

一种有效地综合两种分级设计 方法的 BBL 布局算法

俞明永* 薄建国* 洪先龙**

连永君*** 庄文君*

(*中国科学院北京软件实验室)

(**清华大学)

(***北京大学)

1989年9月11日收到

本文提出了一个新的 Building Block (积木块) 布局算法。在布局中采用多目标形状的方法, 将自底向上 (bottom-up) 及自顶向下 (top-down) 两种分级设计方法有效的结合起来, 从而综合这两种设计方法的优点, 得到令人满意的布局结果。本布局算法已在 UNIX 操作系统支持下的 SUN、VAX-785、HP、GPX 上用 C 语言实现, 实验结果表明本布局法优于 BEAR 系统的布局算法。本布局系统是正在开发的国家 IC CAD 三级系统 (PANDA 系统) 的一个子系统。

主题词 布局, Building Block 分级设计

一、引言

随着集成电路技术的发展, 积木块 (Building Block) 模式的布局越来越受到人们的重视。很多文章对此作出了深入的研究^[1-6]。但从分级设计角度看, 这些文章不是采用自底向上的设计方法就是采用自顶向下的设计方法。但自底向上的设计方法缺乏总体优化的考虑, 自顶向下的设计方法缺乏优化方向的引导。最近 BEAR^[7] 布局系统^[8] 提出了向前看 (Lookahead) 的方法。这种方法的思想是: 在 Top-down 过程中, 每进行一次矩形分割 (Rectangle Dissection)^[9] 就根据给定的向前看的深度及搜索范围系数, 判断当前矩形分割对以下 d 层 (d 即为 lookahead 的深度) 各群的影响好坏。这种方法虽有谋求使 top-down 过程方向优化的思想, 但有以下缺点:

1. 从根本上看, 这种方法是穷举法。因为当搜索范围系数 ≥ 2 时 (该系数可以从零到无穷大), 基本上对所有组合情况均进行了搜索, 运行时间难以忍受。

2. 在 Lookahead 过程中, 如果向前看到的群的形状不固定或不准确, 则 Lookahead 根本就没有意义。

针对以上情况, 本文提出了一种基于多目标形状的 Building Block 布局方法。这种方法在 bottom-up 过程中对各群产生几个目标形状, 以给 top-down 过程优化的方向性指导。

本布局系统采用布局中同时进行总体布线(也可同时进行布线区面积估计及嵌入)及多目标形状的方法。所采用的总体布线及布线区面积估计及嵌入的方法基本与 BEAR 系统一致^[7,8],这里不在叙述。本文将主要介绍多目标形状的布局法。

二、多目标形状的产生

多目标形状由 bottom-up 过程产生,目的是给 top-down 过程以指导。

我们以树的方式来描述一个电路的 bottom-up 结群过程,叶子级节点对应于输入数据中的 Block, 度数为 m 的结点表明该点所对应的群为 $m(m \leq 4)$ 个子群。目标形状的产生与结群并行进行。首先进行叶子级结群,此时群内各孩子的目标形状只有一个(假定输入数据中各单元形状及 pin 位置均固定)。此群的 n 个 ($n \leq 5$) 目标形状根据其孩子的目标形状产生,选择前 n 个最优或满足条件的 n 个目标形状即是。然后逐级向上进行。在这向上过程中,群内各孩子的目标形状已有 n 个 因此,相应群的目标形状是其孩子的多目标形状的不同组合产生。到根级,整个结群过程及多目标形状产生过程结束。

三、多目标形状布局法

1. 基本思想

提出多个目标形状的布局法,是为了有效地将 top-down 及 bottom-up 两个分级设计过程结合起来,综合两者优点,克服两者缺点。另外,目标形状要多个而不是单个的原因: 如果目标形状只有一个,那么在给 top-down 过程指导时, top-down 矩形分割所产生的子矩形就必须向其唯一的相应目标形状看齐。而矩形分割中各子矩形是相互制约的,不可能均向各自唯一的的目标形状看齐,从而导致最后布局结果不理想。采用多个目标形状就可以很好的克服这一缺点。

2. 过程描述

下面我们来看一下 bottom-up 及 top-down 两种设计过程是怎样被结合起来的。

假定经过 bottom-up 过程后,形成了结群树 T 及各群的目标形状。

接下来是 top-down 过程,根据用户要求的长宽比及 bottom-up 过程所产生的面积作为参考,可求出根结点 R 所对应的矩形 RR 。结点 R 有 3 个孩子 A, B 及 C , 下面的任务

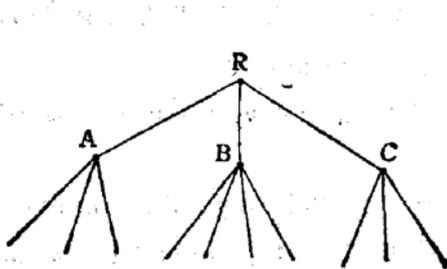


图1 结群树T

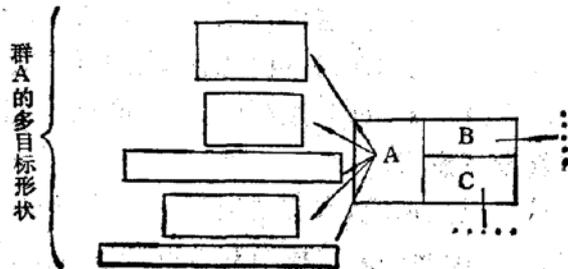


图2 多目标形状与矩形分割

是如何将矩形分割为 3 个子矩形。

矩形分割思想可简要描述如下:

对于有 m 个孩子的群, 对应有 $K(m)$ 种模板 (template)^[9], 每种模板表示了含有 m 个子矩形的拓扑矩形分割。根据矩形 (如矩形 RR) 的已知尺寸及子矩形所对应的孩子号, 即可计算出各子矩形几何尺寸^[8,10]。而各子矩形又有其对应的孩子所具有的多目标形状。见图 2。一种分割产生后, 就可以根据多目标形状, 计算出目标函数 f_{pen} 的值。

$f_{pen} = \text{连接惩罚量 (adj-cost)}^{[8]} + \text{几何惩罚量 (geo-cost)}$

$$\text{geo-cost} = \sum_{r=1}^n (\text{MIN}[(1 + R\text{Tri}) * \text{MIN}(|R\text{Xr} - T\text{Xri}| * Y + |R\text{Yr} - T\text{Yri}| * X, \\ |R\text{Xr} - T\text{Yri}| * Y + |R\text{Yr} - T\text{Xri}| * X)] (i = 1, m))$$

式中 $R\text{Tri}$ 为第 r 个子矩形所对应孩子的第 i 个目标形状的面积冗余比, $R\text{Xr}$ 为第 r 个子矩形宽, $R\text{Yr}$ 为第 r 个子矩形长, $T\text{Xri}$ 为第 r 个子矩形所对应的第 i 个目标形状的宽, $T\text{Yri}$ 为第 r 个子矩形所对应的第 i 个目标形状的长, X 为根结点矩形宽, Y 为根结点矩形长。

对于全部的模板 (template), f_{pen} 最小或小于一阈值的矩形分割即为要求的 top-down 过程中关于该结点的布局。

在整个 top-down 过程中, 各结点的处理基本相同, 只是在叶子级时的目标形状只有一个。

至此, 我们可以发现, 在 top-down 的整个过程中, 对各结点的布局是有目标的, 其目标是所产生的各子矩形尺寸尽可能符合相应孩子的某个目标形状。这就是说, top-down 过程是有优化方向引导的, 其优化方向是由 bottom-up 过程提供的。因此, 本布局法将二种分级设计方法有效地结合起来了。

3. 算法

本多目标形状布局算法可简单描述如下:

Placement (clustering tree, target shapes)

```
{
for (each node in the clustering tree in top-down manner)
{
for (each template)
{
for (each room assignment)
{
for (each block orientation) /* only for leaf level */
{
compute adj-cost;
if (routing area estimation flag = TRUE)
{
estimate routing area;
```


续表

电 路	结群方法	布局方法	向前看	布 局 结 果				
				面积	连线总长	面积利用率	cpu (S)	
name: ami33 # of cells: 33 # of nets: 121 # of pads: 38 # of pins: 423	greedy	BEAR	no	195*161 =31395	4845	75.4%	325	
			yes	185*169 =31265	5375	75.5%	2161	
		本法	—	172*169 =29068	5030	81.2%	2574	
	random	BEAR	no	168*179 =30072	8425	78%	410	
			yes	173*179 =30967	8281	76.2%	1464	
		本法	—	205*138 =28290	8124	83.4%	465	
	matching	BEAR	no	177*167 =29559	5166	79%	496	
			yes	169*168 =28392	4936	83.1%	3201	
		本法	—	170*167 =28390	4874	83.1%	2379	
	name: amd2c # of cells: 47 # of nets: 398 # of pads: 48 # of pins: 1173	greedy	BEAR	no	716*812 =581392	142095	68%	420
				yes	812*716 =581392	137100	68%	1400
			本法	—	685*716 =490480	125756	80.9%	967
random		BEAR	no	700*838 =586600	185189	67.7%	672	
			yes	668*774 =517032	152101	76.8%	4462	
		本法	—	658*704 =463232	145222	85.7%	1426	
matching		BEAR	no	717*686 =491862	126546	80.7%	683	
			yes	728*662 =481936	124297	82.4%	1120	
		本法	—	709*657 =465813	113745	85.2%	1436	

叶子级,其搜索范围为 120% (即搜索系数为 0.2)。可以发现,在面积利用率上,本布局系统比无 Lookahead 的 BEAR 布局结果要高 33—18.0% 平均为 7.0%,比有 Lookahead 的 BEAR 系统布局结果要高 0—12.9%,平均为 5.5%。在连线总长上,本系统结果要比无 Lookahead 的 BEAR 结果要少 -3.8% 到 21.6% (9 个实验中只有一个负值),平

均为 6.9%。比有 Lookahead 的要少 1.2% 到 8.9%，平均为 4.9%。在运行速度上，本系统要比有 Lookahead 的 BEAR 系统快的多，9 种实验中最大的快 7 倍。因此，本布局系统无论在质量上及速度上均是令人满意的。

五、结 论

本文提出了一种基于多个目标形状的 Building Block 布局算法，并有效的地将 top-down 及 bottom-up 两种分级设计方法结合起来，布局结果是令人满意的，并优于 BEAR 系统的布局结果。本布局系统是国家 IC CAD 三级系统的一个子系统。

感谢三级系统机房工作人员的大力协助。

参 考 文 献

- [1] Chi-Song Horng, *et al.*, 18th D. A. Conf., 293—297, 1981.
- [2] M. Brever, *J. Des. Auto. Fault Tolerant Comput.*, 1, 343 (1977).
- [3] B. T. Preas and C. W. Gwyn, 15th D. A. Conf., 206—212, 1978.
- [4] U. Lauther, 16th D. A. Conf., 1—10, 1979.
- [5] B. T. Preas *et al.*, 16th D. A. Conf., 474—481, 1979.
- [6] 俞明永, 陆生勋, 庄文君, 电子学报, 18, 13(1990).
- [7] Wei-ming Dai *et al.*, Proceedings of ICCAD'87, Santa Clara. 34—37, 1987.
- [8] B. Eschermann, Master Thesis, Univ. of California, Berkeley, 1988.
- [9] Wei-ming Dai *et al.*, *IEEE Trans., CAD-6*, 828(1987).
- [10] Ming-yong Yu and Wen-jun Zhuang, Proc. of International Conf. on CAD & CG, 460—465, 1989, Beijing.

A BBL Placement Algorithm Incorporating Two Hierarchical Design Methods

Yu Mingyong*, Bo Jianguo*, Hong Xianlong**, Y. Edmund Lien*** and
Zhuang Wenjun*

(* Beijing Software Lab., Academia Sinica)

(** Tsinghua University)

(*** Peking University)

Abstract

This paper presents a new approach for building block placement. The main idea is the introduction of multi-target shapes in the placement procedure, which incorporates the bottom-up and top-down design manners effectively and make the placement results be satisfactory. This placement algorithm has been implemented in C language. This placement system is running steadily on SUN 3/60, VAX-785, HP 9000-800 and GPX computers under the UNIX operating system. The experimental results show our placement approach outperforms the BEAR placement algorithm. Our placement system is a subsystem of national VLSICAD system (named PANDA system) being developed.

Key words Placement, Building block, Hierarchical design