

基于线网类型分析的过点分配算法*

李 江 洪先龙 乔长阁 蔡懿慈

(清华大学计算机系 北京 100084)

摘要 过点分配是布线过程中位于总体布线和详细布线之间的一个步骤,它用于在详细布线之前确定线网在总体布线单元边界上的物理位置.本文提出一种新的过点分配的启发式算法,它基于总体布线后的线网类型的分析,考虑了总体布线单元中障碍对过点分配的影响,把整个布线区域的过点分配问题转化成一系列的单列(行)总体布线单元边界的过点分配问题,然后利用经典的线性分配方法进行过点的分配.

EEACC: 7410D, 5120

1 引言

集成电路复杂度的增加使得目前的大部分布线系统把布线的过程分成总体布线,过点分配和详细布线三个阶段来完成.总体布线布线区域划分成一个二维的总体布线单元(GRC:Global Routing Cell)阵列,根据穿过每个 GRC 边界的线网的拥挤情况,把所有线网的连接转化成 GRC 之间的连接,产生整个布线区域的粗略总体布线结果;详细布线的任务是完成 GRC 内部线网的物理连接.由于总体布线阶段并没有确定线网在 GRC 边界的具体位置,因此在详细布线前需要确定线网在 GRC 边界的物理位置(即过点),这个过程就是过点分配.

过点分配问题(CPA:Cross Point Assignment)在有通道的门阵列和标准单元的布线中又称为走线道分配(Feed-through Assignment).已经提出了许多关于走线道分配的算法:Hong^[1]把总体布线阵列的每一列的走线道分配问题看成一个垂直通道布线问题,把布线区域的走线道分配问题转化成为许多列的垂直通道布线问题,同时证明了列的分配顺序对分配结果的无关性.Okamoto^[2]则利用最小多货物流的方法,解决了分配过程中线网分配顺序约束和边界分配顺序约束的问题.这两种方法都只适用于有通道的结构,而无法应用于无通道的结构. Levin^[3]提出了一种可适应于有通道和无通道两种结构的过点分配方法.但是它采用自下而上,自左而右的顺序布线方法,并只考虑连线端(Pin)、障碍和已分配的过点对当

* 国家攻关项目资助课题

李 江 男,1971 年生,原清华大学计算机系研究生,现在中国银行云南省分行工作

洪先龙 男,1940 年生,教授,目前从事计算机设计自动化的教学和科研工作

乔长阁 男,1967 年生,副教授,目前从事集成电路计算机辅助设计的研究工作

1996 年 6 月 24 日收到初稿,1996 年 11 月 18 日收到修改稿

前待分配过点的分配的影响,因而分配结果不理想. Kao^[4]提出了带有总体再布线的过点分配算法,并且考虑了垂直约束对分配结果的影响.但是由于其采用的还是顺序布线的方法,因而同样不可避免地存在分配顺序约束对分配结果的影响.同时,由于采用总体再布线的方法,因而花费的时间较长.

通过对已有算法的分析不难看出,过点分配要获得较优的结果,最主要的是要克服分配过程中中线网分配的顺序约束的问题,同时也要考虑 GRC 中障碍对过点分配的影响.基于这样的考虑,本文提出了一种新的过点分配方法,通过把整个布线区域的过点分配转化为一系列的单列(行)的 GRC 边界的过点分配问题,并在综合考虑障碍对布线结果影响的前提下,采用并行的方法进行过点分配.

2 问题描述

在总体布线完成以后,布线区域被划分成为 $M \times N$ 的二维 GRC 阵列. 我们有以下三个假设:(1) 采用两层布线模式,其中一层布水平线,另一层布垂直线;(2) 总体布线的结果存在;(3) GRC 中的障碍为宏单元中的已布线段和所有线网的连线端. 下面是几项定义:

$C_{i,j}$: 表示第 i 行,第 j 列的 GRC, $0 \leq i < m$, $0 \leq j < n$

$HB_{i,j}$: 表示水平边界,它位于 $C_{i,j}$ 和 $C_{i+1,j}$ 之间, $0 \leq i < m-1$, $0 \leq j < n$

$VB_{i,j}$: 表示垂直边界,它位于 $C_{i,j}$ 和 $C_{i,j+1}$ 之间, $0 \leq i < m$, $0 \leq j < n-1$

我们把整个布线区域的过点分配分两个步骤来完成:(1)、进行水平边界 HB 的过点分配:取出一列 GRC,对这列中所有落入 GRC 水平边界 HB 上的线网同时构造费用矩阵.这样做的目的是避免了在这列中线网的分配顺序对分配结果的影响,即保证了在一列中所有的线网是并行分配过点的.(2)、在水平边界 HB 的过点分配完成后,进行垂直边界 VB 的过点分配.垂直边界 VB 的过点分配采用类似水平边界过点分配的方法,行与行之间顺序地进行分配,在行内是并行分配.通过这样的方法,整个布线区域的过点分配问题就转化成了一系列的单列(行)GRC 边界的过点分配问题.

每一个 GRC 的过点分配问题可以描述为一个线性分配问题:在一个 GRC 边界上有 n 个穿过边界的总体布线线网, $N = \{N_1, N_2, N_3, \dots, N_n\}$, 有 m 个布线通道(Track), $T = \{T_1, T_2, T_3, \dots, T_m\}$. 这样形成了 $n \times m$ 的二维费用矩阵 C , 矩阵的元素 $C(a, b)$ 表示线网 N_a 分配到布线通道 T_b 的费用. 其中 $1 \leq a \leq n$, $1 \leq b \leq m$.

定义函数 $f: x \rightarrow f(x)$, $x \in N$, $f(x) \in T$. 对于任意的 x_1, x_2 , 如果 $x_1 \neq x_2$, 那么 $f(x_1) \neq f(x_2)$. 有了这样的定义,就可以把单个 GRC 边界的过点分配问题转化为求关于矩阵 C 的

最小费用问题,即求 $\sum_{i=1}^n (N_i, f(N_i))$ 的最小值. 这可以用经典的线性分配算法^[5]来解决.

3 线网分类

不失一般性,我们把 GRC 中的线网段定义为至少有一端落在 GRC 边界上的线网. 以水平边界为例,我们根据线网段与 GRC 边界的关系以及与 GRC 中的引线端(或单元引脚)的关系把线网段分为十二类.

(1) 单端线网段：

类型 1 定义为线网的一端落在 $C_{i,j}$ 的水平边界 $HB_{i,j}$ 或 $HB_{i-1,j}$ 上, 而另一端是 $C_{i,j}$ 中的一个连线端. 如图 1(a) 所示.

(2) 两端线网段：

类型 2, 类型 3 定义为线网的一端落在 $C_{i,j}$ 的水平边界 $HB_{i,j}$ 或 $HB_{i-1,j}$ 上, 而另一端分别落在 $C_{i,j}$ 的垂直边界 $VB_{i,j-1}$ 或 $VB_{i,j}$ 上, $C_{i,j}$ 中无该线网的连线端. 如图 1(b)、图 1(c) 所示.

类型 4 定义为线网的一端落在 $C_{i,j}$ 的水平边界 $HB_{i,j}$ 上, 而另一端落在 $C_{i,j}$ 的水平边界 $HB_{i-1,j}$ 上, $C_{i,j}$ 无该线网的连线端. 如图 1(d) 所示.

类型 5 定义为线网的一端落在 $C_{i,j}$ 的水平边界 $HB_{i,j}$ 或 $HB_{i-1,j}$ 上, 而另一端落在 $C_{i,j}$ 的垂直边界 $VB_{i,j-1}$ 或 $VB_{i,j}$ 上, $C_{i,j}$ 有该线网的连线端. 如图 1(e) 所示.

类型 6 定义为线网的一端落在 $C_{i,j}$ 的水平边界 $HB_{i,j}$ 上, 而另一端落在 $C_{i,j}$ 的水平边界 $HB_{i-1,j}$ 上, $C_{i,j}$ 中有该线网的连线端. 如图 1(f) 所示.

(3) 三端线网段：

类型 7 定义为线网的一端落在 $C_{i,j}$ 的水平边界 $HB_{i,j}$ 或 $HB_{i-1,j}$ 上, 而另两端分别落在 $C_{i,j}$ 的垂直边界 $VB_{i,j-1}$ 和 $VB_{i,j}$ 上, $C_{i,j}$ 中无该线网的连线端. 如图 1(g) 所示.

类型 8 定义为线网的一端落在 $C_{i,j}$ 的水平边界 $HB_{i,j}$ 或 $HB_{i-1,j}$ 上, 而另两端分别落在 $C_{i,j}$ 的垂直边界 $VB_{i,j-1}$ 和 $VB_{i,j}$ 上, $C_{i,j}$ 中有该线网的连线端. 如图 1(h) 所示.

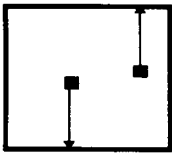


图 1(a) 类型 1

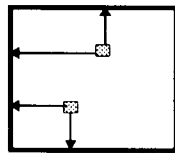


图 1(b) 类型 2

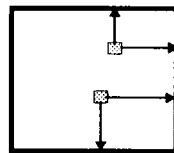


图 1(c) 类型 3

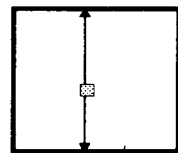


图 1(d) 类型 4

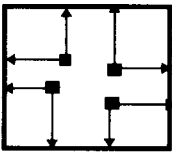


图 1(e) 类型 5

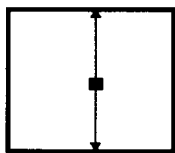


图 1(f) 类型 6

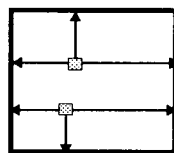


图 1(g) 类型 7

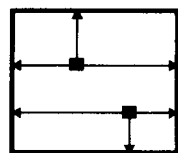


图 1(h) 类型 8

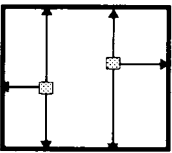


图 1(i) 类型 9

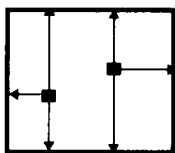


图 1(j) 类型 10

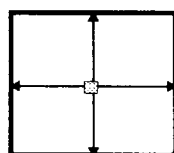


图 1(k) 类型 11

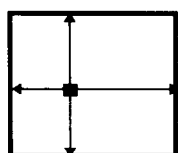


图 1(l) 类型 12

■ : 物理连接点 □ : 假想连接点

图 1 十二种线网段类型

类型 9 定义为线网的两端分别落在 $C_{i,j}$ 的水平边界 $HB_{i,j}$ 和 $HB_{i-1,j}$ 上, 而另一端落在 $C_{i,j}$ 的垂直边界 $VB_{i,j-1}$ 或 $VB_{i,j}$ 上, $C_{i,j}$ 中无该线网的连线端. 如图 1(i) 所示.

类型 10 定义为线网的两端分别落在 $C_{i,j}$ 的水平边界 $HB_{i,j}$ 和 $HB_{i-1,j}$ 上, 而另一端落在 $C_{i,j}$ 的垂直边界 $VB_{i,j-1}$ 或 $VB_{i,j}$ 上, $C_{i,j}$ 中有该线网的连线端. 如图 1(j) 所示.

(4) 四端线网段:

类型 11 定义为线网的四端分别落在 $C_{i,j}$ 的水平边界 $HB_{i,j}$ 和 $HB_{i-1,j}$ 以及垂直边界 $VB_{i,j-1}$ 和 $VB_{i,j}$ 上, $C_{i,j}$ 中无该线网的连线端. 如图 1(k) 所示.

类型 12 定义为线网的四端分别落在 $C_{i,j}$ 的水平边界 $HB_{i,j}$ 和 $HB_{i-1,j}$ 以及垂直边界 $VB_{i,j-1}$ 和 $VB_{i,j}$ 上, $C_{i,j}$ 中有该线网的连线端. 如图 1(l) 所示.

(5) 线网段类型的转化:

以上十二类线网段类型中, 类型 1、类型 2、类型 3 和类型 4 为基本类型, 其余类型可以转化为这四种基本类型的组合; 类型 5 为类型 1 的组合; 类型 6 为类型 1 的组合; 类型 7 为类型 2、类型 3 的组合; 类型 8 为类型 1 的组合; 类型 9 为类型 4、类型 2(类型 3) 的组合; 类型 10 为类型 1 的组合; 类型 11 为类型 4、类型 3 和类型 2 的组合; 类型 12 为类型 1 的组合.

4 费用函数的构造

由于所有的线网类型是基本类型 1 到 4 的组合, 所以只需定义这四种基本类型的费用函数就可以得到所有线网的费用. 在总的费用函数中, 线网类型的费用表示为 $COST_1$; 由于 GRC 中存在障碍, 因此在费用函数中要考虑障碍对过点分配的影响, 这部分费用表示为 $COST_2$; 在分配过程中, 既可以先进行水平边界过点分配再进行垂直边界过点分配, 也可以先进行垂直边界过点的分配, 再进行水平边界过点的分配. 如果采用前一种方法, 则水平边界已分配的过点会对垂直边界待分配的过点的分配产生垂直约束作用; 如果采用后一种方法, 则垂直边界已分配的过点会对水平边界待分配的过点的分配产生水平约束作用, 所以在费用函数中也要考虑垂直(或水平)约束对过点分配的影响, 这部分费用表示为 $COST_3$. 因此总的费用可以表示为:

$$COST = \beta_1 \times COST_1 + \beta_2 \times COST_2 + \beta_3 \times COST_3$$

其中 $\beta_1, \beta_2, \beta_3$ 为常量.

4.1 $COST_1$ 的计算

四种基本类型的 $COST_1$ 费用可以表示为

$$\text{类型 1: } COST_1 = |X - X_0|;$$

$$\text{类型 2: } COST_1 = |X|;$$

$$\text{类型 3: } COST_1 = |W - X|;$$

$$\text{类型 4: } COST_1 = \text{同一列(行)GRC 的下一个对应边界的费用.}$$

其中 W 为待分配的 GRC 边界宽度, 即布线通道数; X_0 为当前的 GRC 中待分配线网的连接端坐标; X 为待分配的过点的坐标, $0 < X \leq W$.

其它类型线网费用的构造可以根据其转化的组合类型的费用取和完成.

4.2 $COST_2$ 的计算:

对于类型 1, 在线网的连线端与待分配边界之间的所有障碍都对 $COST_2$ 有作用, x, y

代表障碍的坐标. 其函数表示为:

$$COST_2 = 1/|x-1|; (\text{垂直边界分配时})$$

$$COST_2 = 1/|y-1|; (\text{水平边界分配时})$$

对于类型 2,3 而言,在进行垂直边界分配时,水平边界的过点已经确定,水平边界的过点与待分配的垂直边界之间的障碍对 $COST_2$ 有作用. x 代表障碍的坐标,其函数表示为: $COST_2 = 1/|x-1|$;

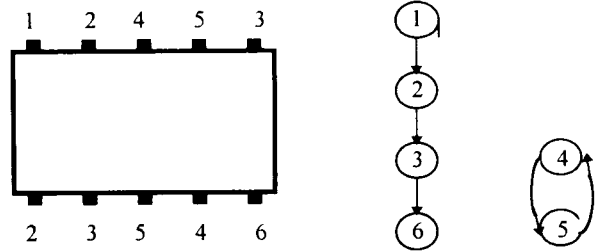
在进行水平边界分配时,由于垂直边界的过点没有确定,可以认为距离待分配边界一定距离内的障碍对 $COST_2$ 有影响(在本算法中,此距离认为是 $C_{i,j}$ 垂直高度的一半), y 代表该距离的坐标,其函数表达为: $COST_2 = 1/|y-1|$.

对于类型 4,由于 $C_{i,j}$ 中无待分配的线网的连线端,也没有穿过 $C_{i,j}$ 垂直边界 $VB_{i,j-1}$ 和 $VB_{i,j}$ 的线网,只是穿过 $C_{i,j}$ 水平边界 $HB_{i,j}$ 和 $HB_{i-1,j}$,所以把 $C_{i,j}$ 中的障碍投影在待分配的水平边界上,如果有障碍投影,则对费用有作用,其函数表示为:

$$COST_2 = \begin{cases} \beta_i & \text{边界的过点有障碍投影,其中 } \beta_i \text{ 为常量} \\ 0 & \text{边界的过点无障碍投影} \end{cases}$$

4.3 $COST_3$ 的计算:

如果我们先进行水平边界,然后再进行垂直边界的过点分配,则由于水平边界的过点已经确定,在进行垂直边界的过点分配时,已分配的水平方向的过点必然对垂直边界过点的分配产生约束作用,所以把这种约束作为费用的一部分,即 $COST_3$. 如图 2(a) 所示的 GRC,在进行垂直边界的过点分配时,用图 2(b) 表示已分配过点的垂直约束图 VCG. 对于形成闭环的垂直约束图的节点,我们把它合并成为一个节点考虑. 那么,费用函数表示为:



(a) 水平方向过点分配后的垂直约束 (b) 垂直约束图

图 2 过点的垂直约束

其中 $Height(N)$ 表示当前线网 N 所在的 VCG 的高度; $Level(N)$ 表示线网 N 在 VCG 中的高度.

$$COST_3 = Level(N) - (Height(N) + 1)/2$$

其中 $Height(N)$ 表示当前线网 N 所在的 VCG 的高度; $Level(N)$ 表示线网 N 在 VCG 中的高度.

5 算法介绍与测试结果

我们的算法是这样实现过点的分配的. 首先读入总体布线的结果,根据总体布线产生的 GRC 阵列信息,重构总体布线图. 然后,进行水平边界 HB 的过点分配. 取出未分配的一列 GRC,对这列中所有 GRC 内的线网同时构造相对水平边界 HB 的费用矩阵,用线性分配的方法进行水平边界 HB 的过点分配. 水平边界 HB 的过点分配完成后,最后再进行垂直边界 VB 的过点分配. 垂直边界 VB 的过点分配采用类似水平边界的过点分配的方法,只是在费用考虑时有垂直约束的问题. 通过这样的方法,整个布线区域的过点分配问题就转化成了一系列的单列(行)的 GRC 边界的过点分配问题.

在单列总体布线单元边界过点分配时,对于当前列的所有穿过总体布线单元边界的线

网,首先构造当前列中所穿过的总体布线单元边界的线网的局部线网.然后依次取出一条局部线网,构造这条线网分配在当前列中其所穿过的所有总体布线单元边界的过点的候选位置的费用.在构造费用时,为了消除线网分配顺序对分配结果的影响,在分配当前的线网时,不考虑其它局部线网的存在,只考虑当前局部线网的拓扑结构和总体布线单元中障碍的影响,对其所通过的所有总体布线单元的边界构造费用.虽然在构造过点的分配费用时,采用了顺序构造的方法,但是由于构造每条线网的费用矩阵时,没有考虑其它线网的影响,所以线网的分配顺序不会对其在边界的候选过点位置的费用产生影响.

在当前列中的所有局部线网在总体布线单元的边界上的所有过点的费用矩阵构成之后,逐次取出一个总体布线单元,对其水平边界费用矩阵,求解它的最小费用匹配,获得这列中的所有线网在总体布线单元水平边界的过点位置.

本算法已在 SUN SPARC 20 工作站 UNIX 操作系统上用 C 语言实现.我们测试了一些实例,测试结果如表 1 所示.

表 1 测试结果

名 称	DJK	XCLK	C2	C5	C7
单元数目/个	159	175	811	1887	2465
线网数目/个	215	264	745	1764	2356
行 数	11	11	17	18	20
列 数	11	11	17	18	20
垂 直 通 道 数	14	32	42	41	18
占用的垂直通道数	7	16	26	26	9
水 平 通 道 数	28	31	37	72	69
占用的水平通道数	11	16	21	20	21
CPU 时间/s	0.55	0.82	6.36	16.44	31.21

从上表所示的结果可以看出,所有的电路测试实例中,水平通道的最大占用率最大的是电路 C2 中的 56.8%,水平通道的最大占用率最小的是电路 C5 中的 27.8%;垂直通道的最大占用率最大的是电路 C5 中的 63.4%,垂直通道的最大占用率最小的是电路 DJK 中的 50%.这样的水平通道和垂直通道的最大占用率,可以更大地保证详细布线的成功完成.另外,与以往的走线道分配算法^[1]相比,算法的运行时间大大缩短,说明算法速度非常快.

6 结 论

本文提出的基于总体布线后线网类型分析的过点分配算法,考虑了总体布线单元中障碍对过点分配的影响,克服了分配过程中线网分配顺序的约束.它可以适用于单元中含有障碍的双层或多层布线门海、标准单元布图中,也可用于有任意布线区域的 BBL 布图系统中.实验表明,它是一种非常有效、通用和高速的过点分配算法.

参 考 文 献

- [1] X. L. Hong, J. Huang, C. K. Cheng et al., FARM: An Efficient Feed-Through Pin Assignment Algorithm, Proc. of 29th Design Automation Conference, 1992, 530~535.

- [2] T. Okamoto, M. Ishikawa and T. Fujita, A New Feed-through Assignment Algorithm Based on Flow Model, Proc. of ICCAD, 1993, 775~778.
- [3] E. Levin, Improved Port Assignment on Channel Boundaries, EDAC, 1992, 333~337.
- [4] W. C. Kao and T. M. Parng, IEEE Trans. CAD, 1995, 14(3) : 337~348.
- [5] R. E. Burkard and U. Derigs, Assignment and Matching Problem: Solution Methods with Fortran-Programs, New York : Springer-Verlag, 1980.

Cross Point Assignment Algorithm Based on Analysis of Net Type

Li Jiang , Hong Xianlong , Qiao Changge and Cai Yici

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

Received 24 June 1996, revised manuscript received 18 November 1996

Abstract Cross Point Assignment (CPA) is a routing process in the routing system between global routing and detailed routing. The aim of this process is to decide the precise physical position on the boundaries of the global routing cells (GRCs). This paper gives a new heuristic algorithm for the CPA, which bases on the analysis of the net type after global routing and takes into account the effect of the obstacles inside the GRC during the process of the CPA. This algorithm translates the CPA for the whole routing area into a sequence of single column (row) cross point assignment, and does the CPA using the classic linear assignment algorithm.

EEACC: 7410D, 5120