

基于序列对表示的对齐约束模块布局算法*

刘 锐^{1,2+}, 洪先龙², 董社勤², 顾 钧³

¹(中国科学院 软件研究所,北京 100080)

²(清华大学 计算机科学与技术系,北京 100084)

³(香港科技大学 计算机科学系,香港)

A Block Placement Algorithm with Predefined Coordinate Alignment Constraint Based on Sequence Pair Representation

LIU Rui^{1,2+}, HONG Xian-Long², DONG She-Qin², GU Jun³

¹(Institute of Software, The Chinese Academy of Sciences, Beijing 100080, China)

²(Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China)

³(Department of Computer Science, Science and Technology University of Hong Kong, Hong Kong, China)

+ Corresponding author: Phn: 86-10-62636694, E-mail: liurui@iscas.ac.cn

<http://www.iscas.ac.cn>

Received 2002-04-01; Accepted 2002-08-14

Liu R, Hong XL, Dong SQ, Gu J. A block placement algorithm with predefined coordinate alignment constraint based on sequence pair representation. *Journal of Software*, 2003,14(8):1418~1424.

<http://www.jos.org.cn/1000-9825/14/1418.htm>

Abstract: In the system on a chip, some cells should be placed along a predefined coordinate within a relatively small region. This constraint comes from the requirement that the data should be ready on the common bus at the same time. In this paper, the module placement problem is studied where some modules have the predefined coordinate alignment (PCA) constraint. The relations are given between constrained modules, from which a necessary condition is induced. A polynomial-time algorithm is developed that can guarantee a feasible placement is always obtainable. The algorithm is implemented and tested on ami33 and ami49. The experimental results show that it is effective.

Key words: module placement; predefined coordinate alignment constraint; sequence pair

* Supported by the National Education Promotion Project (Tsinghua) under Grant No.Jc2001025 (国家教育振兴计划(清华)); the National Natural Science Foundation of China under Grant No.60076016 (国家自然科学基金); the National Grand Fundamental Research 973 Program of China under Grant No.G1998030403 (国家重点基础研究发展规划(973))

LIU Rui was born in 1974. He is a Ph.D. candidate at the Institute of Software, CAS. His research interests are algorithm and analog VLSI CAD. HONG Xian-Long was born in 1940. He is a professor and doctoral supervisor at the Department of Computer Science and Technology, Tsinghua University. His research interests include algorithm and CAD system. DONG She-Qin was born in 1964. He is an associate professor at the Department of Computer Science and Technology, Tsinghua University. His research interests include optimization algorithm and VLSI physical design. GU Jun is with the Department of Electrical and Computer Engineering, University of Calgary, Alberta T2N 1N4, Canada, and is visiting the Department of Computer Science, Hong Kong University of Science and Technology, Hong Kong. His research interests include SAT problem, optimization algorithm and VLSI CAD.

摘要: 在片上系统(SOC)的设计中,常常需要把某些单元摆放在某一预定坐标线附近相对较小的区域内.这一约束的来源是,在总线结构中,通过总线传递的数据,应当同步地出现在总线上,不能有较大的时延.对带有预定坐标线对齐(PCA)约束的模块布局问题进行研究,导出了约束模块之间的一系列关系,由此得到满足约束布局的必要条件.并且设计了多项式时间复杂度的算法,能够确保得到可行的模块布局.使用 ami33 和 ami49 对该算法进行了测试.实验结果证明,算法是有效的.

关键词: 模块布局;预定坐标线对齐约束;序列对

中图法分类号: TP391 **文献标识码:** A

In computer architecture, common bus is a usual configuration, which requires some function units, such as RAM, ROM and CPU, to be placed along the bus and the data ready on bus as well as the data sink ready to receive before next clock pulse. In the system on a chip, the same situation exists that requires some cells to be placed along a predefined Coordinate within a relative small region. Cells beyond this region will lead to large time delay. As a result, it cannot guarantee that all data sources and data sinks are ready in a limited time. We formulate this situation as predefined coordinate alignment (PCA) constraint. As far as we know, no method has been proposed for this problem.

General placements include both slicing and non-slicing. The slicing structure can be obtained by recursively cutting a rectangle into two parts by either a vertical or a horizontal line^[1]. Otherwise the structure is non-slicing. Slicing structure provides a simple way for optimizing the block orientations, defining reasonable channels in global routing and appropriately ordering the channels during detail routing. Wong and Liu^[2] proposed a normalized Polish expression to represent a slicing structure that enables the efficient neighborhood search. Typically the slicing structure is very limited since most of the dissections are non-slicing. To cover this intrinsic disadvantage, several coding schemes for non-slicing are recently proposed, namely, Sequence-Pair^[3,4], Bound-Sliceline-Grid^[5,6], Corner Block List (CBL)^[7] and O-tree^[8]. The Sequence-Pair representation uses two sets of permutations to present the topological relations of modules. We adopt the sequence pair for its concision.

In this paper, we study the PCA constraint and give the relations between constrained modules, from which a necessary condition is induced. We also derive the relation between a feasible sequence pair and a feasible placement. Furthermore, we develop a polynomial-time procedure that transforms a sequence-pair into a feasible one. The algorithm is tested on MCNC benchmarks, which gets encouraging experimental results.

The paper is organized as follows. Section 1 gives a formal definition of the problem and reviews the concept of the sequence pair. Section 2 gives the relations between constrained modules and a necessary condition is induced. In Section 3 we describe the adaptation algorithm that solves the PCA constraint problem. In Section 4, the experimental results and the conclusion are given.

1 Preliminary

1.1 Problem formulation

A set $M = \{M_1, M_2, \dots, M_n\}$ of rectangular modules lie parallel to the coordinate axes. Each rectangular module M_i is defined by a tuple (h_i, w_i) , where h_i and w_i are the height and the width of the module M_i , respectively. The aspect ratio of M_i is defined as h_i/w_i . A packing $P = \{(x_i, y_i) | 1 \leq i \leq n\}$ is an assignment of coordinates to the lower left corners of the rectangle such that there are no two rectangular modules overlapping. There are two kinds of modules in the set of modules: soft modules and hard modules. A soft module is a module whose width and height can be changed as long as the aspect ratio is within a given range and the area is as given. A hard module is a module whose width and height are fixed.

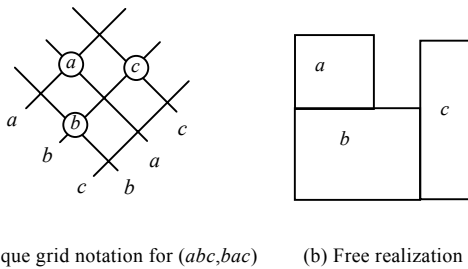


Fig.1

The module placement problem with PCA constraint is described as follows. An instance of the problem consists of two disjoint rectangular and hard module sets F and A . Any module in F is a free module that can be placed anywhere in the final placement. Any module in A is a module with the PCA constraint, whose left boundary or right boundary must fall in the region $[x_p, x_p + \Delta x]$ in the final placement. The x_p is predefined coordinate and Δx ($\Delta x < \text{width or height of each module}$) is a relative small number representing the width of net. A placement is called *feasible* if no module

of $F \cup A$ is overlap and the left boundary or right boundary of each module in A falls in the region $[x_p, x_p + \Delta x]$. Otherwise, it is called *infeasible*. For PCA constraint in y coordinate, it is similar.

1.2 Sequence pair

For the free packing problem (RP), a useful coding scheme is proposed in Refs.[3,4] as follows.

A *sequence pair* for a set of modules is a pair of sequences of the module names. For example, (abc, bac) is a sequence pair for module set $\{a, b, c\}$. It is easily understood that the variety of the sequence pair for n modules is $(n!)^2$.

A sequence pair imposes a horizontal/vertical (H/V) constraint for every pair of modules as follows:

$(...a...b..., ...a...b...) \Rightarrow a$ should be placed to the left of b

$(...b...a..., ...a...b...) \Rightarrow a$ should be placed below b

For example, sequence pair (abc, bac) imposes a set of H/V constraints: $\{a$ should be placed to the left of $c; b$ should be placed to the left of $c; b$ should be placed below $a\}$.

The H/V constraints of a sequence pair can be intuitively grasped using the *oblique-grid* notation. For example, Fig.1 (a) shows the oblique grid of sequence pair (abc, bac) . It is an $n \times n$ grid obliquely drawn on the plane which is constructed so that the first sequence is observed when one reads the module names on the positive slope lines from left to right and the second sequence is observed similarly with respect to the negative slope lines. It shows the H/V constraints: module c is in the right quarter view range (between -45° and $+45^\circ$) of module a on the oblique grid, then c should be placed to the right of a .

It has been proven in Refs.[3,4] that the set of H/V constraints imposed by every sequence pair is satisfiable, and an area minimum packing under the constraint can be obtained in polynomial time, and further, there is a sequence pair which leads an (globally) area minimum packing. Then, the sequence pair is easily utilized as a coding scheme of a stochastic algorithm.

To construct an area minimum placement for a sequence pair, one-dimensional compaction is carried out under the H/V constraints of the sequence pair. The modules are greedily pushed to the left and to the bottom as shown in Fig.1(b). The resultant placement is called the *free realization* of the sequence pair.

The time complexity is $O(n^2)$ and it can be reduced to $O(n \log n)$ ^[9]. In our implementation, we adopt the method proposed by Refs.[12,13], which can improve the area usage ratio and as a result construct a better placement.

2 Theoretical Results

There are two steps that can be used to adjust some modules to satisfy the PCA constraint. In the step 1, for any

module a in set A if

$$x(a)+w(a)<x_p, \quad (1)$$

where $x(a)$ is X coordinate of module a and $w(a)$ is width of module a , push a right by $x_p-x(a)-w(a)$ so that

$$x(a)=x_p-w(a). \quad (2)$$

All the modules m right to a ($\Gamma_a^- < \Gamma_m^-$, $\Gamma_a^+ < \Gamma_m^+$) are pushed right accordingly.

In the step 2, for any module b in set A if

$$x(b)<x_p \text{ and } x(b)+w(b)>x_p+\Delta x, \quad (3)$$

where $x(b)$ is X coordinate of module b and $w(b)$ is width of module b , push b right by $x_p-x(b)$ so that

$$x(b)=x_p. \quad (4)$$

All the modules m right to b ($\Gamma_b^- < \Gamma_m^-$, $\Gamma_b^+ < \Gamma_m^+$) are pushed right accordingly.

Lemma 1. If there is no module m in set A such that $x(m)>x_p+\Delta x$, the free realization can be adjusted to a feasible placement by the two steps and the corresponding sequence pair is said to be feasible. Otherwise, the realization cannot be adjusted to feasible placement by the two steps, and the corresponding sequence pair is said to be infeasible.

Proof. Use step 1 for any module with PCA constraint satisfying (1), and step 2 for any module with PCA constraint satisfying (3). Then all the modules with PCA constraint and satisfying $x(m)<x_p+\Delta x$ can just satisfy the PCA constraint.

As the two steps only push module right and do not modify the topology relation of modules, the module, say m and $x(m)>x_p+\Delta x$, can not be pushed left from the *free realization*. So if it with PCA constraint, it cannot be adjusted to satisfy the constraint. \square

Definition 1. relation-1: Given module a and module b , if $\Gamma_a^- < \Gamma_b^-$ and $\Gamma_a^+ > \Gamma_b^+$, module a and module b are said to have relation-1.

Definition 2. relation-2: Given module a and module b , if $\Gamma_a^- < \Gamma_b^-$, $\Gamma_a^+ < \Gamma_b^+$ and there is no module c such that $\Gamma_a^+ < \Gamma_c^+ < \Gamma_b^+$ and $\Gamma_a^- < \Gamma_c^- < \Gamma_b^-$, module a and module b are said to have relation-2.

From the Definitions 1 and 2 as well as geometry constraints implied by the sequence pair, we could prove the following lemma.

Lemma 2. In a feasible sequence pair (Γ^+, Γ^-) , any two modules in set A have either relation-1 or relation-2.

Proof. The proof is by contradiction. In a feasible sequence pair (Γ^+, Γ^-) , assume there are two modules a and b in A , and they have neither relation-1 nor relation-2. Without losing generality, assume $\Gamma_a^- < \Gamma_b^-$, $\Gamma_a^+ < \Gamma_b^+$ and there exists module c s.t. $\Gamma_a^+ < \Gamma_c^+ < \Gamma_b^+$ and $\Gamma_a^- < \Gamma_c^- < \Gamma_b^-$.

In the *free realization*, $x(c)=x(a)+w(a)$ and $x(b)=x(c)+w(c)$. Then we get

$$x(b)=x(a)+w(a)+w(c). \quad (5)$$

From the assumption that the sequence pair is feasible, we get $x(a)+w(a)<x_p$ or $x_p<x(a)<x_p+\Delta x$. After a is adjusted to satisfy the PCA constraint,

$$x(a)=x_p-w(a) \text{ or } x(a)=x_p. \quad (6)$$

From (5) and (6), we get

$$x(b)=x_p+w(c)>x_p+\Delta x \quad (7)$$

$$\text{or } x(b)=x_p+w(a)+w(c)>x_p+\Delta x. \quad (8)$$

From Lemma 1, we know in a feasible sequence pair there is not any module s.t. $x(m)>x_p+\Delta x$. This is contradicting to (7) and (8). Therefore, the Lemma 2 is proved. \square

Lemma 3. Any placement with PCA constraints has a corresponding sequence pair. Furthermore, there is a sequence pair leading to an optimal solution of the problem.

Proof. Let us consider the second claim since the first claim can be induced directly from that. There exists an

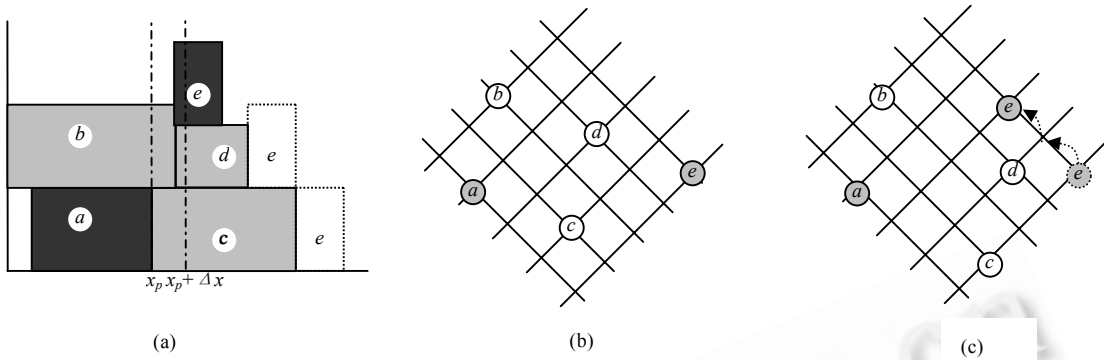


Fig.2 The process of adaptation procedure. (a) Shows the packing corresponding to the sequence pair in (c). (b) Shows the input sequence pair and the dashed show the transfer of module e .

optimal solution P for the placement with PCA constraints. From Refs.[3,4], there is a sequence pair to the placement that can be adjusted to P using the two steps. And it is also the sequence pair leading to the optimal solution P .

A necessary condition for the feasible sequence pair is given in Lemma 2. Though not all sequence pair is feasible, Lemma 3 can guarantee the optimality of the solution.

3 Adaptation

We adopt the simulated annealing algorithm to explore the solution space. The infeasible sequence pair can be just ignored. However, the optimal solution may be reached through some infeasible sequence pair, so the ignorance of the infeasible one not only (I) do harm to the smoothness of the solution space, but also (II) impairs the convergence of the result. In account of (I) and (II), we present a procedure called *adaptation*, which can transfer a given sequence pair to a feasible one by changing only the position of the modules with PCA constraint.

Procedure adaptation

Input: module set A and F , a sequence pair.

Output: a feasible sequence pair.

begin

for $j:=1$ to n do // n is the number of modules in $F \cup A$

begin

get the current position ($curX$, $curY$) of the j th module a in the second sequence by the packing method.

if module a is free mode **then**

Put the module a in the position ($curX$, $curY$).

else if a is a module with PCA constraint **then**

while ($curX > x(a) + \Delta x$)

Assume the ($j-1$)th module in the second sequence is b ;

Exchange the position of a and b in the first sequence;

Get the new position ($curX$, $curY$) of a by the packing method;

end while;

if $curX + w(a) < x_p$

then

$x(a) = x_p - w(a)$;

```

else
    x(a)=max(curX, xp); y(a)=curY;
end if;
end for;
end.
    
```

Figure 2 illustrates the excuse process of the adaptation. If there are some modules, say module a in Fig.2, whose right boundary is left to the x_p , push the module right minimally so that the right boundary reach the x_p . If there are some modules, say module e in Fig.2, whose left boundary is right to the $x_p + \Delta x$, swap this module to the top of the first sequence so that $x(e) < x_p + \Delta x$. Then if the $x(e) < x_p$, push it right so that its left boundary falls into the region $[x_p, x_p + \Delta x]$.

Theorem 1. The adaptation procedure transforms the given sequence pair to a feasible one in $O(n^2)$ time, without changing the topological relation of the free modules.

Proof. To prove the output sequence pair of the adaptation procedure is feasible, we only need to prove that after exiting the *while* loop (9), each module with PCA constraint is in the proper position, which make the whole sequence pair to be feasible. For the j th module in the second sequence a , let $S^j = \{x | \Gamma_x^+ < \Gamma_a^+\} \cap \{x | \Gamma_x^- < \Gamma_a^-\}$, $x(a) = \sum_{e \in S^j} w(e)$. If $x(a) > x_p + \Delta x$, the operation executed in the *while* loop (9) exchanges the position of a and b in the

first sequence, where b is the $(j-1)$ th module in the second sequence. Therefore, $S^{j-1} = S^j - b$ and $x(a) = \sum_{e \in S^{j-1}} w(e)$. The

post condition of *while* loop (9) guaranteed each module a with PCA constraint satisfying $x(a) \leq x_p + \Delta x$. According to Lemma 1, the resulting sequence pair is feasible. Furthermore, (10) uses step 1 and step 2 to get the packing resulting. □

4 Experiment Results and Conclusion

Table 1 Experimental results of ami33 and ami49

	With PCA constraint		Without constraint	
	Run time (s)	Area usage ration (%)	Run time (s)	Area usage ration (%)
Ami33	270	94.41	243	95.35
Ami49	400	94.31	334	94.34

The *adaptation* is embedded in a standard implementation of simulated annealing. The adaptation transforms the (Γ^+, Γ^-) to $(\Gamma^{+'}, \Gamma^{-'})$, then $(\Gamma^{+'}, \Gamma^{-'})$ is evaluated. We adopted the technique^[10,11] that if $(\Gamma^{+'}, \Gamma^{-'})$ is accept, the annealing process still use (Γ^+, Γ^-) to generate the next sequence pair. The algorithm is tested on ami33 and ami49. The results are shown in Fig.3 and the data with and without PCA constraint are given in Table 1 for comparing. From Table 1, it is clear to see the run time of our algorithm is very acceptable.

References:

[1] Otten R. Automatic floorplan design. In: Proceedings of the 19th ACM/IEEE Design Automation Conference. 1982. 261~267.
 [2] Wong DF, Liu CL. A new algorithm for floorplan design. In: Proceedings of the 23rd ACM/IEEE Design Automation Conference. 1986. 191~107.

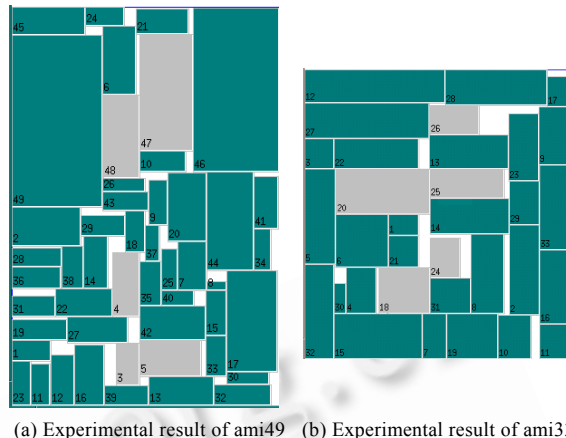


Fig.3

- [3] Murata H, Fujiyoshi K, Nakatake S, Kajitani Y. Rectangle-Packing-Based module placement. In: Proceedings of the IEEE International Conference Computer-Aided Design. 1995. 472~479.
- [4] Murata H, Fujiyoshi K, Nakatake S, Kajitani Y. VLSI module placement based on rectangle-packing by the sequence pair. IEEE Transactions on Computer-Aided Design, 1996,15(11):1518~1524.
- [5] Nakatake S, Fujiyoshi K, Murata H, Kajitani Y. Bounded-Slicing structure for module placement. Technical Report, Institute of Electronics, Information, and Communication Engineers (IEICE), 1994,VLD94(313):19~24.
- [6] Nakatake S, Fujiyoshi K, Murata H, Kajitani Y. Module placement on BSG-structure and IC layout applications. In: Proceedings of the IEEE International Conference Computer-Aided Design. 1996. 484~491.
- [7] Hong XL, Huang G, *et al.* Corner block list: An effective and efficient topological representation of non-slicing floorplan. In: Proceedings of the IEEE International Conference Computer-Aided Design 2000. 2000.
- [8] Guo PN, Cheng CK, Yoshimura T. An O-tree representation of non-slicing floorplan and its applications. In: Proceedings of the 36th ACM/IEEE Design Automation Conference. 1999. 268~273.
- [9] Tang XP, Wong DF. FAST-SP: A fast algorithm for block placement based on sequence pair. In: Proceedings of the ASP-DAC. 2001, 521~526.
- [10] Murata H, Fujiyoshi K, Daneko M. VLSI/PCB placement with obstacles based on sequence pair. IEEE Transactions on Computer-Aided Design, 1998,17(1).
- [11] Lai JB, Lin MS, Wang TC, Wang LC. Module placement with boundary constraints using the sequence-pair representation. In: Proceedings of the ASP-DAC. 2001. 515~520.
- [12] Sawa T, Nagao A, Kambe T, Shirakawa I, Chihara K. A method for rectangle packing problem. IEICE Technical Report, CAS97-38, VLD97-38, DSP97-53, 1997. 159~166 (in Japanese).
- [13] Nagao A, Kambe T, Shirakawa I. A layout approach to monolithic microwave IC. In: Proceedings of the 1998 ISPD. 1998. 65~72.

www.jos.org.cn