

基于拓扑分析的多层通道布线算法*

卢永江

(浙江大学 电气工程学院, 浙江 杭州 310027)

摘要: 本文提出了一种基于拓扑分析的多层通道布线算法。算法把整个布线过程分成拓扑分层和物理布线两个部分。拓扑分层利用线段交叠图及模拟退火算法解决线段分层及通孔最少化问题, 物理布线过程引入虚拟走线道解决交叉问题, 再利用轮廓线跟踪的方法来决定最终确定各线段的布线位置。算法还解决了多层布线分层的管脚约束问题和相邻约束问题。实验结果表明, 这是一种有效的方法。

关键词: 多层通道布线; 线段交叠图; 通孔最小化; 模拟退火算法; 轮廓线跟踪

中图分类号: 7410D; 512D **文献标识码:** A

1 引言

通常 VLSI 多层通道布线采用的分层原则是 H—V 严格分层原则, 这种分层原则对于提高布通率很有效, 但引入了大量通孔。通孔降低了芯片的可靠性和成品率, 增加了制造成本, 还增大了芯片面积, 对电路性能也会产生一定影响。因此, 减少芯片中通孔的数目是十分具有现实意义的问题。后来发展的一些通道布线算法打破了这种严格分层的限制, 但往往把注意力放在 100% 布通率和最少走线道上。虽然有一些通孔最小化算法^[1-5], 但由于先布线再压缩通孔或规定一些过死的限制条件, 所以难以取得较优的结果; 也有些算法^[6], 也提到了先分层再布线的思想, 但是它必须把整个线网布在同一层, 而且对不可同层线网仍然采用传统的算法进行布线, 无法使通道在整体上通孔最小化。另外, 一般的多层通道布线算法都是针对固定布线层数, 如果布线层变化就得重新设计算法, 不适合当今布线层越来越多的趋势, 而且也没有充分利用各布线层来降低通道高度。还有大多数通道算法都是基于有网格的, 无法考虑变线宽问题。

本文提出了一种无网格通道布线算法, 不但在通孔最小化问题上优于一般的优化算法。而且能充分利用各布线层资源, 降低通道高度。设线网的接点排列在矩形通道的上下边界上, 通道的左右边界允许有引出线, 这里假定都是二接点线网, 如果是多接点线网则将其分解成多个二接点线网。

2 拓扑分层

2.1 通道拓扑模型的概念

在通道拓扑模型^[7]中, 线网看作是连接两个接点的一条直线, 因此通道可以认为是一组交叉的直线。

定义 1: 若两个线网交叉, 必有一个交点, 该交点称为拓扑交叉点。可以简单认为该交点的坐标是分别连接两个线网的直线的交点, 同时规定线网的每个管脚端也看作一个拓扑交叉点。

定义 2: 对于每一交叉线网, 在交叉点附近存在候选通孔, 为简化起见, 初始假定候选通孔位置即为线网相邻交叉点或交叉点与引出点之间的中点。如图 1(a)中的黑点所示。

定义 3: 连接两个候选通孔(或管脚端)的连线称为线段。如图 1(a)中 a_i, b_i, c_i 所示。

定义 4: 给定通道, 定义图 $G_s = (V_s, E_v \cup E_c)$ 为它的线段 - 相交图 Segment-Crossing Graph^[3] (SCG 图), 如图 1(b)所示, 其中顶点集 V 中每个顶点代表一条线段。边集分成两个子集, 称通孔边集 (via-edges set) E_v 与相交边集 (cross-edges set) E_c 。 E_v 中的边 (通孔边) 表示该边两个连接顶点所对应的线段由

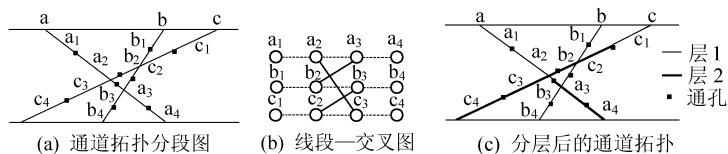


图1 通道模型

* 收稿日期: 2002-11-04 修订日期: 2003-01-04

候选通孔相连接,在图 1(b)中用点虚线表示; E_c 中的边(相交边)表示该边两个连接顶点所对应的线段是相互相交的,在图 1(b)中用实线表示。

一个多层通道布线拓扑可用 SCG 图表示,而可以将 k 层拓扑分层问题描述为 SCG 图特定的 k 着色问题。若对一个 k 层布线,每一层布线用一种颜色表示, k 层布线的分层问题就相对于在该 k 层布线的线段-相交图上用 k 种颜色对每个顶点进行着色。当一通孔边连接的两个顶点着同一种颜色时,这个候选通孔就不会成为通孔;而一相交边连接的两个顶点不能着同一种颜色。因此,线段的分层问题就等价于对线段-相交图的每个顶点着上合适颜色,使得在满足 E_c 边连接的两个顶点着不同的颜色的前提下,使 E_v 边连接的两顶点尽可能着相同的颜色。

在 SCG 图中作如下定义:赋予顶点 u 的颜色称为顶点 u 的已着色(active color),记作 $a(u)$,已赋颜色的顶点称作已着色点(active vertex)。

2.2 物理约束对着色的限制

对某些引脚只能出现在固定层导致与它相连的线段也被限制布在该层及某些线网必须布在某些特定的布线层的情况,由于与该线段相对应的顶点则必须着以与特定布线层相应的颜色,并且这些颜色在着色过程中固定不变,将此类顶点 u 称为静态限制着色顶点(static restrictedly-colored vertex), u 所着色的颜色的集合称为 u 的静态限制色(static restricted color)集 $s(u)$ 。SCG 着色应满足:

$$a(u) \in s(u) \text{ if } u \text{ is a static restrictedly-colored vertex} \quad (1)$$

对于只能制造连接相邻金属层通孔的工艺,则在 SCG 中顶点 u 的着色应满足:

$$a(v) - a(u) = 1 \text{ for all } v \in \{v | (u,v) \in E_v\} \text{ for all } u \text{ in } G_{SCG} \quad (2)$$

为了表征以上约束,引入以下定义:

所有满足顶点 u 的约束(1)(2)的颜色称为顶点 u 的可着色集,记作 $r(u)$ 。

2.3 SCG 中的通孔数目的计算

在可以制造连接非相邻布线层通孔(stack via)的工艺中,实际制造一个 stack via 相当于制造多个通孔。制造一个连接布线层 1 和布线层 3 的 stack via 相当于制造两个通孔。而在 SCG 中,一个 stack via 是用一个候选通孔来表示的。所以,应该用下式来计算某个候选通孔 v_m^n 实际所代表的通孔数目:

$$v_m^n = \max_{u \in V_c(v_m^n)} a(u) - \min_{u \in V_c(v_m^n)} a(u) \quad (3)$$

2.4 基于模拟退火分层算法

由于拓扑分层存在多种约束,难以找到一个确定的搜索原则,可用随机搜索的方法,我们采用的是模拟退火方法。在满足约束条件的情况下,根据贪婪原则产生一种初始解。在算法参数的选择上,参数选择 $T_0=20$;温度下降函数为

$$T_k = a * T_{k-1}, a = 0.99。$$

算法从原解生成新解的方法如图 2 所示。

由于在对 SCG 图中的顶点 u 重新着色后,通孔数目的变化只与 u 的相容邻接顶点集 $A(u)$ 有关,而与 SCG 图的其它顶点无关,所以在 generate(S)以后我们不用计算新老解(两个 SCG 图)下的通孔总数,而通过计算 $A(u)$ 内的通孔数目的变化来计算通孔减少数目 ΔC 。

3 虚拟走线道分配

走线道的分配是将已确定布线层的线段分配到通道的走线道上。由于各线网的线段只有在一定的顺序布线才不会相交。根据几何定理,两个线段相交只有一个交点,而且只有在交点前后线段的上下顺序才会交换。因此设想有一个扫描线从左到右逐个经过每个交叉点,故引入交叉点拓扑排序图来表示线段进行布线的顺序。

```

Procedure generate(S)
/* S is a colored SCG */
(1) Select randomly a vertex  $u$  with the number of
the set  $r(u)-a(u)$  large than 0 of from SCG;
(2) Uncolor  $u$ ;
(3) Select randomly a color  $a'(u)$  form the set
 $r(u)-a(u)$  and assign it to vertex  $u$ ;
(4) Renew
(5) Return S with a new color  $a'(u)$  assigned to  $u$ .

```

图 2 原解生成新解的算法

定义 5: 定义这样两种偏序关系, 对于所有线网的管脚端, 按其水平坐标从左到右建立偏序关系 $R1$ 如图 3(a)中交点序列 $\{c'', a', b'', b', a'', c'\}$; 对于每个线网上的拓扑交叉点根据其水平坐标从左到右的顺序建立偏序关系 $R2$, 若线网是垂直线网可安从下到上建立偏序关系, 如图 3(a)中交点序列 $\{a', \hat{a}, \hat{a}, a''\}$ 及 $\{c'', \hat{a}, \hat{a}, c'\}$ 等。

定义 6: 给定一个通道, 定义拓扑交叉点约束图

Topological_Intersect - Constraint Graph (TCG 图) $G=(V, E)$, 如图 3(b)其中顶点集 V 中的每个顶点表示一个拓扑交叉点; 边集 E 中的边表示有偏序关系 $R1$ 或 $R2$ 。显然图 3(b)是一无回路的有向图, 而且必然存在一个偏序序列。

算法引入虚拟走线道, 可以认为虚拟走线道是无宽度的, 它只是说明走线道在通道中的上下顺序。在叙述算法前, 有必要说明一下: 初始认为通道没有任何走线道, 所有走线道都是在布线过程中生成; 在处理通孔和线段交换位置时, 采取最早原则, 同时必须在水平方向上满足设计规则要求。由于在算法中需要了解某线段与其它线段之间的情况, 因此建立一个动态数组的, 在该数组中记录所有正在处理的线段的信息, 数组中顺序表示当前线段在通道中的相对位置。

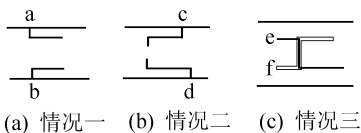


图 4 交叉点处理图例

对交叉点的处理分为如下几种情况:

1) 若拓扑交叉点是左引脚端, 则在靠近引脚端所在的边处插入一个走线道, 将该引脚端引到该走线道, 如图 4(a)所示。

2) 若拓扑交叉点是右引脚端, 则在靠近引脚端所在的边处插入一个走线道, 将正在处理的线段引到该轨道, 同时将右引脚端引到该走

线道, 如图 4(b)所示。

3) 若拓扑交叉点是非引脚类型的, 延伸其相应前一个线段, 并在合适位置将两个线网的线段交换位置, 布到对应的走线道上, 如图 4(c)所示。

算法见图 5。

- (1) Create TCG;
- (2) Select a node from TCG with its predecessor node is 0, and deal with the node;
- (3) Update the information of the dynamic Array;
- (4) remove the node from TCG;
- (5) Repeat step2 to step 4 until all node is done;

图 5 虚拟走线道分配算法

4 实际布线

在虚拟走线道分配时, 保证了同一走线道上的线不可能在同一层上出现相交, 而且根据走线道的相对位置可决定线网在同一布线层上的上下关系, 因此采用轮廓线跟踪^[8]的方法来最终确定走线的位置。算法从下到上处理每个走线道上的线段 (或者通孔), 将线段 (或者通孔) 布到其所在层当前轮廓线的外围, 与轮廓线保持一定的距离, 以满足设计规则要求, 然后更新轮廓线, 处理下一个线段 (或者通孔), 直至所有的走线道都已经处理。

5 算法分析

本算法的计算量主要在线网着色的模拟退火算法上。模拟退火算法中, 每次迭代的时间复杂度为 $O(n)$, 每个温度总共要经过 $O(n^2)$ 次的迭代, 而温度下降的次数是固定的, 因而模拟退火算法时间复杂度为 $O(n^3)$ 。一般来讲平均每个线网的拓扑交叉点是有限, 可认为是一个常数 k , 因此本算法的时间复杂度为 $O(kn^3)$, 其中 n 为线网总数。

6 试验结果

本算法在 VC++6.0 下开发实现。对文献中的一些典型例子进行了布线, 并将所得结果与两种通孔优化算法^[4,5]的结果进行了比较。比较的结果列于表 1 中。由表 1 可见, 在 3 层布线的情况下, 使用本算法, 通孔数得到了大幅度的减少, 走线道比 HVH 模式略有增加, 但比 VHV 模式要少。

表 1 3 层通道布线结果及比较

| 布线实例来源 | | Ex.3b 文献[1] | Ex.4b 文献[1] | Ex.1 文献 [1] | Ex.5 文献 [1] |
|-----------------|------|----------------|----------------|-------------------|-------------------|
| 优化前通孔数 | | 107 | 186 | 57 | 150 |
| 最大通道密度 | | 17 | 17 | 12 | 20 |
| 文献[5]算法结果 | 通孔数 | 75 | 34 | -- | 93 |
| | 走线道数 | 10 | 7 | -- | 10 |
| 文献[4]算法结果 (HVH) | 通孔数 | 74 | -- | -- | 90 |
| | 走线道数 | 10 | -- | -- | 10 |
| 文献[4]算法结果 (VHV) | 通孔数 | 54 | -- | -- | 80 |
| | 走线道数 | 17 | -- | -- | 20 |
| 本文算法结果 | | 37 | 11 | 68 | 52 |

传统通道布线算法先 H—V 布线后进行通孔最少化优化的限制，而首先对通道进行拓扑分析，利用线段 - 相交图和模拟退火算法使通孔最少化，然后在进行物理布线。算法不但实现了通孔最小化的优化目标，而且能充分利用各布线层资源，降低通道高度。算法还解决了多层布线分层的管脚约束问题和相邻约束问题。另外算法是基于无网

格的，能够解决线网变线宽变间距布线问题。

参考文献：

[1] Yoshimura Takeshi, Kuh E S. Efficient algorithms for channel routing [J]. *IEEE Trans CAD*, 1982, 1(1): 180-190.

[2] D Braun, *et al.* Chanmeleon; A New Multi-Layer Channel Router [A]. Proc. of 23rd Design Automation Conference [C]. 1986, 495-502.

[3] K Takahashi, T Watanabe. A Heuristic Algorithm to Solve Constrained Via Minimization for Three-Layer Routing Problem [A]. in Proc.ISCAS'98 [C].

[4] The Khe-Sing, *et al.* A layer-out modification approach to via minimization [J]. *IEEE Trans CAD*, 1991, 10(4): 536-541.

[5] 胡卫明, 严晓浪. 性能驱动的多层布线有约束分层及其神经网络求解方法 [J]. *半导体学报*, 1999, 20(12): 1115-1121.

[6] 甘骏人, 王小港, 罗志宏. 一种基于通孔数最小化的多层通道布线 [J]. *计算机学报*. 2002, 25(8): 830-836.

[7] Haruyama S E. Topological channel routing [J]. *IEEE Trans. on Computer-Aided Design of Integrated Circuits*, 1992-10, 11(10): 1177-1197.

[8] D N Deutsch. Compacted channel routing [A]. In Proceedings of the IEEE International Conference on Computer Aided Design [C]. 1985, 223-225.

作者简介：卢永江（1976-），2002 年获杭州电子工学院计算机应用硕士，现为浙江大学电气工程博士研究生，研究方向集成电路版图自动化。

A Multi-Layer Channel Routing Based on Topological Analysis

LU Yong-jiang

(Dept of Information & Electronic Engineering, Zhejiang University, Hangzhou 310012, China)

Abstract：An innovative approach to solve the channel routing problem is proposed. The whole problem is broken into two topics: topological layer definition and physical wire layout. For the first topic, Segment-Crossing Graphics and Simulated Annealing Approach are used to solve the problem of via minimization and layer assignment. For the second topic, virtual routing is introduced in the procedure of wiring to solve the problem of crossing, and then the contour following approach is used to determine positions of all wire-segments. Using the proposed algorithm, the problem of layer assignment in multi-layer wiring under the constraint of pin-out and the constraint of neighborhood can be solved. This algorithm has been verified to be effective through experiment.

Key words: multi-layer channel routing; Segment-Crossing Graph; via minimization; Simulated Annealing; Contour following