

固定边框的多电压布图规划算法

杜世民^{1,2}, 夏银水², 杨润萍¹, 钱利波²

(1. 宁波大学科学技术学院, 浙江宁波 315211; 2. 宁波大学信息科学与工程学院, 浙江宁波 315211)

摘要: 多电压设计是应对 SoC 功耗挑战的一种有效方法, 但会带来线长、面积等的开销。为减少线长、芯片的空白面积及提高速度, 提出了一种改进的固定边框多电压布图方法。对基于 NPE (Normalized Polish Expression) 表示的布图解, 采用形状曲线相加算法来计算其最优的布图实现, 并通过增量计算方法来减少计算 NPE 及多电压分配的时间。为使所得布图解满足给定的边框约束, 提出了一个考虑固定边框约束的目标函数, 并采用删除后插入 (Insertion after Delete, IAD) 算子对 SA 求得布图解进行后优化。实验结果表明, 和已有方法相比, 所提出方法在线长和空白面积率方面有较明显优势, 且所有电路在不同高宽比、不同电压岛数下均实现了极低的空白面积率 ($\ll 1\%$)。

关键词: 低功耗; 多电压; 布图规划; 固定边框

中图分类号: TP391

文献标识码: A

文章编号: 0372-2112 (2017)08-1873-09

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

DOI: 10.3969/j.issn.0372-2112.2017.08.010

Fixed-Outline Multi-voltage Floorplanning Algorithm

DU Shi-min^{1,2}, XIA Yin-shui², YANG Run-ping¹, QIAN Li-bo²

(1. Department of Information Science and Engineering, Ningbo University, Ningbo, Zhejiang 315212, China;

2. College of Science & Technology, Ningbo University, Ningbo, Zhejiang 315211, China)

Abstract: Multiple Supply Voltage (MSV) design is an effective ways to deal with SoC power challenge, but brings overhead on wirelength area, etc. To reduce wirelength, chip's dead space ratio and improve algorithm speed, an improved fixed outline multi-voltage floorplanning algorithm is proposed. For a given floorplan solution represented by NPE (Normalized Polish Expression), We use the shape curve adding algorithm to calculate its optimal floorplan implementation, and adopt an incremental calculation method in the process of NPE calculation and voltage assignment to reduce runtime. An improved objective function including fixed-outline constraint is proposed, and the IAD (Insertion After Delete) operator is used to post-optimize the obtained floorplan of SA. Experimental results show that compared with the existing methods, our algorithm has advantages on wirelength and dead space ratio, and all circuits achieve extremely low dead space ratio ($\ll 1\%$) under different aspect ratio and different voltage island number.

Key words: low power; multiple supply voltage; floorplanning; fixed-outline

1 引言

基于多供电电压 (Multiple Supply Voltage, MSV) 的设计方法是降低 SoC 功耗最为有效的方法之一^[1-4]。它是根据模块的性能需求给每个模块分配合适的供电电压来实现降低功耗的目的。同时, 为降低电源网络布线网络的复杂度, 在布图规划时需要将相同电压的模块汇聚在一起, 形成电压岛 (Voltage Islands, VI)。这使传统的物理设计流程变得更加复杂^[1,2]。

目前针对 MSV 问题的研究, 主要集中在物理设计

的布图^[1-3,5-7]和后布图阶段^[8,9]。其中, 在布图阶段主要有两类方法: 第一类是先采用动态规划^[1,3]、整数线性规划^[5]等方法给每个模块分配电压, 然后在布图时将相同电压的模块划分到同一电压岛, 如文献[3]和[5]。这类方法的优点是只需进行一次电压分配, 即可获得最优的功耗节省效果, 但它约束相同电压的模块放置在同一电压岛内, 会带来过大的线长开销; 第二类方法是将电压分配步骤嵌入到布图的迭代过程中, 即对每次扰动后产生的布图解, 采用动态规划等方法进行电压分配, 如文献[1,2]等。这类方法的优点是布

图迭代过程中对面积、线长和功耗等多个指标进行协同优化,因此可获得各个指标较优的布图.但缺点是需对每个布图解进行一次电压分配,使得算法耗时较长.随着层次化设计和 IP 核的广泛应用,固定边框(Fixed-Outline)约束已成为现代布图的必然要求^[10],而目前这两类方法大多未考虑芯片的边框约束,且产生的空白面积(Dead Space, DS)较大.

在电压岛的形状方面,目前主要有矩形电压岛^[1,3,5]和非矩形电压岛^[2,7]两种类型.文[2]提出对若干相邻的矩形电压岛进行“粘贴”来生成一种 L 形的电压岛的方法.文[7]提出了一种基于风险和治疗的策略(Hazard and Heal Strategy)来生成非规则形状的电压岛.虽然非矩形电压岛由于其形状上较大的自由度,一定程度上增大功耗的优化空间,但同时极大增加了后续电源布线网络设计的难度.

为降低由 MSV 带来的线长开销和芯片的空白面积率,提高算法的速度,本文提出了一种考虑固定边框约束的多电压布图方法.首先,采用形状曲线相加算法来计算正则波兰表达式(Normalized Polish Expression, NPE)^[11]表示的布图解,可以获得每个 NPE 最优的布图实现,从而加快模拟退火(Simulated Annealing, SA)算法的收敛速度;其次,在计算 NPE 和分配电压时,采用增量计算的方法来减少所需耗费的时间.最后,为使所得布图解满足给定的边框约束,提出了一个考虑固定边框约束的目标函数,并采用删除后插入(Insertion after Delete, IAD)算子对 SA 所得布图解进行后优化.

2 问题描述

给定如下输入信息:(1) 一个包含 n 个模块的集合 $B = \{b_1, b_2, \dots, b_n\}$ 及芯片的 m 个供电电压 $V_c = \{v_c^1, v_c^2, \dots, v_c^m\}$; (2) 每个模块 b_i 的面积 a_i 和高宽比范围 $[l_i, r_i]$, $1 \leq i \leq n$; (3) 所有模块之间的线网连接 $net_i, i = 1, 2, \dots, M$; (4) 每个模块 b_i 可行的供电电压集 $V_i \in V_c$ 及对应的功耗集 $P_i, 1 \leq i \leq n$; (5) 电压岛个数 K ; (6) 芯片的固定边框约束:包括高宽比 λ 和空白面积率 δ .

要求产生一个包含 K 个矩形电压岛的布图,使得芯片的功耗、线长和空白面积率降到最低,同时满足给定的固定边框约束.

3 所提出算法

为减少线长和空白面积率,加快算法的收敛,并满足指定的固定边框约束,提出了一种由两个阶段构成的改进多电压布图算法,如图 1 所示.

在阶段 I 中,采用 SA 算法搜索最优布图解,它由 3 个步骤构成.首先,对每次扰动后产生的新布图解,采用形状曲线相加算法计算出其所有可能的布图实现,然

后根据给定的边框约束,从形状曲线上找出最优的布图实现.其次,对所得到的布图解,根据给定的电压岛数 K ,采用动态规划方法进行电压分配并构建出 K 个矩形电压岛.然后,依据所得到的布图及电压分配结果计算该布图解的功耗、线长等指标,并采用改进的目标函数进行评估,如此迭代直到 SA 算法满足终止条件.

考虑到对软模块进行布图时,给定的边框约束一般较苛刻(如空白面积率 $\delta \leq 1\%$)^[12,13].当电路规模较大时,SA 算法搜索到的布图解不一定能满足这一约束.为此,在 SA 算法之后增加了一个后布图优化步骤,如图 1 阶段 II 所示.首先对 SA 布图解进行判断,判断其是否满足固定边框约束,若不满足,采用文[12]提出的 IAD 算子对其进行后优化.

上述流程中,采用形状曲线相加算法来计算布图解,可以获得每个布图解最优的布图实现,一方面降低了布图的 DS,另一方面也减少了 SA 算法的迭代次数.其次,对算法中最为耗时的两个步骤——布图解计算和电压分配采用增量计算的方法,有效减少了算法时间.最后,通过对目标函数的改进,使所求得布图解不仅可以满足固定边框约束,而且可获得更优的线长和功耗.

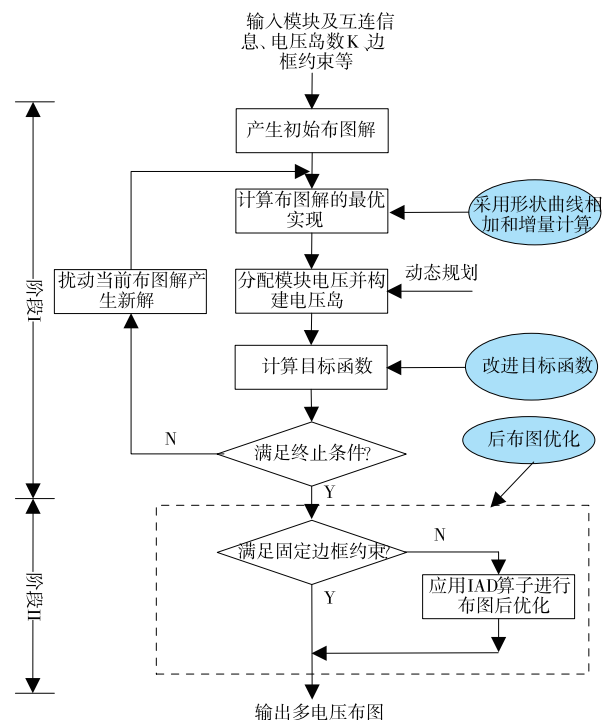


图1 所提出算法的流程图

3.1 布图解的计算

由于每个模块高宽尺寸在一定范围内可变,一个布图解(NPE)对应多个可能的布图实现,为加快 SA 算法的收敛,需找出每个 NPE 的最优布图实现.为此,先

将每个模块可行的尺寸范围表示为一条形状曲线,然后对每次扰动后产生的新布图解,应用形状曲线相加算法计算出整个 NPE 的形状曲线,最后根据给定的边框约束从中找出最优的布图实现。

由 2.1 节输入信息可知,模块 b_i 的高度 w_i 和宽度 h_i 满足以下式子:

$$\begin{cases} w_i * h_i = a_i \\ l_i \leq h_i/w_i \leq r_i \end{cases} \quad (1)$$

上式可以用图 2 中 A~B 之间的这段曲线来表示,称之为模块 b_i 的形状曲线,它的两个端点 A 和 B 称为该形状曲线的顶点。将每个模块表示为形状曲线后,它们之间的“*”或“+”运算就可转化为相应形状曲线之间的相互运算。当两个模块作“*”或“+”运算时,分别对应于各自的形状曲线进行横向或纵向相加。下面以一个包含 3 个模块的布图解“ $S_1 = 1\ 2 * 3 +$ ”为例,说明模块之间形状曲线相加的过程。表 1 给出了这 3 个模块的几何信息。

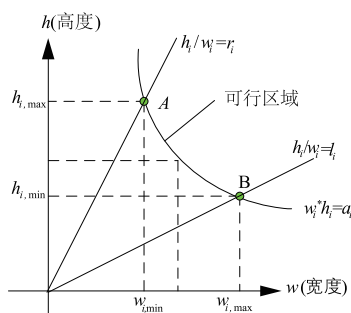


图2 模块 b_1 的形状曲线

表 1 S_1 中模块的几何信息

模块	面积	高宽比
b_1	8	[1/2 2]
b_2	12	[1/3 3]
b_3	18	[1/2 2]

图 3(a) 给出了 b_1 和 b_2 的形状曲线作横向相加的示意图。图中红色曲线 $A_1 - A_2$ 和 $B_1 - B_2$ 分别为 b_1 和 b_2 的形状曲线,曲线 $D_3 - D_2 - D_1$ 为它们“*”运算后生成的超模块 s_1 的形状曲线。其中, s_1 的形状曲线分两段构成:(a) 当 b_2 高度从 2 增加到 4 时,为使 b_1 与它水平组合后 DS 最小, b_1 的高度与 b_2 保持相同,此时, b_1 和 b_2 组合后产生的 DS 为 0,因此该区间内 s_1 的面积保持不变,如图 3(a) 中 $D_3 - D_2$ 段所示;(b) 当 b_2 高度从 4 变化到 6 时,为使 s_1 中的 DS 最小, b_1 的高度必定取最大值 4,此时 s_1 的高度取决于 b_2 ,其空白面积随 b_2 高度的增加而线性增长,如图 3(a) 中的 $D_2 - D_1$ 段所示。可见,对两个模块的形状曲线进行“*”运算时,只需简单地在它们的顶点处进行相应的横向相加就可以了。

类似地,可以求出 s_1 和 b_3 的形状曲线作纵向相加的结果,如图 3(b) 所示。图中,红色曲线 $C_1 - C_2$ 为 b_3 的形状曲线,曲线 $E_1 - E_2 - E_3 - E_4$ 为 s_1 和 b_3 “+”运算后生成的超模块 s_2 的形状曲线,即为布图解 S_1 的形状曲线,它记录了该布图解所有可能的尺寸。

计算出布图解 S_1 的形状曲线后,还要根据给定的高宽比约束 λ 找出 S_1 最优的布图实现。为此,通过原点分别作两条斜率为 λ 和 $1/\lambda$ 的直线,与 S_1 的形状曲线相交于 G_1 和 G_2 ,将两者中 DS 较小的一点作为该布图解的最优实现。例如在图 3(b) 中, G_1 的 DS 小于 G_2 ,故选择 G_1 作为 S_1 的最优实现。可见,应用形状曲线相加算法可以获得每个布图解的最优实现,从而可以减少 SA 算法的迭代次数。

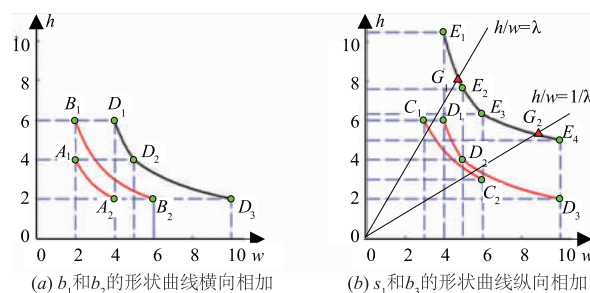


图3 布图解 S_1 的形状曲线计算过程

3.2 算法加速

3.2.1 布图解计算的加速

由 3.1 节可知,计算一个包含有 n 个模块的 NPE 的形状曲线,需进行 $n - 1$ 次形状曲线之间的相加运算。当电路规模较大时,计算 NPE 的形状曲线就成为制约算法速度的一个主要瓶颈。为减少这部分运算的时间,提出了一种布图解的增量计算方法。

对 NPE 表示的当前布图解,有以下 3 种扰动:(1) 随机交换两个相邻的操作数;(2) 随机选择一个连续的操作符链进行取反;(3) 随机交换两个相邻的操作数和操作符。

不难看出,对 NPE 的 3 种扰动都只是局部改变了 NPE 的结构,因此,在计算新解时,只需计算 NPE 中发生改变的部分。为此,对切分树中的每个运算符节点 p_i ($i = 0, 1, 2, \dots, n - 2$),引入一个布尔变量 u_i ,来记录该节点的运算符及下面的两个操作数扰动后是否有变化,如果未发生变化(用 $u_i = 0$ 表示),则不需要重新计算该节点的形状曲线。例如,在图 4 的布图解扰动示例中,仅 p_0 和 p_2 两个节点的运算符进行了取反,而节点 p_1, p_3 和 p_4 保持不变,因此,在计算新解时,仅重新计算节点 p_0 和 p_2 的形状曲线就可以了。

3.2.2 电压分配加速

多电压分配是整个算法最为耗时的步骤之一。本

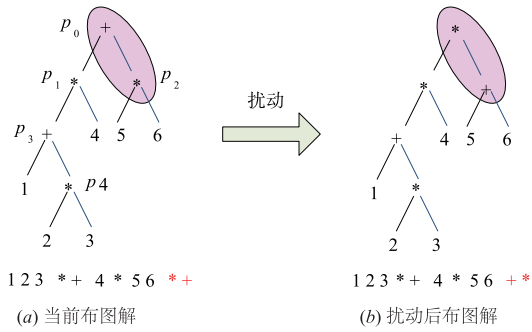


图4 对布局解的扰动

文采用文[1]提出的动态规划方法来分配模块电压,这种方法是在电压分配的同时构建电压岛,可以保证生成矩形形状的电位岛,从而减少后续供电网络设计的难度.它是从根节点开始,自上而下的对切分树中所有运算符节点进行搜索,找出每个节点在指定电压岛数时的最优电压分配方案.例如,对运算符节点 p_i ,若其电压岛数 k_i ,则它最多有 $(k_i + 1)$ 种不同的电压分配方案(其左右子树都可以有 $0 \sim k_i$ 个电压岛),将其中功耗最优结果保存到数组 $c[p_i][k_i]$ 中, $c[p_i][k_i]$ 用来记录 p_i 在 k_i 个电压岛时的最优功耗成本.为减少每次电压分配的时间,将前述增量计算方法推广到电压分配过程中,即对任意运算符节点 p_i ,若该节点对应的布尔变量 $u_i = 0$,则其功耗成本不需要重新计算.

3.3 目标函数改进

多电压设计的目的是为了降低芯片的功耗,并尽量减少线长、面积等的开销.已有方法中,对布局解 S_i 之相应布图 F_i ,一般采用以下函数来评估其性能^[1,2,5]:

$$\begin{cases} \text{Cost}(F_i) = \alpha A + \beta W + \gamma P \\ \alpha + \beta + \gamma = 1 \end{cases} \quad (2)$$

式中, A 、 W 和 P 分别表示所得布图的面积、线长和功耗, α 、 β 和 γ 为权重系数.在上述目标函数中,并未考虑固定边框约束,而现代布图规划实际上是一个固定边框的问题^[10,12,13].为此,将式(2)中的 A 用 R 来取代,改进后目标函数如下:

$$\text{Cost}(F_i) = \alpha R + \beta W + \gamma P \quad (3)$$

其中, R 的表达式为:

$$R = \max(W_c - W_f, 0) + \max(H_c - H_f, 0) / \lambda \quad (4)$$

式中 W_c 和 H_c 分别为所得布图的宽度和高度, W_f 和 H_f 为给定固定边框的宽度和高度,可由下式计算:

$$W_f = \sqrt{(1 + \delta)A_r / \lambda}, H_f = \sqrt{(1 + \delta)A_r \lambda} \quad (5)$$

式中 A_r 为所有模块的总面积; R 中第一、二项分别表示所得布图在宽度和高度方向上超出给定边框的大小,第二项中除以 λ 是为了均衡两个方向上的超出量.考虑到式(3)中的各项指标的绝对数值相差较大,对它们进行了归一化处理.显然, R 的大小不仅代表了所得布

图的宽度和高度与给定边框约束的之间的“距离”,而且还表征了布图的面积.与已有目标函数相比,改进后的目标函数有以下两个优点:(1)它实现了对面积、线长、功耗和固定边框约束的协同优化;(2)随着算法的迭代, R 的取值逐渐趋向于零,算法将更多的优化 W 和 P ,因此可以获得线长和功耗更优的布图解.

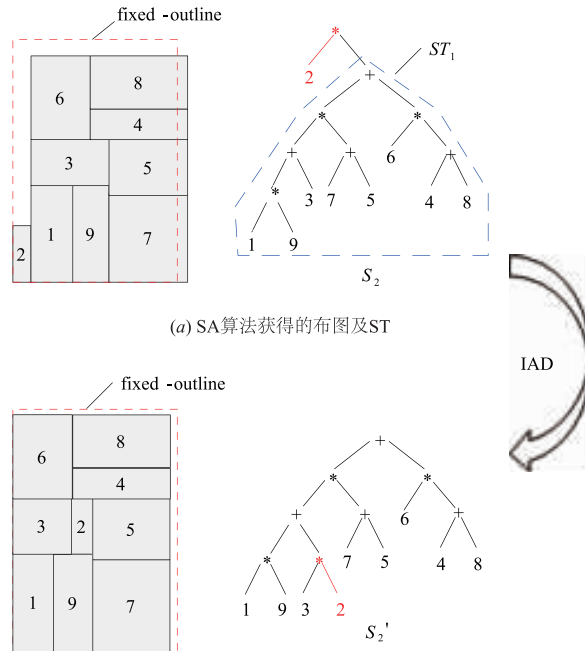


图5 应用IAD算子对SA布局解进行后优化

3.4 基于 IAD 算子的后布图优化

考察 SA 所求得解(设为 S_2)对应的布图 F_2 之所以不满足边框约束,是因为在 S_2 中个别运算符下的两个操作数尺寸相差较大,组合后产生了较大的 DS,从而使所得布图超出了指定的边框.例如在图 5(a)中,根节点(*)的左子树 b_2 的尺寸远小于右子树 ST_1 ,导致它们运算后在宽度上超出了固定边框(图中红色虚线框).这种尺寸相差较大的两个操作数之间的组合运算称为非优组合.

若能设法消除 S_2 中存在的非优组合,就有可能将其转化满足边框约束的布图解.从图 5(a)不难发现,若将模块 b_2 从 F_2 中删除,然后将其重新插入到 F_2 中的合适位置,并适当调整其它模块的尺寸,就有可能得到一个合法的布图,如图 5(b)所示.图 1 阶段 II 中的后布图优化方法正是基于这一思想.这种用于消除非优组合的操作称为删除后插入(IAD)算子.应用 IAD 算子来消除 SA 算法所求得布图解中的一对非优组合的步骤如下:

(1)找出 SA 布图解中的非优组合:若布图解中两个操作数组合后产生的 DS 大于 A_{th} ,即为非优组合;

(2) 删除非优组合中面积较小的操作数(设为 op_d);

(3) 将删除后的 op_d 重新插入到 SA 布图解的合适位置;

op_d 的插入位置采用枚举法来确定. 为了保持 SA 所求得解的电压分配结果, 将 op_d 和 S_2 中相同电压的操作数依次进行“*”和“+”运算, 直到找到满足边框约束的插入位置为止, 利用该方法消除一对非优组合最多需进行 $2 * (n_A - 1)$ 次插入操作, n_A 为与 op_d 相同电压的模块数.

4 实验结果

所提出算法已用 C 语言编程实现, 并在 3.7 GHz CPU、4.0GB RAM 的 PC 上对 GSRC 中 3 个规模最大的电路进行了实验. 电路中所有模块的高宽比范围为 $[0.3, 3]$, 供电电压集 $V_c = \{1.0v, 1.1v, 1.2v, 1.3v, 1.5v\}$, 其中 1.5v 为芯片工作电压, 允许的空白面积率 $\delta = 1\%$ [12,13]. 为与文[1~3,5]方法进行公平比较, 每个模块 b_i 可行的供电电压集 V_i 及模块 b_i 在电压 v_j 下的功耗 p_i^j 的计算方法均与它们相同 ($p_i^j = a_i \cdot (v_j)^2$). 线长采用半周长法 (HPWL: Half Perimeter Wire Length) 计算. 目标函数式(3)中的权重系数经过多次实验取值如下: $\alpha = 0.4, \beta = \gamma = 0.3$. 实验结果表明该组取值可以使所得布图满足给定的固定边框约束, 且可获得较好的功耗和线长.

表 2 列出了 $\lambda = 1$ 时本文方法与文[1,2]方法在功耗、功耗节省率、空白面积率、线长和运行时间上的比较, 表中 K 为电压岛数, 第一列电路名下的括号中数字为全部模块工作在 1.5v 时的功耗, 即最大功耗. 注意文[1]未考虑固定边框约束, 但其给出的多电压布图结果的高宽比接近于 1, 故将其与本文 $\lambda = 1$ 时的结果进行比较. 由表可见, 与文[1]相比, 所提出算法可减少 17.2% 的功耗和 30.7% 的线长, 平均空白面积率从 2.68% 减少到 0.08%, 且在算法速度上有较大程度提高. 这是因为所提出算法采用形状曲线相加算法来计算布图解, 可以获得最优的布图实现, 一方面降低了布图的 DS, 另一方面也减少了 SA 算法的迭代次数. 其次, 采用增量计算来计算布图解和进行电压分配, 提高了算法速度. 目标函数上所作的改进, 不仅使所得布图解满足了指定的边框约束, 而且获得更优的线长和功耗. 与文[2]相比, 所提出算法在功耗和线长上分别降低 3.6% 和 21.7%, 同时平均空白面积率有明显降低. 在算法速度上, 由于文[2]采用了基于分割的非随机搜索算法, 它仅对一棵通用的切分树进行搜索, 因此要优于本文和[1]这类基于 SA 的算法, 但它搜索的解空间不及 SA 算法广泛, 故在线长、功耗和空白面积率都较所提出算法有一定劣势. 图 6(a) 和 (b) 分别给出了 n200 和 n300 在高宽比 $\lambda = 1$ 时 5 个电压岛的布图结果.

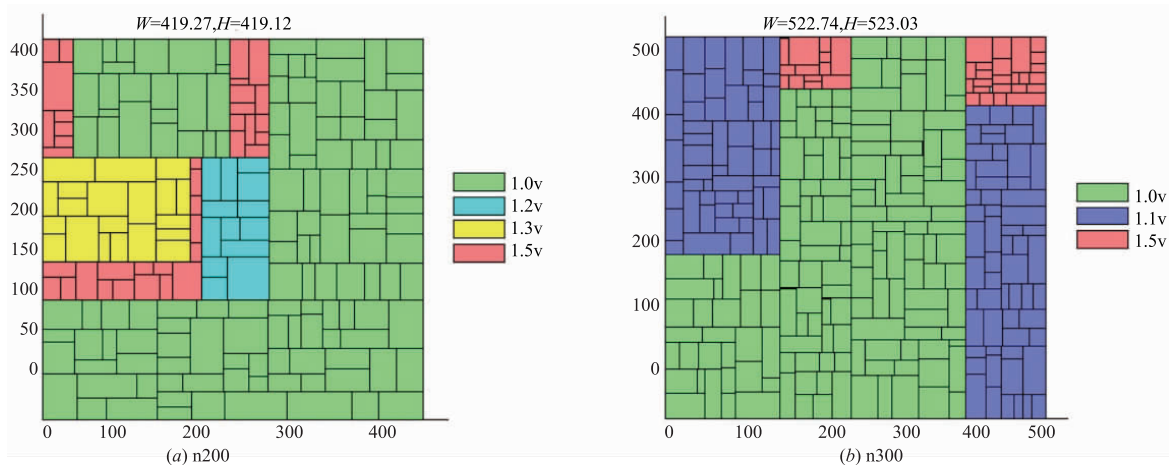


图6 n200和n300 5个电压岛时的多电压布图结果($\lambda=1$)

表 3 列出了 $\lambda = 1$ 时本文与文[3,5]实验结果的比较. 文[3]和[5]都是在布图之前给模块分配电压, 然后采用一个两阶段(电压岛级和芯片级)的布图过程来优化线长和面积. 和文[3]相比, 所提出算法平均减少了 12.8% 的线长, 平均空白面积率从 3.78% 减少到 0.08%, 且在算法速度上有较明显优势. 这是因为文[3]布图之前分配电压, 布图时需将相同电压的模块汇聚

在一起, 这给模块增加了额外的几何约束, 导致线长开销增大. 其次, 它分别采用基于 $B * -tree$ [14] 和 $SKB-tree$ [3] 的布图方法迭代的进行两阶段的布图, 使得算法较为耗时较长且 DS 较大. 文[5]仅进行一次两阶段的布图, 因此算法速度较快, 但同时导致线长开销增大, 平均较本文算法增加 21.5%, 平均空白面积率也有所增加. 文[5]中采用枚举法来获得最优的电压岛布图,

表 2 与文[1,2]的实验结果比较($\lambda = 1$)

电路	K	[1]					[2]					本文				
		功耗	功耗节省率 (%)	空白面积率 (%)	线长 (HPWL)	时间 (s)	功耗	功耗节省率 (%)	空白面积率 (%)	线长 (HPWL)	时间 (s)	功耗	功耗节省率 (%)	空白面积率 (%)	线长 (HPWL)	时间 (s)
N100 (403877)	1	332093	17.77	0.83	129970	409.72	311312	22.92	0.57	150665	3.48	273071	32.39	0.14	110601	35.79
	2	303367	24.89	0.60	134350	416.05	275436	31.8	0.82	145740	4.06	228669	43.38	0.04	115735	36.67
	3	256686	36.45	4.53	135760	475.06	230923	42.82	0.62	145590	3.77	219167	45.73	0.01	113936	37.15
	4	232262	42.49	1.60	143100	508.93	223960	44.55	0.62	145590	4.06	214457	46.90	0.12	115965	38.09
	5	231121	42.77	3.14	138820	572.86	217656	46.11	0.62	145590	4.06	208712	48.32	0.17	113733	38.09
	6	224879	44.32	3.36	143090	597.98	211684	47.59	0.62	145590	4.06	210984	47.76	0.12	122049	39.69
N200 (395316)	1	341132	13.71	6.06	311390	1762.17	231949	41.22	2.08	277209	7.25	275057	30.42	0.08	241505	153.85
	2	309471	21.72	1.49	304800	1770.16	220800	44.04	0.85	285184	7.25	244862	38.06	0.03	248384	164.97
	3	259555	34.34	3.15	308200	1814.28	214654	45.59	1.19	285220	7.25	241116	39.01	0.17	220360	161.08
	4	246987	37.52	5.70	320040	2080.43	210125	46.73	0.94	284737	7.54	228520	42.19	0.07	240871	163.66
	5	261257	33.91	3.43	307200	2036.82	206081	47.75	0.98	283693	8.12	217479	44.99	0.01	242220	165.60
	6	247342	37.43	0.89	326680	2058.11	202017	48.77	0.98	283693	7.83	214674	45.70	0.11	242309	167.02
n300 (614632)	1	541437	11.91	1.98	492890	3499.20	529433	13.86	2.11	462339	8.12	477353	22.34	0.07	361133	350.24
	2	494289	19.58	2.17	505650	3398.88	461417	24.93	2.11	462339	8.12	431473	29.80	0.06	374754	354.38
	3	436625	28.96	1.96	512980	3576.74	422318	31.29	2.11	462339	8.70	401239	34.72	0.02	395219	362.21
	4	457999	25.48	3.51	503830	3931.9	388222	36.84	2.11	462339	8.70	356427	42.01	0.07	378634	375.04
	5	379049	38.33	1.15	506360	3843.81	356810	41.95	2.11	462339	8.70	314404	48.85	0.06	384810	386.66
	6	386475	37.12	2.68	520480	4077.38	336185	45.30	2.11	462339	8.99	311286	49.35	0.05	374352	385.97
总平均		330113	30.48	2.68	319199	2046.14	291721	39.11	1.31	297363	6.67	281608	40.66	0.08	244254	189.79
归一化		1.172	0.750	34.359	1.307	10.781	1.036	0.962	16.879	1.217	0.035	1	1	1	1	1

随着电压岛数 K 的增加,布图解的数目将指数增长,因此仅适用于 K 较少的场合.在功耗上,文[3]和[5]在布图之前分配电压时,仅以功耗为优化目标,而所提出算法在布图过程中进行电压分配,同时考虑了功耗和模块的物理位置,因此在功耗上较文[3]和[5]增加 14.6%.

表 4 列出了本文算法在不同高宽比 λ 下所得到的实验结果的比较.由表可见,当高宽比 λ 从 1 增加到 2 和 3 时,电路的功耗基本维持不变;线长分别增加了 4.4% 和 12.8%,表明当高宽比增加时,会不可避免带来线长的增长;平均空白面积率较 $\lambda = 1$ 也有所增加,但所有电路在不同电压岛数下的空白面积率均远低于 1%,说明利用所提出算法解决固定边框下的多电压布图问题

是有效的.图 7 给出了 n300 分别在 $\lambda = 2$ 和 $\lambda = 3$ 时 5 个电压岛的布图结果.

表 5 列出了本文算法和文献[2]在高宽比 λ 分别取 2 和 3 时的比较结果,其中,表中的每列数据为各电路在电压岛数 K 从 1 增加到 6 时获得的平均值.由表可见,当高宽比 $\lambda = 2$ 时,所提出算法可较[2]减少 5.8% 的功耗和 16.6% 的线长,平均空白面积率从 1.38% 减少到 0.14%.当高宽比 $\lambda = 3$ 时,所提出算法在功耗和线长上可分别降低 6.9% 和 15.9%,平均空白面积率也明显低于文[2].可见,和已有算法相比,所提出算法在不同高宽比时在功耗、空白面积率和线长等方面可以获得更好的性能.

表 3 与文[3,5]的实验结果比较($\lambda = 1$)

电路	K	[3]					[5]					本文				
		功耗	功耗节省率 (%)	空白面积率 (%)	线长 (HPWL)	时间 (s)	功耗	功耗节省率 (%)	空白面积率 (%)	线长 (HPWL)	时间 (s)	功耗	功耗节省率 (%)	空白面积率 (%)	线长 (HPWL)	时间 (s)
n100	1	246722	38.91	3.83	127045	107.00	246722	38.91	0.06	129189	21.80	273071	32.39	0.142	110601	35.79
	2	220067	45.51	5.10	121637	83.00	220067	45.51	0.10	128463	20.80	228669	43.38	0.040	115735	36.67
	3	207131	48.71	1.84	123900	102.00	207131	48.71	0.15	133364	19.90	219167	45.73	0.010	113936	37.15
	4	203397	49.64	3.45	121116	80.00	203397	49.64	0.24	128729	20.00	214457	46.90	0.118	115965	38.09
n200	1	233437	40.95	4.03	270886	872.00	233437	40.95	0.23	287205	92.40	275057	30.42	0.083	241505	153.85
	2	219564	44.46	3.64	270856	882.00	216941	45.12	0.17	287263	80.20	244862	38.06	0.032	248384	164.97
	3	208144	47.35	4.36	272987	806.00	208144	47.35	0.33	293247	86.70	241116	39.01	0.175	220360	161.08
	4	205861	47.92	4.64	272936	694.00	205860	47.93	0.10	288518	88.20	228520	42.19	0.068	240871	163.66
n300	1	382534	37.76	2.87	425769	2440.00	382534	37.76	0.25	464475	174.40	477353	22.34	0.070	361133	350.24
	2	294120	52.15	4.06	410570	2327.00	294120	52.15	0.33	463345	179.40	431473	29.80	0.058	374754	354.38
平均		242098	45.34	3.78	241770	839.30	241835	45.40	0.20	260380	78.38	283375	37.02	0.080	214324	149.59
归一化比值		0.854	1.225	47.513	1.128	5.611	0.853	1.226	2.462	1.215	0.524	1	1	1	1	1

表 4 不同高宽比 λ 下的实验结果比较

电路	K	$\lambda = 1$					$\lambda = 2$					$\lambda = 3$				
		功耗	功耗节省率 (%)	空白面积率 (%)	线长 (HPWL)	时间 (s)	功耗	功耗节省率 (%)	空白面积率 (%)	线长 (HPWL)	时间 (s)	功耗	功耗节省率 (%)	空白面积率 (%)	线长 (HPWL)	时间 (s)
N100	1	273071	32.39	0.14	110601	35.79	253725	37.18	0.21	116214	35.42	267101	33.87	0.055	141644	35.71
	2	228669	43.38	0.04	115735	36.67	237924	41.09	0.16	125758	37.13	241660	40.17	0.312	125591	36.93
	3	219167	45.73	0.01	113936	37.15	228570	43.57	0.11	121149	37.13	218658	45.86	0.128	133607	38.66
	4	214457	46.90	0.12	115965	38.09	215389	46.67	0.21	119468	38.52	212855	47.30	0.017	123567	39.06
	5	208712	48.32	0.17	113733	38.09	208436	48.39	0.15	127434	38.75	212673	47.34	0.137	126874	39.56
	6	210984	47.76	0.12	122049	39.69	212269	47.44	0.18	117669	40.02	210970	47.76	0.145	138350	41.01
N200	1	275057	30.42	0.08	241505	153.85	293074	25.86	0.15	271792	152.78	280229	29.11	0.120	263925	152.04
	2	244862	38.06	0.03	248384	164.97	255502	35.37	0.00	244781	161.64	253426	35.89	0.105	246521	158.29
	3	241116	39.01	0.17	220360	161.08	238621	39.64	0.06	238019	163.58	238970	39.55	0.217	272593	164.79
	4	228520	42.19	0.07	240871	163.66	236755	40.11	0.13	240841	163.63	235080	40.53	0.142	265168	166.66
	5	217479	44.99	0.01	242220	165.60	226585	42.68	0.09	245943	168.73	228744	42.14	0.095	270687	168.72
	6	214674	45.70	0.11	242309	167.02	227213	42.52	0.14	236816	173.03	222291	43.77	0.160	260599	170.94
n300	1	477353	22.34	0.07	361133	350.24	474759	22.76	0.08	374063	348.00	469596	23.60	0.172	424909	356.32
	2	431473	29.80	0.06	374754	354.38	430034	30.03	0.14	416533	357.78	420978	31.51	0.061	435227	363.63
	3	401239	34.72	0.02	395219	362.21	389754	36.59	0.26	396503	370.86	347718	43.43	0.114	440205	363.72
	4	356427	42.01	0.07	378634	375.04	344563	43.94	0.10	423896	370.44	353951	42.41	0.285	444421	380.36
	5	314404	48.85	0.06	384810	386.66	319950	47.94	0.18	387272	413.73	339072	44.83	0.034	427127	385.83
	6	311286	49.35	0.05	374352	385.97	319485	48.02	0.10	386614	373.96	320958	47.78	0.055	417210	381.39
总平均		281608	40.66	0.08	244254	189.79	284034	39.99	0.14	255043	191.40	281940	40.38	0.13	275457	191.31
归一化		1	1	1	1	1	1.009	0.983	1.768	1.044	1.008	1.001	0.993	1.687	1.128	1.008

表 5 高宽比 $\lambda=2$ 和 3 时与文献[2]的比较

电路	$\lambda=2$						$\lambda=3$					
	文献[2]			本文			文献[2]			本文		
	总功耗	空白面积率(%)	线长(HPWL)	总功耗	空白面积率(%)	线长(HPWL)	总功耗	空白面积率(%)	线长(HPWL)	总功耗	空白面积率(%)	线长(HPWL)
N100	254968	1.11	146461	226052	0.17	121282	254968	0.65	159642	227320	0.13	131606
N200	222842	1.35	283289	246292	0.10	246365	224985	1.86	303119	243123	0.14	263249
N300	424046	1.69	462339	379758	0.14	397480	424046	1.24	494703	375379	0.12	431517
平均	300619	1.38	297363	284034	0.14	255043	301333	1.25	319155	281941	0.13	275457
归一化	1.058	9.857	1.166	1	1	1	1.069	9.615	1.159	1	1	1

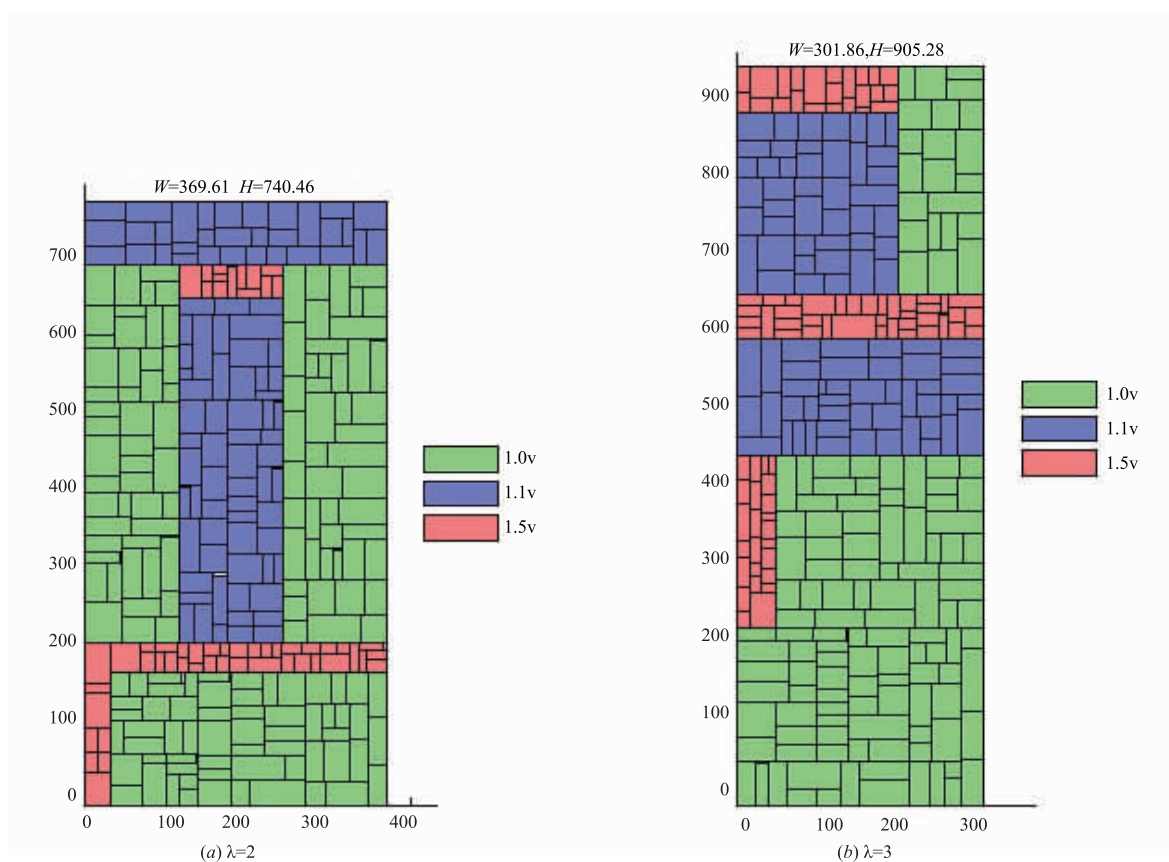


图7 n300不同高宽比下5个电压岛的多电压布图结果

5 结论

本文提出了一种考虑固定边框约束的改进多电压布图方法,它通过在布图解的计算、目标函数等方面的改进,来降低线长和芯片的空白面积率,并提高算法速度.在布图解的计算上,采用形状曲线来获得每个NPE最优的布图实现,同时通过增量计算方法减少计算NPE形状曲线和动态规划分配电压所耗费的时间,加快了算法的收敛.在目标函数上,提出了一个考虑固定边框约束的目标函数,并

采用IAD算子对SA所得布图解进行后布图优化,不仅可获得更优的线长和功耗,同时满足指定的固定边框约束.实验结果表明,和已有方法相比,所提出方法在线长和空白面积率具有优势.下一步将开展面向三维集成电路^[15]的多电压布图规划研究,通过降低功耗来改善三维芯片的热分布.

参考文献

- [1] Qiang Ma, Evangeline F Y Young. Multivoltage floorplan de-

- sign[J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2010, 29(4): 607 – 617.
- [2] Chu Zhufei, Xia Yinshui, Wang Lunyao, Wang Jian. Efficient nonrectangular shaped voltage island aware floorplanning with nonrandomized searching engine [J]. Microelectronics Journal, 2014, 45(4): 382 – 393.
- [3] Jai-Ming Lin, Zhi-Xiong Hung. SKB-Tree: A fixed-outline driven representation for modern floorplanning problems[J]. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2012, 20(3): 473 – 484.
- [4] Qian Zaichen, Evangeline F. Y. Young. Multi-voltage floorplan design with optimal voltage assignment [A]. Proceedings of the 2009 international Symposium on Physical Design [C]. New York: ACM, 2009. 13 – 18.
- [5] 杜世民, 夏银水, 储著飞, 杨润萍. 电压岛驱动的多级布图规划优化算法[J]. 西安电子科技大学学报(自然科学版), 2015, 42(6): 184 – 190.
Du Shi-min, Xia Yin-shui, Chu Zhu-fei, Huang Cheng, Yang Run-ping. Voltage island-driven multilevel floorplanning optimization algorithm [J]. Journal of Xidian University, 2015, 42(6): 184 – 190. (in Chinese)
- [6] Aminollah Mahabadi, Ahmad Khonsari, Behnam Khodabandloo, et al. Critical path-aware voltage island partitioning and floorplanning for hard real-time embedded systems [J]. Integration, the VLSI Journal, 2015, 48(1): 21 – 35.
- [7] Zhen Meng, Song Chen, Lu Huang. Irregularly shaped voltage islands generation with hazard and heal strategy [A]. IEEE 16th International Symposium on Quality Electronic Design [C]. California: IEEE, 2015. 1 – 4.
- [8] Wan-Ping Lee, Hung-Yi Liu, Yao-Wen Chang. An ILP algorithm for post-floorplanning voltage-island generation [A]. Proceedings of the 2007 IEEE/ACM International Conference on Computer-Aided Design [C]. New Jersey: IEEE Press, 2007. 650 – 655.
- [9] Wai-Kei Mak, Jr-Wei Chen. Voltage island generation under performance requirement for SoC designs [A]. Proceedings of the Asia and South Pacific Design Automation Conference [C]. Washington, DC: IEEE Computer Society, 2007. 798 – 803.
- [10] S N. Adya, I L Markov. Fixed-outline Floorplanning: enabling hierarchical design [J]. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2003, 11(6): 1120 – 1135.
- [11] D F Wong, C L Liu. A new algorithm for floorplan design [A]. Proceedings of the 23rd ACM/IEEE Design Automation Conference [C]. Las Vegas: IEEE Computer Society Press, 1986. 101 – 107.
- [12] 杜世民, 夏银水, 储著飞, 等. 面向软模块的稳定固定边框布图规划算法[J]. 电子与信息学报, 2014, 36(5): 1258 – 1265.
Du Shi-min, Xia Yin-shui, Chu Zhu-fei, et al. A stable fixed-outline floorplanning algorithm for soft module [J]. Journal of Electronics & Information Technology, 2014, 36(5): 1258 – 1265. (in Chinese)
- [13] Yan J Z, Chu Chris. DeFer: deferred decision making enabled fixed-outline floorplanning algorithm [J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2010, 29(3): 367 – 381.
- [14] P Sivaranjani, A Senthil Kumar. Thermal-aware non-slicing VLSI floorplanning using a smart decision-making PSO-GA based hybrid algorithm [J]. Circuits, Systems, and Signal Processing, 2015, 34(11): 3521 – 3542.
- [15] 王伟, 张欢, 方芳, 等. 2TF: 一种协同考虑过硅通孔和热量的三维芯片布图规划算法[J]. 电子学报, 2012, 40(5): 971 – 976.
WANG Wei, ZHANG Huan, FANG Fang, et al. TF: A collaborative considered TSV and thermal floorplanning algorithm for three-dimensional chip [J]. Acta Electronica Sinica, 2012, 40(5): 971 – 976. (in Chinese)

作者简介



杜世民 男, 1976 年生于浙江东阳. 宁波大学科学技术学院副教授, 博士. 研究方向为集成电路设计自动化、低功耗 SoC 设计.
E-mail: dushimin@nbu.edu.cn



夏银水(通信作者) 男, 1963 年生于浙江余姚. 宁波大学信息科学与工程学院教授, 博士生导师, 研究方向为集成电路综合与优化、低功耗 SoC 和 IP 核设计.
E-mail: xiayinshui@nbu.edu.cn