

# 一种可重构阵列的最小瑕点覆盖算法<sup>\*</sup>)

张祖平 陈建二

(中南大学信息科学与工程学院 长沙410083)

**摘要** 关于可重构阵列的瑕点覆盖问题受到了很多文献的关注,特别地,关于可重构阵列的最小瑕点覆盖问题等价于二分图的受约束最小点覆盖问题,并被证明是 NP-完全问题。针对本问题提出的算法运行时间为  $O(1.19^k + kn)$ , 这里  $k$  为可替换行与列的数目,改进了原有的最好结果,其运行时间为  $O(1.26^k + kn)$ ,较好地组合并扩展了研究参数计算的最新技术与经典匹配理论,且具有较好的实用价值。这是关于可重构阵列的最小瑕点覆盖问题算法又一较大的改进,也是目前最小点覆盖问题相关参数算法的较有意义的改进。

**关键词** 可重构阵列,瑕点覆盖,点覆盖,图匹配,参数计算

## Improved Algorithms about Minimum Fault Coverage in Reconfigurable Arrays

ZHANG Zu-Ping CHEN Jian-Er

(College of Information Science & Engineering, Central South University, Changsha 410083)

**Abstract** The fault coverage problem for reconfigurable arrays has received considerable attention in the literature. In particular, the minimum fault coverage problem for reconfigurable arrays is equivalent to a constrained minimum vertex cover problem on bipartite graphs and the problem has been proved to be NP-complete. This paper gives an algorithm developed for the problem, in which classical results in matching theory and recently developed techniques in the study of parameterized computation are nicely combined and extended. The algorithm is practically efficient with its running time bounded by  $O(1.19^k + kn)$ , where  $k$  is the minimum number of replacement rows and columns. The algorithm is a significant improvement over the previous algorithms for the minimum fault coverage problem for reconfigurable arrays with its running time bounded by  $O(1.26^k + kn)$ , as well as over the related parameterized algorithms for the minimum vertex cover problem.

**Keywords** Reconfigurable arrays, Fault coverage, Vertex cover, Graph matching, Parameterized computation

## 1 引言

随着 VLSI(超大规模集成电路)技术的发展,电路芯片的规模越来越大,在制作过程中可能引起的错误也随之增加,为解决芯片出错元件的问题,人们提出了一系列可重构的方法,其基本思想是在芯片中预设一些备用元件用以“修补”元件阵列中的出错元件。这一领域中研究得特别多的是可重构阵列问题(reconfigurable arrays),即如何用备用行和列来修补一元件阵列中的出错元件问题<sup>[1]</sup>。一个典型的可重构阵列包括一个元件矩阵加上一些备用行集与备用列集,如图1表示的可重构阵列有7行与9列同样的元素再加上2个备用行集与3个备用列集。当元件矩阵中出现出错元件时,我们用备用行或备用列来替代含有出错元件的行或列,从而对元件矩阵进行修补。一般地,在可重构阵列中,备用行与备用列的数目比元件矩阵的行与列的数目要小得多,因此用最小的备用行列数来修补元件矩阵中的出错元件是人们所最期望的。另外由于备用的行与列的数目都是有限的,我们也可能还需要预先指定可替代行与列的范围。这个问题可以正规地描述如下。

可重构阵列中的最小瑕点覆盖(Min-FCRA):给定一有出错元素的  $n$  行  $m$  列矩阵  $A$  和备用行数  $k_1$  及备用列数  $k_2$ ,在

矩阵  $A$  中找到一个包括  $k_1$  行和  $k_2$  列的集合  $R$  使得:(1) $R$  中的行数和列数的总和最小,(2)所有  $A$  中出错元素都包含在  $R$  的行中或列中,(3) $k_1 \leq k_1, k_2 \leq k_2$ , (或报告没有这样的集合  $R$  存在)。

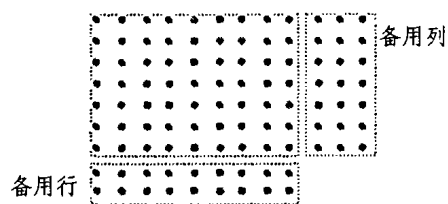


图1 一个有2个备用行与3个备用列的可重构阵列

当 Min-FCRA 中的集合  $R$  存在时,我们即可用  $k_1$  个备用行和  $k_2$  个备用列将  $R$  中的行列换掉从而除去  $A$  中所有出错元素,如果这样的集合  $R$  不存在,则出错矩阵  $A$  是不可修补的。显而易见,问题 Min-FCRA 的研究在超大规模集成电路工业中是非常重要的。

过去十几年中,人们对 Min-FCRA 问题进行了大量的研究,文<sup>[1]</sup>提出了 critical set 的概念并利用人工智能中的  $A^*$  算法<sup>[2]</sup>给出了一个算法,其时间复杂度为  $O(2^{k_1+k_2} + m\sqrt{n})$

<sup>\*</sup>)国家杰出青年自然科学基金(69928201)、国家自然科学基金(90104028)、长江学者奖励计划资助,张祖平 副教授,博士生,主要研究领域为算法理论、计算机网络与优化及大型数据库系统,陈建二 博士,教授,博士生导师,长江学者特聘教授,主要研究领域为计算机网络、计算机优化和计算机图形学等。

其中的  $k_1, k_2$  分别为备用行数及备用列数,  $n, m$  为矩阵的行数与列数), 此算法大大改进了以前对 Min-FCRA 所提出的算法, 通过实验表明, 即使对具有 1024 行和 1024 列的矩阵, 20 个备用行和 20 个备用列的大规模可重构阵列系统, 其所提算法仍在可接受的时间内解决 Min-FCRA 问题。文[3]进一步研究了 Min-FCRA 问题, 证明了 Min-FCRA 问题是 NP-难的, 从而解决了 Min-FCRA 问题的复杂性, 且提出了一个新的时间复杂度为  $O(1.26^{k_1+k_2} + (k_1+k_2)(m+n))$  (其中的  $k_1, k_2$  分别为备用行数及备用列数,  $m, n$  分别为矩阵的列数与行数) 的算法, 计算表明: 对于有 1024 行和 1024 列的矩阵、20 个备用行和 20 个备用列的可重构阵列系统, 此算法比文[1]中的算法要快 100,000,000 多倍。因此, 如果能进一步改善文[3]的算法时间复杂度中的项  $1.26^{k_1+k_2}$ , 我们可进一步提高解决 Min-FCRA 问题的速度。本文对文[3]中提出的技术做了更深入的研究, 并提出了一些新的技术, 将其时间复杂度进一步改善为  $O(1.19^{k_1+k_2} + (k_1+k_2)(m+n))$ , 这一改进对解决 Min-FCRA 问题的速度又有了数量级上的提高。对于同样的实例, 我们的新算法将比文[3]中的算法的速度提高将近 10 倍, 同时我们的技术为进一步改进算法速度提出了较好的研究方向。

## 2 二分图中的受约点覆盖问题

在研究可重构阵列中的 Min-FCRA 问题时, 人们注意到此问题与二分图中的一个受约点覆盖问题是等价的。为此, 我们先介绍一些图论的必要知识。

给定一个图  $G=(V, E)$ , 如果图  $G$  的点集  $V$  可分成二个不相交的子集  $V=U \cup L, U \cap L = \emptyset$ , 使得每一条图中的边都恰有一个端点在  $U$  中而另一个端点在  $L$  中, 则称图  $G$  为一个二分图。二分图一般以  $G=(U \cup L, E)$  来表示以指明其结点的二分法。图  $G$  中的一个边的子集  $M$  称为一个匹配, 如果  $M$  中没有任何两条边具有公共端点, 图  $G$  中的一个匹配  $M$  称为一个完美匹配, 如果图  $G$  中每一结点都是  $M$  中一条边的一个端点。图  $G$  中一个结点子集  $C$  称为  $G$  的一个点覆盖, 如图  $G$  中每条边至少有一个端点在  $C$  中。图  $G$  的一个点覆盖  $C$  称为一个最小点覆盖, 如果  $C$  中结点的个数不多于图  $G$  的任何一个点覆盖。有关二分图的图匹配和点覆盖的更详细的讨论可参阅文[3,4]。

Min-FCRA 问题的一个实例为包括有  $n$  行与  $m$  列的可重构阵列  $A$  以及 2 个整数  $k_1$  与  $k_2$ , 我们能构造一个二分图  $G_A=(U \cup L, E)$  且  $U=\{u_1, \dots, u_n\}$  与  $L=\{v_1, \dots, v_m\}$  使得在  $u_i$  与  $v_j$  中有一条边当且仅当在阵列  $A$  中的第  $i$  行与第  $j$  列是有缺陷的。容易看出, 在点集  $\{u_1, \dots, u_n\}$  中有  $k_1$  点与在  $\{v_1, \dots, v_m\}$  中有  $k_2$  个点的点集  $C$  是图  $G_A$  的点覆盖当且仅当对应的  $k_1$  行与  $k_2$  列成为阵列  $A$  的一个覆盖。特别地, 点集  $C$  是  $G_A$  的一个最小点覆盖当且仅当  $A$  中对应的行与列为阵列  $A$  的一个最小覆盖。

我们定义以下图论问题:

二分图的受约最小点覆盖问题 (Min-CVCB): 给定一个二分图  $G=(U \cup L, E)$  和两个整数  $k_1$  和  $k_2$ , 构造图  $G$  中的一个最小点覆盖  $C$  使得  $C$  中包含最多  $k_1$  个  $U$  中的结点和最多  $k_2$  个  $L$  中的结点 (或报告没有这样的最小点覆盖存在)。

文[3]证明了以下两个定理:

**定理 1**<sup>[3]</sup> 如果 Min-CVCB 问题可在时间  $O(c^{k_1+k_2} + m\sqrt{n})$  内解决 (这里的  $n$  和  $m$  分别是图中结点数和边的条

数,  $k_1$  和  $k_2$  是问题 Min-CVCB 中给定的参数,  $c$  为一正常数), 则 Min-FCRA 问题也可在时间  $O(c^{k_1+k_2} + m\sqrt{n})$  内解决 (这里的  $n$  和  $m$  分别是矩阵的行数和列数,  $k_1$  和  $k_2$  是问题 Min-FCRA 中给定的参数)。

**定理 2**<sup>[3]</sup> 解决一个 Min-CVCB 问题的实例  $(G; k_u, k_l)$  的时间复杂度为  $O(m\sqrt{n} + t(k_u+k_l))$ , 这里  $G$  是一个有  $n$  点  $m$  边的二分图,  $t(k_u+k_l)$  是解决 Min-CVCB 的一个实例  $(G'; k'_u, k'_l)$  的时间复杂度, 其中  $k'_u \leq k_u, k'_l \leq k_l$ , 且  $G'$  有完美匹配。

根据定理 1, 为进一步改进解决 Min-FCRA 问题的算法, 我们只需对 Min-CVCB 问题提出更好的算法。又根据定理 2, 要改进解决 Min-CVCB 问题的算法, 只需改进解决具有完美匹配的二分图的最小点覆盖算法。

假设  $G=(U \cup L, E)$  是一个有完美匹配的二分图, 则  $U$  中结点的个数和  $L$  中结点的个数必须相等。不难证明, 结点集  $U$  和  $L$  都分别是图  $G$  的最小点覆盖<sup>[3]</sup>。如果有完美匹配的二分图  $G$  只有两个最小点覆盖, 即  $U$  和  $L$ , 则称  $G$  为一基本二分图。

文[4]给出了以下定理, 称为 Dulmage-Mendelsohn 定理。

**定理 3**<sup>[4]</sup> (Dulmage-Mendelsohn 定理) 任何一个有完美匹配的二分图  $G=(U \cup L, E)$  都可分解为一组基本二分图序列  $B_i=(U_i \cup L_i, E_i), i=1, 2, 3, \dots, r$ , 且满足以下条件: 如果  $G$  中一条边  $e$  的两个端点在两个不同基本二分图  $B_i$  和  $B_j$  中,  $i < j$ , 则  $e$  在  $B_i$  中的端点一定在  $U_i$  中而  $e$  在  $B_j$  中的端点一定在  $L_j$  中。

由 Dulmage-Mendelsohn 定理分解出来的基本二分图  $B_i$  称为块, 块  $B_i$  叫  $d$ -块如果  $|U_i|=|L_i|=d, d$  称为块  $B_i$  的度。联接二个块间的边为块间边, 其中块间边连着的  $U$  部分的块称为有出度的块, 块间边连着的  $L$  部分的块称为有入度的块, 如对于块  $B_i$  和  $B_j, i < j$ , 则  $e$  在  $B_i$  中的端点一定在  $U_i$  中而  $e$  在  $B_j$  中的端点一定在  $L_j$  中, 则  $e$  称为块间边,  $B_i$  称为有出度的块,  $B_j$  称为有入度的块。

文[3]研究了实现 Dulmage-Mendelsohn 定理的算法的复杂性, 证明了以下定理:

**定理 4**<sup>[3]</sup> 将一个有完美匹配的二分图  $G$  按定理 3 进行分解可以在时间  $O(m^2)$  内完成, 这里  $m$  是图  $G$  的边数。

## 3 改进的 Min-CVCB 算法

根据定理 1, 我们只需讨论解决 Min-CVCB 问题的算法, 又根据定理 2, 经过时间  $O(m\sqrt{n})$  处理后, 图  $G$  分解成为一个有完美匹配的二分图  $G'$ , 其中  $m, n$  分别为图  $G$  的边数与点数。再根据定理 3 与定理 4, 经过时间  $O(m^2)$  后, 图  $G'$  已分解成为基本二分图系列, 这里的  $m$  为图  $G'$  的边数。根据以上分析, 下面只需要找出已是基本二分图系列的图的最小点覆盖且满足点数小于一给定的常数即可解决整个问题。

令  $G=(U \cup L, E)$  为一有完美匹配的二分图, 由前讨论, 图  $G$  的最小点覆盖刚好有  $k=|U|=|L|$  个结点 (这里  $|U|$  和  $|L|$  分别表示  $U$  和  $L$  中的结点的个数), 按照定理 3, 将有完美匹配的二分图  $G=(U \cup L, E)$  分解为一组基本二分图序列  $B_i=(U_i \cup L_i, E_i), i=1, 2, \dots, r$  满足以下条件: 如果  $G$  中一条边  $e$  的两个端点在两个不同的基本二分图  $B_i=(U_i \cup L_i, E_i)$  和  $B_j=(U_j \cup L_j, E_j)$  中,  $i < j$ , 则  $e$  在  $B_i$  中的端点一定在  $U_i$  中而  $e$  在  $B_j$  中的端点一定在  $L_j$  中。

可以得出,如果  $C$  是图  $G$  的最小点覆盖,则对图  $G$  的每一基本二分图  $B_i=(U_i \cup L_i, E_i)$ , 我们都有  $|C \cap (U_i \cup L_i)| = |U_i| = |L_i|$ , 因为  $C \cap (U_i \cup L_i)$  是基本二分图  $B_i$  的一个点覆盖,且  $B_i$  只有两个最小点覆盖  $U_i$  与  $L_i$ , 我们必有:

$$C \cap (U_i \cup L_i) = U_i \text{ 或 } C \cap (U_i \cup L_i) = L_i$$

以上分析说明,给定图  $G$  的任一最小点覆盖  $C$ , 对  $G$  中每一个基本二分图  $B_i=(U_i \cup L_i, E_i)$ , 或者  $C$  包含所有  $U_i$  中的结点而不包含任何  $L_i$  中的结点, 或者  $C$  包含所有  $L_i$  中的结点而不包含任何  $U_i$  中的结点。

现在来描述解决 Min-CVCB 问题的算法主体部分。令  $(G; k_u, k_l)$  为 Min-CVCB 问题的一个实例, 这里  $G=(U \cup L, E)$  为一个有完美匹配的二分图。可以假定  $G$  最多有  $2(k_u + k_l)$  个点, 这是因为如果  $G$  有大于  $2(k_u + k_l)$  个点, 则由于  $G$  是有完美匹配的二分图, 根据以上分析,  $G$  的最小点覆盖点数一定大于  $k_u + k_l$  个点, 因此已经给出 Min-CVCB 问题的否定结论。

首先应用 Dulmage-Mendelsohn 分解定理<sup>3</sup>将图  $G$  分解为基本块  $B_1, \dots, B_r$ , 使得每个块边间联接一个低标注块的  $U$ -点到一高标注块的  $L$ -点。不难看出图  $G$  的最小点覆盖一定是基本块  $B_1, \dots, B_r$  的最小点覆盖的并。因为每个基本块  $B_i=(U_i \cup L_i, E_i)$  的最小点覆盖只有2个, 即  $U_i$  与  $L_i$ , 图  $G$  的最多  $k_u$  个  $U$ -点与最多  $k_l$  个  $L$ -点的约束点覆盖  $K$  一定是从基本块中进行适当选择的  $U$ -部分与  $L$ -部分的并。

构造一个有向无环图(DAG) $D$ , 其点为  $\{B_1, \dots, B_r\}$  使得有一个边从  $B_i$  到  $B_j$  当且仅当有一条块间边从  $B_i$  中的  $U$ -部分到  $B_j$  中的  $L$ -部分。为  $D$  中的每个点  $B_i$  分配一个权重使得点  $B_i$  有一权重  $d$  如果  $B_i$  在  $G$  中是  $d$ -块。点集  $\{B_1, \dots, B_r\}$  称为簇如果  $B_1, \dots, B_r$  任意两点间都可在其中找到一条连接的路径。

算法的执行是由一个搜索树来描述的, 它的叶节点对应图  $G$  中可能的最多  $k_u$  个  $U$ -点与最多  $k_l$  个  $L$ -点的约束最小点覆盖, 每个搜索树的内部点对应搜索过程的分枝。令  $F(k_u + k_l)$  为查找二分图  $G$  的最多  $k_u$  个  $U$ -点与最多  $k_l$  个  $L$ -点的最小点覆盖的搜索树的叶结点数, 算法基于上面构造的 DAG  $D$ 。列出在搜索过程中可能的分枝情况。

**情形1**  $D$  中有权重  $\geq 4$  的点组成的簇。

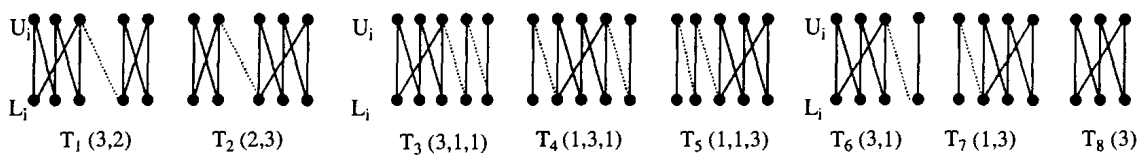


图2  $G$  中有度大于等于4的块示意图

图  $D$  中权重  $\geq 4$  的点相当于图  $G$  中基本块的度大于等于4, 在一个簇中相当于图  $G$  中度大于或等于4的块可与其它块相连, 如图2所示。

令  $B_i=(U_i \cup L_i, E_i)$ , 对图2中的  $F_1, d=|U_i|=|L_i| \geq 4$ 。因为图  $G$  中的受约束最小点覆盖  $K$  或者包含整个的  $U_i$ -部分而与  $L_i$ -部分无关或者包含整个的  $L_i$ -部分而与  $U_i$ -部分无关。分析这种情形, 或者包括块中的整个  $U_i$ -部分或者包括块中的整个  $L_i$ -部分, 对于每一种情况, 都在受约束最小点覆盖  $K$  中至少添加了4个点而点  $B_i$  将从 DAG  $D$  移走。这样图2中的  $F_1$  分枝满足如下递推关系(1):

$$F(k_u + k_l) \leq 2F(k_u + k_l - 4) \quad (1)$$

对于图2中的  $F_2$  分枝, 或者包括图中的整个  $U_i$ -部分 ( $U_i$  与  $L_i$  或不一定是包括哪一个), 这样在受约束最小点覆盖  $K$  中至少添加了4个点; 或者包括图中的整个  $L_i$ -部分而  $U_i$ -部分不包括, 这样  $L_i$ -部分一定包括(因为有块间边), 这样在受约束最小点覆盖  $K$  中至少添加了5个点。图2中的  $F_3$  分枝与  $F_2$  分枝类似, 都满足如下递推关系(2):

$$F(k_u + k_l) \leq F(k_u + k_l - 4) + F(k_u + k_l - 5) \quad (2)$$

情形1中如果权重大于或等于4的点还与其它多点相连, 则至少在  $K$  中可以添加4个点的  $U_i$ -部分或5个点的  $L_i$ -部分或者添加5个点的  $U_i$ -部分或4个点的  $L_i$ -部分, 这样满足递推关系(2)。

**情形2** 图  $D$  中有权重大于等于3而小于4的点组成的簇。

从图  $D$  中去掉情形1, 则只剩下权重小于4的点。现设定图  $D$  中的簇有一个权重大于等于3而小于4的点, 即相当于图  $G$  中基本块的度小于4而大于或等于3, 在一个簇中相当于图  $G$  中度大于或等于3而小于4的块与其它块相连, 如图3所示。

图3  $G$  中有度大于等于3而小于4的块示意图

图3中的  $T_1$ , 表示图  $D$  中权重为3的点右边至少与1个权重为2的点相连, 相当于图  $G$  中度为3的块  $k_1$  右边至少与1个度为2的块  $k_2$  相连(或者包括左右再与其它块相联的情形), 受约束最小点覆盖  $K$  或者包括块  $k_1$  中的整个  $U_i$ -部分, 这样从图中去掉了块  $k_1$  的  $L_i$ -部分(如果  $k_2$  的右边还有块的话, 一定会去掉更多的  $L$  部分, 增加相应的  $U$  部分), 同时向最小点覆盖  $K$  中至少添加了3个点; 或者包括块  $k_1$  中的整个  $L_i$ -部分, 这样从图中去掉了  $U_i$ -部分, 但一定包括块  $k_2$  的  $L$ -部分, 图中块  $k_2$  的  $U$ -部分也去掉了(如果  $k_2$  的右边还有块的话, 一定会去掉更多的  $U$  部分, 增加相应的  $L$  部分), 这样一共去掉了5个点, 也向最小点覆盖  $K$  中至少添加了5个点, 对于每一种情

况, 都向最小点覆盖中添加了点而将整个部件从 DAG  $D$  移走。这样图3中的  $T_1$  分枝满足如下递推关系:

$$F(k_u + k_l) \leq F(k_u + k_l - 3) + F(k_u + k_l - 5) \quad (3)$$

图3中的  $T_2$  表示权重为3的点左边至少与1个权重为2的点相连, 相当于图  $G$  块中的度为3的块  $k_1$  左边至少与1个度为2的块  $k_2$  相连(或者包括左右再与其它块相联的情形), 其分析与  $T_1$  类同, 满足递推关系(3),  $T_3$  与  $T_5$  类同  $T_1$ , 满足递推关系(3), 而  $T_4$  表示权重为3的点至少与2个权重为1的点相连, 相当于图  $G$  中度为3的块  $k_1$  至少与2个度为1的块  $k_2$  相连(或者包括左右再与其它块相联的情形), 其分析与图2中的  $F_1$  类同, 满足递推关系(1)。

在情形2中,去掉  $T_1$  至  $T_5$  所对应的分枝,图  $D$  中只剩下权重为3的点至多与一个权重为1的点相连,相当于图  $G$  中度为3的块  $k_1$  至多与一个度为1的块相连,所对应的分枝为图3的  $T_6, T_7, T_8$ 。

对于图3中  $T_6$  分枝,即度为3的块  $k_1$  与1个度为1的块  $k_2$  相连,向受约束最小点覆盖  $K$  中添加点的可能性只有如下3种:(1)包含  $k_1$  中的  $U$ -部分与  $k_2$  中的  $U$ -部分,(2)包含  $k_1$  中的  $U$ -部分与  $k_2$  中的  $L$ -部分,(3)包含  $k_1$  中的  $L$ -部分与  $k_2$  中的  $L$ -部

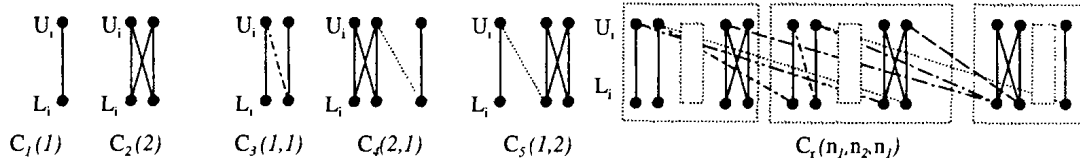


图4  $G$  中度小于等于2的块示意图

图4中的  $C_1$  与  $C_2$  为独立的权重分别为1、2的块,添加到受约束最小点覆盖  $K$  中的可能性只有2种:或者  $U$  点添加到  $K$  中的  $U$  部分或者  $L$  点添加到  $K$  中的  $L$  部分。 $C_3, C_4, C_5$  分别为权重为1的块左边或右边与其它块相连,添加到受约束最小点覆盖  $K$  中的可能性只有3种。对于块间相互联接多的情况,我们可以归约为图4中的  $C_r$  分枝,即只有出度的块排前面,只有入度的块排后面,既有出度又有入度的块排在中间。

图4中  $C_r$  部分添加到受约束最小点覆盖  $K$  中的可能性是容易得出的。如果  $K$  包含前面部分某块的  $L$  点,则  $K$  一定包含中间部分与之相连的块的  $L$  点,也一定包含后面部分与中间部分相连的相应块的  $L$  点。如果  $K$  包含前面部分某块的  $U$  点,则中间部分与之相连的块可能  $U$  点包含在  $K$  中,也可能  $L$  点包含在  $K$  中,而如果是  $L$  点包含在  $K$  中,则后面部分与之相连的相应块一定是  $L$  点包含在  $K$  中。

下面采用动态规划技术将所有有限分枝(形如分枝  $T_6, T_7, T_8$  及情形3中包括的所有分枝)构造受约束最小点覆盖  $K$ 。

首先,构造动态规划列表  $L$ ,将形如情形2中分枝  $T_6, T_7, T_8$  及情形3中的分枝  $C_1, C_2, \dots, C_r$  作为列表的行名(中间可能有相同的分枝,为了明确起见,只列其中的一个作为示例), $1, 2, \dots, k_u$  作为列表的列名, $L[i, j]$  为  $T$  的充分必要条件为有一种方法在  $1, 2, 3, \dots, i$  行中找到  $j$  个点到  $K$  中的  $U$  部分,如下表1。

表1 动态规划列表示例

行号	行名	0	1	2	3	4	5	6	7	...	$k_u$
1	$T_6$	T			T	T					
2	$T_7$	T	T		T	T	T		T		
3	$T_8$	T	T		T	T	T	T	T		
4	$C_1$	T	T	T	T	T	T	T	T		
5	$C_2$	T	T	T	T	T	T	T	T		
6	$C_3$	T	T	T	T	T	T	T	T		
	...										
$s$	$C_r$	T	T	T	T	T	T	T	T	...	T

表1中的  $T_6$  行表示从  $T_6$  中可以取0个、3个或4个点到  $K$  中的  $U$  部分, $T_7$  行表示在  $T_6$  行的基础上,从  $T_7$  中再取0个、1个或4个点到  $K$  中的  $U$  部分,依此类推。

在动态规划列表中如果最后的行全为  $T$ ,则可以通过适当的数据结构记下最后一行  $T$  的形成,从而知道  $K$  中  $U$  部

分。图3中  $T_7$  分枝与  $T_6$  分枝类似。对于图3中  $T_8$  分枝,向  $K$  中添加点的可能性只有2种:或者包含  $U$ -部分的点或者  $L$ -部分的点。

情形3 图  $D$  中权重  $\leq 2$  的点形成的簇。

从图  $D$  中去掉情形1与情形2中的簇后,图中的点权重最多为2。对应的图  $G$  中的情形为只有度小于等于2的块,如图4。

分的形成过程,由此  $K$  中的  $L$  部分也相应形成。如果列表中最后行列不为  $T$ ,则说明在  $D$  中找不到满足条件的约束点覆盖  $K$ 。

从表1可知,列表中变化的值最多为  $k_u$  列与  $(k_u + k_l)$  行(因为最多  $2(k_u + k_l)$  个点),由于下一行的值要依赖于上一行生成,因此每行需要花费最多  $O(k_u^2)$  时间,整个表的形成需要花费最多  $O(k_u^2) * (k_u + k_l)$  即最多  $O((k_u + k_l)^3)$  时间。

#### 4 算法及其分析

总结前面章节中的讨论,引出如下算法 CVCB-Main,并将算法各步解释如下。

算法 CVCB-Main

输入:一个二分图  $G=(U \cup L, E)$  且两个整数  $k_u$  与  $k_l$

输出: $G$  的一个有最多  $k_u$  个  $U$ -点与最多  $k_l$  个  $L$ -点的最小点覆盖  $K$

1. 将每个度大于  $k_l$  的  $U$ -点  $u$  包括在  $K$  中且将  $u$  从  $G$  中移去; $k_u = k_u - 1$ 。

2. 将每个度大于  $k_u$  的  $L$ -点  $v$  包括在  $K$  中且将  $v$  从  $G$  中移去; $k_l = k_l - 1$ 。

3. 应用定理2简化实例使得  $G$  是一个有完美匹配的二分图;有最多  $2(k_u + k_l)$  个点(整数  $k_u$  与  $k_l$  及最小点覆盖  $K$  也相应地改变了)。

4. 应用 Dulmage-Mendelsohn 分解方法(定理3)来分解图  $G$  为基本块  $B_1, B_2, \dots, B_r$  并进行拓扑排序。

5. 应用第3节的情形1的所有分枝与情形2中的  $T_1$  至  $T_5$ , 分枝表达式相应形成。

6. 采用动态规划技术,解决第3节中情形2的  $T_6$  至  $T_8$  与情形3中所有分枝。

第1-2步对于高度数的点形成了直接的结论,因为如果度大于  $k_l$  的  $U$ -点  $u$  不在最小点覆盖  $K$  中,则所有  $u$  的邻居都将在  $K$  中,这将超出了界  $k_l$ ,因此每个度大于  $k_l$  的  $U$ -点  $u$  应该自动地包含在  $K$  中。类似地应用判断度大于  $k_u$  的  $L$ -点。当然如果在这些步骤中  $k_u$  或  $k_l$  成为负数,则将停止并声称不存在期望的最小点覆盖。第1-2步明显花线性时间。定理2允许在第3步进一步减少二分图  $G$  使得  $G$  有完美匹配(整数  $k_u$  与  $k_l$  也一起减少了),这一步的运行时间为  $O((k_u + k_l)^{1.5})$ [3],图  $G$  中的点数现在最多为  $2(k_u + k_l)$ ,图  $G$  中的边数也最多为  $O(k_u + k_l)$ 。

第4步应用了定理3将图  $G$  分解为基本二分子图,由定理

4, 这费时为  $O((k_v+k_l)^2)$ 。

如在第3节讨论的, 如果应用情形1中的任一分枝, 分枝搜索过程并满足递推公式(1)与(2); 如果应用情形2中的  $T_1$  至  $T_5$  分枝, 分枝搜索过程并满足递推公式(3)与(1); 最后动态规范列表所花时间最多为  $O((k_v+k_l)^3)$ 。

根据文[5]中的 Master 定理, 容易验证:  $F(k_v+k_l) = 1 \cdot 19^{k_v+k_l}$  满足所有递推关系(1)、(2)与(3), 由此我们知道解决 Min-CVCB 问题的时间为  $O(1 \cdot 19^{k_v+k_l} + (k_v+k_l)^3 + m\sqrt{n})$ , 这里  $m$  为图  $G$  的边数, 即相当于矩阵列  $A$  的错误点数,  $n$  为图  $G$  的点数, 即相当于矩阵列  $A$  的行数与列数之和  $m+n$ 。通常情况下,  $k_v+k_l$  比  $n$  小得多, 因此我们有如下定理。

**定理5** Min-CVCB 问题能在时间  $O(1 \cdot 19^{k_v+k_l} + m\sqrt{n})$  内解决, 其中  $k_v+k_l$  为给定的参数,  $n, m$  分别为二分图的点与边数, 也即 Min-FCRA 问题能在时间  $O(1 \cdot 19^k + kn)$  内解决, 这里  $k$  为可替换的行与列数,  $n$  为矩阵的行与列数。

**结论与进一步的研究** 参数计算定理在工业应用中设计实际的算法是相当有用的, 特别是对于许多 NP-难的最优化问题。在本文中, 研究了在 VLSI 制造领域得到了广泛关注的可重构阵列的最小瑕点覆盖的复杂性, 并提出了简单而有意义的改善算法。本文提出的算法概念简单、容易实现, 且很好地组合了最近在研究参数计算中提出的技术与经典几何构造的结果。

本文的改进是基于减少指数运行时间中的底数, 在研究

NP-难的优化问题中具有理论与实际的重要性。如在参数化的点覆盖算法中减少底数 0.01, 则在生物化学的相关应用程序中可以达到改善 60% 的运行时间<sup>[6]</sup>。算法相对以前关于该问题的算法减少了底数 0.07 (从  $1 \cdot 26^k$  到  $1 \cdot 19^k$ )。

进一步的研究是再详细分析本文第3节中的各种情形, 将算法的运行时间指数部分改善为  $(1+\epsilon)^k$ 。

## 参考文献

- Hasan N, Liu C L. Minimum Fault Coverage in Reconfigurable Arrays. In: Proc. 18th Int. Symp. on Fault-Tolerant Computing (FTCS'88), 1988. 348~353
- Nilsson N J. Principles of Artificial Intelligence. Tioga Publishing Co., 1980
- Chen J, Kanj I. An Constrained Minimum Vertex Covers of Bipartite Graphs: Improved Algorithms. In: Proc. 27th Intl. Workshop on Graph-Theoretic Concepts in Computer Science (WG'2001). Lecture Notes in Computer Science 2204, 2001. 55~65
- Lovasz L, Plummer M D. Matching Theory, Annals of Discrete Mathematics 29, North-Holland, 1986
- Cormen T H, Leiserson C E, Rivest R L. Introduction to Algorithms. McGraw-Hill Book Company, New York, 1992
- Bodlaender H L, Downey R G, Fellows M R, et al. Parameterized complexity analysis in computational biology. Computer Applications in the Biosciences, 1995, 11: 49~57
- Carey M, et al. Towards Heterogeneous Multimedia Information Systems: The Garlic Approach. In: Proc. RIDE-DOM Workshop, 1995. 124~131
- Kapitskaia O, Tomasic A, Valduriez P. Scaling Heterogeneous Databases and the Design of Disco: [INRIA Technical Report]. 1997
- Garca-Molina H, Labio W, Yerneni R. Capability sensitive query processing on internet sources. In: Proc. of the 15th Intl. Conf. on Data Engineering, Sydney, Australia, Mar. 1999
- Morris K, Ullman J, Gelder A. Design Overview of the NAIL! System. In: Proc. ICLP, 1986
- Kirk T, Levy A, Sagiv Y, Srivastava D. The Information Manifold. AAAI Symposium on Information Gathering in Distributed Heterogeneous Environment, 1995
- Yang J, Xu L. Describing capabilities of internet data sources for information discovery and sharing. ADC 2001. 21~28
- Sycara K, Widoff S, Klusch M, Lu J. LARKS: Dynamic Matchmaking Among Heterogeneous Software Agents in Cyberspace. Kluwer, 2002
- Cali A, Calvanese D, Giacomo G D, Maurizio Lenzerini. On the Expressive Power of Data Integration Systems. In: Proc. of the 21st Int. Conf. on Conceptual Modeling (ER 2002). 338~350
- Halevy A Y. Theory of Answering Queries Using Views. SIGMOD Record, 2000, 29(4): 40~47
- Navathe S, Donahoo J. Towards Intelligent Integration of Heterogeneous Information Sources. In: Proc. of the 6th Intl. Workshop on Database Re-engineering and Interoperability, March 1995
- Jakobovits R. Integrating Heterogeneous Autonomous Information Sources: [Univ. of Washington Technical Report, UW-CSE-971205]. July 1997
- Li C, Yerneni, Vassalos V, Papakonstantinou Y, Garcia-Molina H, Ullman J. Capability based mediation in tsimmis. In: Proc. of ACM SIGMOD Conf. 1998
- Haas L, Kossman D, Wimmers E, Yang J. Optimizing queries across diverse data sources. In: Proc. of VLDB Conf. 1997

(上接第172页)

奠定了基础。相对于传统的“源描述”, 该框架中的模式信息规范清晰描述了参与集成的各异构数据源局部模式与中介模式的语义映射, 为选择查询相关数据源提供依据; 其次, 借助查询能力的精确刻画, 为确定查询子目标的次序并向参与查询的数据源传送最小的查询提供保证。

我们下一步要进行的工作是利用已有的大量传统数据, 如关系数据、结构文档等作为数据基础, 采用虚拟法建立基于数据源能力描述的异构集成原型系统。

## 参考文献

- Navathe S, Donahoo J. Towards Intelligent Integration of Heterogeneous Information Sources. In: Proc. of the 6th Intl. Workshop on Database Re-engineering and Interoperability, March 1995
- Jakobovits R. Integrating Heterogeneous Autonomous Information Sources: [Univ. of Washington Technical Report, UW-CSE-971205]. July 1997
- Li C, Yerneni, Vassalos V, Papakonstantinou Y, Garcia-Molina H, Ullman J. Capability based mediation in tsimmis. In: Proc. of ACM SIGMOD Conf. 1998
- Haas L, Kossman D, Wimmers E, Yang J. Optimizing queries across diverse data sources. In: Proc. of VLDB Conf. 1997
- Carey M, et al. Towards Heterogeneous Multimedia Information Systems: The Garlic Approach. In: Proc. RIDE-DOM Workshop, 1995. 124~131
- Kapitskaia O, Tomasic A, Valduriez P. Scaling Heterogeneous Databases and the Design of Disco: [INRIA Technical Report]. 1997
- Garca-Molina H, Labio W, Yerneni R. Capability sensitive query processing on internet sources. In: Proc. of the 15th Intl. Conf. on Data Engineering, Sydney, Australia, Mar. 1999
- Morris K, Ullman J, Gelder A. Design Overview of the NAIL! System. In: Proc. ICLP, 1986
- Kirk T, Levy A, Sagiv Y, Srivastava D. The Information Manifold. AAAI Symposium on Information Gathering in Distributed Heterogeneous Environment, 1995
- Yang J, Xu L. Describing capabilities of internet data sources for information discovery and sharing. ADC 2001. 21~28
- Sycara K, Widoff S, Klusch M, Lu J. LARKS: Dynamic Matchmaking Among Heterogeneous Software Agents in Cyberspace. Kluwer, 2002
- Cali A, Calvanese D, Giacomo G D, Maurizio Lenzerini. On the Expressive Power of Data Integration Systems. In: Proc. of the 21st Int. Conf. on Conceptual Modeling (ER 2002). 338~350
- Halevy A Y. Theory of Answering Queries Using Views. SIGMOD Record, 2000, 29(4): 40~47