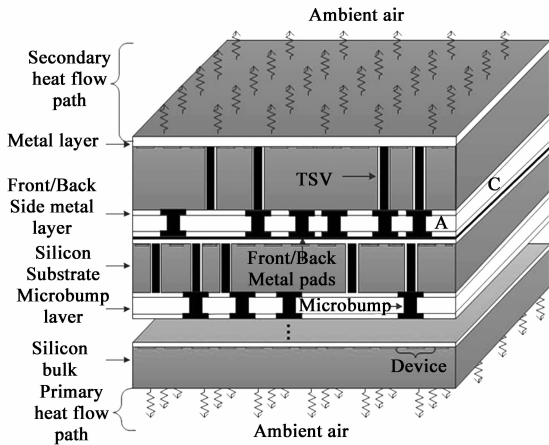


图1 堆叠式3D芯片模型^[12]

如图2所示,本文首先要对3D芯片内核进行离散化^[12],将内核等分为 $n_x(x轴) \times n_y(y轴) \times n_z(z轴)$ 个长方体质元,并将每个质元等效为一个节点,即每个质元所产生的热量都被加到它的中心点,并将中心点温度作为质元温度. 本文采用不同灰度的模块来表示不同材料质元:(1)灰度较浅的块为硅衬底离散后的硅质元,本文采用 $16.7\mu\text{m}$ 厚度将剪薄硅衬底离散为3层,依次采用 $16.7\mu\text{m}/50\mu\text{m}/100\mu\text{m}/150\mu\text{m}$ 厚度将常规厚度($500\mu\text{m}$)硅衬底层离散为8层;(2)图2中A和B块为金属互连层的质元,本文金属互连层的铜材料占比设为50%,3D芯片功耗分布在互连层,其中附着于常规衬底的互连层功耗占总功耗的4/5,附着于剪薄衬底的互连层占总功耗的1/5;(3)图1中的A和C部分是粘接层的微压焊块质元与绝缘材料质元,其中微压焊块层厚度为 $30\mu\text{m}$,微压焊块质元直径为 $30\mu\text{m}$,均匀排列,微压焊块热导率为 $60\text{W}/(\text{m} \cdot \text{K})$,绝缘材料热导率为 $0.83\text{W}/(\text{m} \cdot \text{K})$;(4)图2中C块为TSV质元,TSV直径为 $6\mu\text{m}$,TSV的热导率为 $406\text{W}/(\text{m} \cdot \text{K})$,均匀排列在微压焊块质元下方.

3D芯片离散化后,采用下式进行稳态热分析:

$$GT = P \quad (1)$$

式中, T 与 P 分别是节点温度分布向量与功耗密度分布向量, G 为热导矩阵.

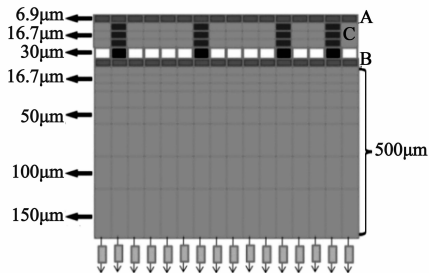


图2 两器件层堆叠式3D芯片的离散建模

3.2 温度与功耗的相互影响

随着芯片工作温度 T 的提高,导致芯片漏电流 I_{lk} 显著地增高^[1],二者的函数关系如下式所示:

$$I_{lk}(T) = I_{lk}(V_0, T_0) (AT^2 e^{\alpha V_{ds} + \beta/T} + Be^{\gamma V_{ds} + \delta}) \quad (2)$$

式中, V, T 分别是电压与温度向量,且 V_0, T_0 为它们的初始向量, $I_{lk}(V_0, T_0)$ 初始的漏电流向量, $A, B, \alpha, \beta, \gamma$ 和 δ 是不同芯片规模下的经验常数^[1]. 根据式(1)和式(2)可以看出,温度升高导致功耗增加,而功耗增加又会反过来会推高温度.

3.3 温度与热导的相互影响

文献[2]详细地说明了温度与硅材料热导率在热分析中的相互作用. 当温度从300K升到400K时,根据下式给出的温度与硅材料热导率之间的二次拟合函数,硅的热导率将下降32%左右.

$$k(T) = 0.0018 * T^2 - 0.7575 * T + 166.5162 \quad (3)$$

即温度 T 和热导 G 在热分析中同样存在一个正反馈关系. 根据式(1)~(3)可以看出,温度升高导致硅热导率下降,反过来会推高温度与功耗.

3.4 温度、功耗、热导之间的相互影响

如图3中框所示,热分析中热导矩阵 G 和功耗密度向量 P 决定了节点温度向量 T ,反映了热导和功耗对于温度的影响. 如图3左框所示,根据节点温度向量 T 来计算热导矩阵 G ,以反映温度对热导的影响. 如图3右框所示,根据节点温度向量 T 来计算功耗密度向量 P ,以反映温度对于功耗的影响. 因此, $T/P/G$ 之间存在着直接的相互影响关系.

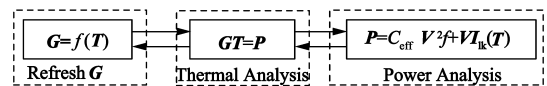


图3 $T/P/G$ 之间的相互影响关系

4 TPG 热分析方法及其高效算法

4.1 考虑温度/功耗/热导之间相互影响的热分析方法 TPG 及其算法流程

在初始温度条件下,TPG首先利用初始的功耗密度向量 P_0 和热导矩阵 G_0 计算温度向量 T_0 ;然后根据 T 分别刷新 P 和 G ,从而计算出更为精确的 T . 就这样TPG方法采用迭代的方法不断产生新的 T 直到收敛于 T 的准确解,最后通过对 P 进行累加获取精确的芯片功耗 P_{chip} . 图4给出了TPG方法的算法流程图,TPG算法包含以下算法步骤.

Step 1 先输入室温 $T_a = 45^\circ\text{C}$ 下的 G_0 和 T_0, P_0 ;再将 $G_0/T_0/P_0$ 赋值给 $G/T/P$.

Step 2 求解 $GT = P$ 以获得温度向量 T .

Step 3 判断任意节点 i 的温度变化量 $|\delta T_i|$ 是否小于给定误差限 $\varepsilon_1 = 1e^{-2}$,如果是,则停止迭代跳到Step 5.

Step 4 根据温度向量 T 刷新热导矩阵 G 和功耗密度向量 P , 返回 Step 2 进行下一轮的迭代。

Step 5 输出精确的温度向量 T 和芯片功耗 P_{chip} 。

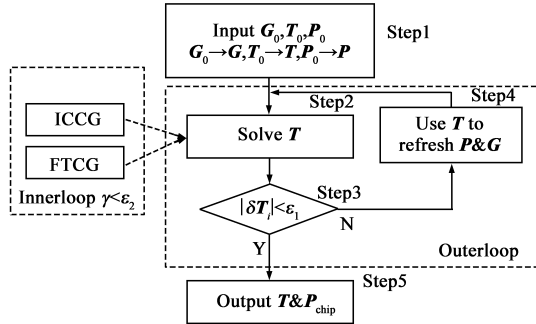


图4 TPG方法的整体算法流程图

由于在 Step 4 中根据 T 刷新 G 和 P 的复杂度很小, 而 Step 2 求解 $GT = P$ 的算法复杂度则很大, 所以选用高效率算法来求解 T 决定了 TPG 热分析的效率。鉴于热导矩阵 G 是非均匀变化的, 无法使用 FT 算法来直接求解 $GT = P$, 必须采用迭代算法来精确求解 T , 以构成具有内外两个循环迭代求解的 TPG 热分析算法。

如果采用 ICCG 算法来求解 Step 2 $GT = P$, 就构成了 TPG-ICCG 热分析算法。如表 1 所示, 对一个具有 14M 节点的测例, 当 ICCG 的迭代判出残差限设为 $\epsilon_2 = 1e^{-6}$ 时, 尽管外循环的迭代次数仅有 13 次就能够收敛, 但由于内循环的迭代次数非常多, 进行一次 TPG 分析所需要迭代总数达到 5336 次, 平均内循环的迭代次数为 410 次, 这表明 ICCG 收敛速度太慢, 难以为大规模热分析提供满意的求解效率。

表 1 TPG-ICCG、TPG-FTCG 的收敛次数对比

Outer loop of TPG	Iteration number of Inner loop	
	TPG-ICCG	TPG-FTCG
1	638	21
2	591	20
3	669	13
4	581	5
5	573	5
6	468	1
7	419	1
8	336	1
9	269	1
10	267	1
11	218	1
12	178	1
13	129	1
14		1
15		1
16		1
17		1
18		1
Total	5336	77

4.2 基于快速傅里叶变换进行预条件求解的共轭梯度算法 TPG-FTCG

为了提高 TPG 热分析的求解效率, 在图 4 中本文将 FTCG 算法^[3]引入 TPG 热分析中, 构建了 TPG-FTCG 算法。如图 5 所示, FTCG 算法能够以较高的收敛速度对 $GT = P$ 进行求解。

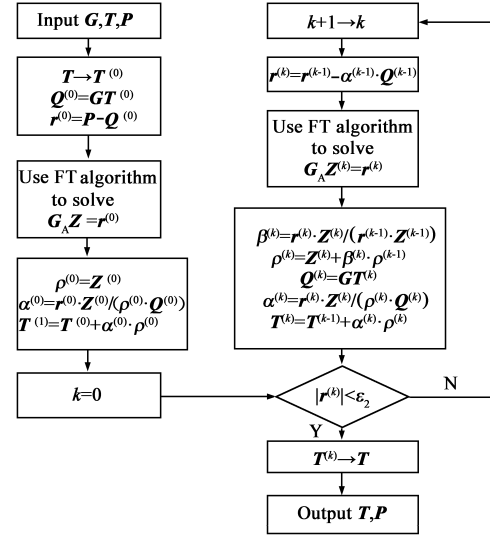


图5 构成TPG_FTCG的FTCG算法流程图

3D 芯片存在非同质热导问题, 无法直接用室温 $T_a = 45^\circ\text{C}$ 的热导矩阵来构建满足 FT 算法直接求解的热导矩阵 G_A , 下面 TSV 质元为例, 说明如何构建满足要求的 G_A 。在图 6 中, 将 TSV 质元中心点温度设为节点温度, 需要获取该质元六个方向的热导才能计算出其中中心点温度。两个相邻质元的热导可以看作是两质元中心点到边界面热导值的串联, 即

$$G_{\text{TSV}}^{-x} = g_{\text{si}}^{x+1/2} // g_{\text{TSV}}^{x-1/2}, G_{\text{TSV}}^{+x} = g_{\text{si}}^{x-1/2} // g_{\text{TSV}}^{x+1/2};$$

$$G_{\text{TSV}}^{-y} = g_{\text{si}}^{y+1/2} // g_{\text{TSV}}^{y-1/2}, G_{\text{TSV}}^{+y} = g_{\text{si}}^{y-1/2} // g_{\text{TSV}}^{y+1/2};$$

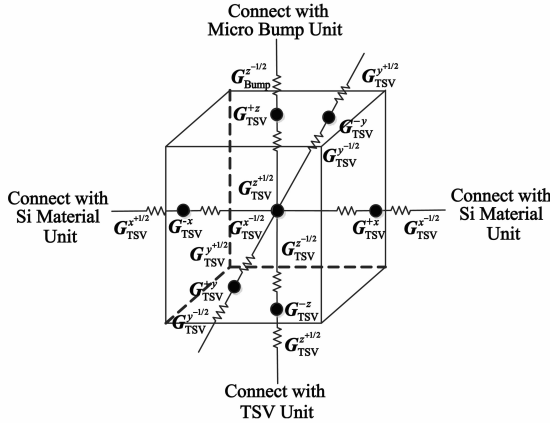
$$G_{\text{TSV}}^{-z} = g_{\text{si}}^{z+1/2} // g_{\text{TSV}}^{z-1/2}, G_{\text{TSV}}^{+z} = g_{\text{si}}^{z-1/2} // g_{\text{TSV}}^{z+1/2}.$$

其中 $G_{\text{TSV}}^{-x}, G_{\text{TSV}}^{+x}, G_{\text{TSV}}^{-y}, G_{\text{TSV}}^{+y}, G_{\text{TSV}}^{-z}, G_{\text{TSV}}^{+z}$ 分别表示该 TSV 质元的 $-x$ 方向和 $+x$ 方向, $-y$ 方向和 $+y$ 方向, $-z$ 方向和 $+z$ 方向的热导。由于 TSV 质元垂直方向相邻的质元材料不同 ($+z$ 方向是微压焊块质元, $-z$ 方向是 TSV 质元), G_{TSV}^{-z} 与 G_{TSV}^{+z} 的值并不相同, 这就使得该层热导矩阵 G_A^{TSV} 无论在水平方向与垂直方向的值分布都不均匀。

文献[14]规定: 参与 FT 算法的预条件矩阵值不能小于原矩阵最大值的 50%, 本文选取每层最大的热导值 (无论是水平热导还是垂直热导) 的 93% 作为预条件的矩阵热导值。以图 6 为例, 该 TSV 所在硅衬底层的预条件矩阵水平方向热导值即为

$$G_{\text{TSV}}^{\text{p}} = 93\% * \max(G_{\text{TSV}}^{-x}, G_{\text{TSV}}^{+x}, G_{\text{TSV}}^{-y}, G_{\text{TSV}}^{+y}, G_{\text{Si}}^{x,y})$$

垂直方向热导值即为

图6 TSV所在硅衬底层的TSV质元模型^[12]

$$G_{TSV}^y = 93\% * \max(G_{TSV}^{-z}, G_{TSV}^{+z}, G_{Si}^{\pm z})$$

式中 $G_{Si}^{\pm xy}$ 表示两个硅质元之间的水平互导, $G_{Si}^{\pm z}$ 表示该层硅质元与其垂直正负方向的互导. 获得每层的预条件水平热导和垂直热导后, 就可以构成了符合 FT 算法要求的预条件矩阵 G_A , 用于 FT 算法求解 $G_A Z^{(k)} = r^{(k)}$ 以获得 $Z^{(0)}$ 和 $Z^{(k)}$.

对于进行热传导的硅材料而言, 当工作温度从 45℃ 升高到 90℃ 时, 其热导率从 136.0199 下降到 113.1025, 即下降了 16.85%, 这表明: 对于一个最高工作温度小于 90℃ 的芯片, 相对于室温 T_a 下的预条件矩阵 G_A , 实际热导矩阵 G 的热导变化小于 16.85%, 远小于文献[12]设定的 50% 限制. 对于 FT-TCG 算法而言, G_A 越接近于 G , 则其收敛的速度越快, 如表 1 所示, 本文 FT-TCG 算法除了前 5 次迭代中需要 521 次迭代以外, 剩余迭代仅需要 1 次就可以将迭代残差降到设定值 10^{-6} 之下, 具有很好的收敛特性.

4.3 改进的单循环算法 TPG-Sli

根据 TPG-FTCG 算法残差收敛速度极快的特点, 本文进一步删除了图 6 中 FT-TCG 算法的循环体部分, 仅使用 FT 算法进行一次 T 的预条件共轭梯度求解, 提出了单循环的改进算法 TPG-Sli, 以进一步降低 TPG 热分析算法的时间复杂度.

如图 7 所示, TPG-Sli 算法是一个单循环的迭代算法, 每次迭代不是直接求解 T , 而是先根据残差向量 $r = P|Q = P|GT$, 再使用 FT 算法直接求解 $G_A Z = r$ 以获得补偿向量 Z , 接着计算共轭梯度方向, 最后沿着共轭梯度方向对 T 进行修正, 即 $T = T + \alpha Z$. 与 TPG-FTCG 算法相比, TPG-Sli 算法尽管速度有所提高, 但每次求出的 T 含有较大的误差, 导致最终的热分析精度有所下降.

4.4 TPG-Sli 的 GPU 并行加速及其优化

由于 TPG-Sli 算法的 GPU 并行加速仅存在于残差向量 r 的预条件求解算法步中, 即使用 FT 算法来进行 $G_A Z = r$ 求解. 基于 GPU 并行加速技术^[16], 本文编写了

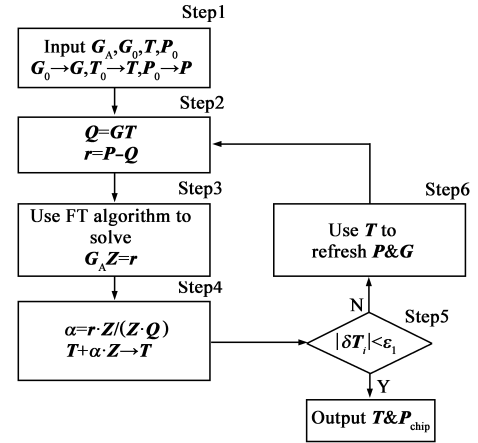


图7 TPG-Sli算法流程图

FT 算法的 GPU 并行加速版本 FT_GPU, 如图 8 所示. 由于 G_A 是一个不变的预条件矩阵, 所以我们在 CPU 端对 G_A 进行两次 FFT 变换后, 得到一个由位于对角线上的 $n_x \times n_y$ 个标准三对角子矩阵构成的新矩阵 \bar{G}_A . 为了使用追赶法求解 $\bar{G}_A \bar{Z} = \bar{r}$, 本文还对 \bar{G}_A 进行了 LU 分解, 得到一个预存的两对角矩阵 G_p , FT_GPU 在 Step 4 直接从 CPU 端读入 G_p , 用于求解 $(G_p)^T G_p \bar{Z} = \bar{r}$.

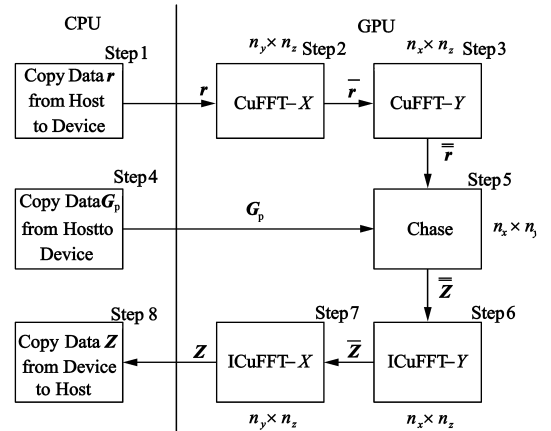


图8 GPU并行加速的FT_GPU算法流程图

下面结合具体的算法步骤对 FT_GPU 进行算法并行性分析:

Step2 与 Step3: 在 GPU 端调用 Cu-FFT 库函数先后沿 x 轴和 y 轴对 r 和 \bar{r} 进行快速傅里叶变换, 得到新的向量 \bar{r} 和 \bar{r} , 这两步变换需要 GPU 先后对 $n_z \times n_y$ 和 $n_z \times n_x$ 个矩阵向量乘法进行并行计算.

Step5: 在 GPU 端对 $(G_p)^T G_p \bar{Z} = \bar{r}$ 进行追赶法求解. 此步求解需要 GPU 对 $n_x \times n_y$ 个具有标准三对角子矩阵进行并行计算.

Step6 和 Step7: 在 GPU 端调用 Cu-FFT 库函数先后沿 y 轴和 x 轴对 \bar{Z} 和 Z 进行快速傅里叶反变换, 得到向量 Z 和 Z , 这两步变换需要 GPU 对 $n_z \times n_x$ 和 $n_z \times n_y$ 个

矩阵向量乘法进行并行计算。

从以上的并行计算分析可以看出:与已有 FT 算法^[4]仅简单地调用 Cu-FFT 库函数进行 GPU 加速相比,本文 FT_{-GPU}算法的并行计算全部在 GPU 端完成,仅需要和 CPU 进行 3 次数据传输,分别在 Step1 和 Step4 将 \mathbf{r} 和 \mathbf{G}_p 从 CPU 端拷入 GPU 端,在 Step8 将 \mathbf{Z} 从 GPU 端拷回 CPU 端,可以大量减少数据传输时间。

TPG-Sli 需要多次调用 FT 算法进行预条件的求解,将 FT 算法升级为 GPU 加速的 FT_{-GPU}算法后,TPG-Sli 就升级为其 GPU 加速版本 TPG-Sli_{-GPU}。由于预条件矩阵 \mathbf{G}_A 是不变的,所以对它进行 LU 分解所获得的两对角矩阵 \mathbf{G}_p 也是固定的,TPG-Sli_{-GPU}多次调用 FT_{-GPU}算法,在 Step4 多次将 \mathbf{G}_p 从 CPU 端拷贝至 GPU 端,需要耗费大量的数据传输时间。鉴于 \mathbf{G}_p 是一个两对角矩阵,仅包含少于 $2N$ 个非零元素 $N = n_x \times n_y \times n_z$,我们可以将 \mathbf{G}_p 直接预存到 GPU 端,以节约数据传输时间,本文将这种改进的 FT_{-GPU}算法称为 FT_{-iGPU},而将调用 FT_{-iGPU}算法的 TPG-Sli 称为 TPG-Sli_{-iGPU}。

5 实验结果

本次实验采用的是 Visual Studio 2010 中的 C/C++ 语言实现算法,所有实验均运行于 3.20GHz 主频的 Intel® Core (TM) i7-3930kCPU,内存容量为 32GB,显卡为 NVIDIA GTX 680,显存容量为 4GB。

所有实验都采用了 3 种不同的芯片功耗分布进行对比验证,为了保证测例的典型性与普适性,本文只在测例 1 中采用均匀的功耗分布,测例 2 和测例 3 分别采用了 AMD 四核 phenom 架构和 Intel 四核 nehalem 架构的功耗分布。3D 芯片共有两个器件层,即附着于剪薄硅衬底的互连层、3 层含 TSV 的剪薄硅衬底层与微压焊块层、附着于常规衬底的互连层、8 层常规厚度硅衬底共 14 层,每层共有 1001×1001 个节点,所以本测例的求解规模是 14M 个节点。三个测例的芯片总功耗均设为 100W,其中附着于常规衬底的器件层产生了 80W 功耗。

5.1 算法的精度对比

TPG-ICCG 和 TPG-FTCG 算法都具有内外两个循环,TPG-Sli 作为单循环的 TPG 分析算法,只具有外循环、而

无内循环。以上 3 种算法的判出条件为节点温度的改进量 $|\delta T_i| < \varepsilon_1 = 10^{-2}$,TPG-ICCG 和 TPG-FTCG 的内循环判出条件为残差 $|\mathbf{r}| < \varepsilon_2 = 10^{-6}$ 。为了评估算法精度,本文将采用残差限为 $\varepsilon_2 = 10^{-16}$ 的 TPG-ICCG 求解结果作为比较基准(golden),来比较以上 3 种算法在最大温度 T_{\max} 、平均温度 T_{ave} 以及芯片功耗 P_{chip} 三个参数上的分析误差。如表 2 示,3 种方法计算出的 T_{\max} 、 T_{ave} 以及 P_{chip} 几乎相同,绝对误差均在 10^{-2} 左右,相对误差均小于 0.1%,这表明以上 3 种算法都是精确的 TPG 热分析算法。

5.2 算法的时间复杂度对比

采用三个测例对文中涉及的 TPG-ICCG、TPG-FTCG、TPG-Sli_{-CPU}、TPG-Sli_{-GPU}、TPG-Sli_{-iGPU} 的算法运行时间进行比较,其运行时间依次标为 $T_1 \sim T_5$,如表 2 示,本文使用 T_{i-1}/T_i 来分析采用单项加速技术所带来的改进效果,使用 T_1/T_i 来表示:相较于 TPG-ICCG 算法的运行时间 T_1 ,本文提出的 4 种改进算法所带来的总体加速效果。通过对表 3 中数据的分析,可以得出如下结论:

(1) 对于 TPG-FTCG,由于其内循环迭代次数较 TPG-ICCG 算法大大的减少,其求解速度是 TPG-ICCG 的 16 倍左右。

(2) 与 TPG-FTCG 相比,TPG-Sli_{-CPU} 不含内循环,只用一个外循环进行 TPG 求解,因此可以有效降低算法的时间复杂度,其求解速度是 TPG-FTCG 的 3 倍左右,是 TPG-ICCG 的 56 倍左右。

(3) 与 TPG-Sli_{-CPU} 相比,TPG-Sli_{-GPU} 通过 GPU 并行加速来大幅度降低算法的运行时间,它的求解速度是 TPG-Sli_{-CPU} 的 2 倍左右,是 TPG-ICCG 的 90 倍。

(4) 与 TPG-Sli_{-GPU} 相比,由于将两对角矩阵 \mathbf{G}_p 预存在显存中,可以减少 CPU 端与 GPU 端的数据通信时间,使得 TPG-Sli_{-iGPU} 在时间复杂度上又有了进一步的降低。因此,其求解速度是 TPG-Sli_{-GPU} 的 1.2 倍,是 TPG-ICCG 的 120 倍左右。

(5) 如上所示,本文先后采用了 4 种加速技术成功地将 TPG 热分析效率提高了 120 倍,获得了满意的加速效果,表明为了最大限度地提高热分析的效率,必须从算法设计和计算平台这两方面对 TPG 算法进行改进。

表 2 不同算法的时间复杂度比

Algorithms	Case1			Case2			Case3		
	Time (s)	T_{i-1}/T_i	T_1/T_i	Time (s)	T_{i-1}/T_i	T_1/T_i	Time (s)	T_{i-1}/T_i	T_1/T_i
TPG-ICCG (T_1)	13690	Base	Base	12053	Base	Base	12879	Base	Base
TPG-FTCG (T_2)	625	21.9	21.9	804	15.0	15.0	760	16.9	16.9
TPG-Sli _{-CPU} (T_3)	202	3.1	67.8	215	3.7	56.1	304	2.5	42.4
TPG-Sli _{-GPU} (T_4)	129	1.6	106.1	143	1.5	84.3	136	2.2	94.7
TPG-Sli _{-iGPU} (T_5)	109	1.2	125.6	121	1.2	99.6	116	1.2	111.0

表 3 算法的精确度对比

	Parameters	Golden	TPG-ICCG		TPG-FTCG		TPG-Sli _{-CPU}	
			Value	Error	Value	Error	Value	Error
Case1	T_{\max} (°C)	78.196231	78.195622	-0.000609	78.180692	-0.015539	78.141363	-0.054868
	T_{ave} (°C)	78.101935	78.101519	-0.000416	78.093433	-0.008502	78.057004	-0.044931
	P_{chip} (W)	80.194368	80.170624	-0.023744	80.161273	-0.033095	80.160237	-0.034131
Case2	T_{\max} (°C)	81.481079	81.486108	0.005029	81.436856	-0.044223	81.422451	-0.058628
	T_{ave} (°C)	78.157804	78.162403	0.004599	78.117964	-0.03984	78.106626	-0.051178
	P_{chip} (W)	80.183141	80.169842	-0.013299	80.160871	-0.02227	80.163021	-0.02012
Case3	T_{\max} (°C)	82.426832	82.428755	0.001923	82.410445	-0.016387	82.264097	-0.162735
	T_{ave} (°C)	78.230618	78.234756	0.004138	78.219453	-0.011165	78.117496	-0.113122
	P_{chip} (W)	80.187963	80.184032	-0.003931	80.160837	-0.027126	80.158132	-0.029831

6 结论

本文对考虑 $T/P/G$ 之间相互影响的 TPG 热分析方法进行了系统地研究. 为了寻找 TPG 高效的求解算法, 提出了一种基于快速傅里叶变换为预条件的高效双循环算法 TPG-FTCG 及其改进的单循环算法 TPG-Sli, 并且编写了 TPG-Sli 算法 GPU 并行加速版本 TPG-Sli_{-GPU} 算法及其改进算法 TPG-Sli_{-iGPU}. 实验结果表明, TPG-Sli_{-iGPU} 算法不仅可以快速地进行 TPG 热分析, 能够对 3D 芯片进行精确而高效的热分析.

参考文献

- [1] Liao W P, et al. Temperature and supply voltage aware performance and power modeling at microarchitecture level [J]. IEEE Transactions on CAD, 2005, 24(7): 1042 – 1053.
- [2] Liu Z, Tan S X D, et al. Compact nonlinear thermal modeling of packaged integrated systems [A]. Proceedings of ASP-DAC [C]. Yokohama: IEEE, 2013. 157 – 162.
- [3] Wu X H, et al. Area minimization of power distribution network using efficient nonlinear programming techniques [J]. IEEE Transactions on CAD, 2004, 23(7): 1086 – 1094.
- [4] Daloukas K, et al. A parallel fast transform-based preconditioning approach for electrical-thermal co-simulation of power delivery networks [A]. Automation & Test in Europe Conference & Exhibition [C]. Grenoble: IEEE, 2013. 1689 – 1694.
- [5] Zhan Y, Goplen B, et al. Electro-thermal analysis and optimization techniques for nano-scale integrated circuits [A]. Proceedings of ASP-DAC [C]. Yokohama: IEEE, 2006. 219 – 222.
- [6] Li P, Pileggi L T, et al. IC thermal simulation and modeling via efficient multigrid-based approaches [J]. IEEE Transactions on CAD, 2006, 25(9): 319 – 326.
- [7] Wang T Y, Lee Y M, et al. 3D thermal-ADI—an efficient chip-level transient thermal simulator [A]. Proceedings of ISPD [C]. Monterey: IEEE, 2003. 10 – 17.
- [8] Luo Z Y, Tan S X D, et al. Localized statistical 3D thermal analysis considering electro-thermal coupling [A]. Proceedings of ISCAS [C]. Taipei: IEEE, 2009. 1289 – 1292.

- [9] Zhan Y, Sapatnekar S S. High efficiency Green function-based thermal simulation algorithms [J]. IEEE Transactions on CAD, 2007, 26(9): 1661 – 1675.
- [10] Oh D K, Chen C P, et al. 3DFFT: Thermal analysis of non-homogeneous IC using 3D FFT Green function method [A]. Proceedings of ISQED [C]. San Jose: IEEE, 2007. 567 – 573.
- [11] Qian H F, Sapatnekar S S, et al. Fast poisson solvers for thermal analysis [A]. Computer-Aided Design [C]. Monterey: IEEE, 2010. 698 – 702.
- [12] Lee Y M, Wu T H, et al. A hybrid numerical and analytical thermal simulator for 3-D ICs [A]. Proceedings of DATE [C]. Grenoble: IEEE, 2013. 1379 – 1384.
- [13] Yan J Q, Luo Z Y, et al. Accurate architecture-level thermal analysis methods for MPSoC with consideration for leakage power dependence on temperature [A]. Proceedings of ISQED [C]. Santa Clara: IEEE, 2013. 178 – 183.
- [14] Feng Z, Li P. Fast thermal analysis on GPU for 3D-ICs with integrated microchannel cooling [A]. Proceedings of ICCAD [C]. San Jose: IEEE, 2010. 551 – 555.
- [15] Huang K, Yang X, et al. Efficient electro-thermal co-analysis on CPU + GPU heterogeneous architecture [A]. Proceedings of ISQED [C]. Santa Clara: IEEE, 2012. 364 – 369.
- [16] Chen J, Wu X J, Cai R. Parallel processing for accelerated mean shift algorithm with GPU [J]. Journal of Computer-Aided Design & Computer Graphics, 2010, 03: 461 – 466.

作者简介



潘月斗 男, 1966 年出生, 博士, 副教授. 主要从事交流电动机智能控制理论, 研究及高速高精交流电动机驱动系统的计算机数字控制系统设计.
E-mail: ydpan@ustb.edu.cn

王嘉琪 男, 1989 年出生, 硕士研究生. 主要从事电热综合分析、高性能并行计算方面的研究工作.
E-mail: 3706157@qq.com