

DOI:10.11918/201909174

2 维约束覆盖数组最小规模下限的提升

盛云龙¹,魏长安¹,姜守达¹,付 尧¹,赵伟志²

(1. 哈尔滨工业大学 电子与信息工程学院 自动化测试与控制研究所,哈尔滨 150080;
2. 北京工业大学,北京 100000)

摘 要: 针对现有方法评价 2 维约束覆盖时没有考虑约束,而给出的最小规模过大的问题. 为获得更准确的 2 维约束覆盖数组的最小规模,评价现有算法生成的 2 维约束覆盖数组,本文提出一种可以提升 2 维约束覆盖数组最小规模下限的禁忌边分解方法. 采用禁忌边分解方法将描述被测系统输入配置的图分解成两个子图,通过计算覆盖两个子图中全部顶点的子覆盖数组的规模和剩余需要覆盖的取值组合数,与单纯计算需要覆盖的取值组合数相比,提升了 2 维约束覆盖数组的最小规模,所提出的方法能够得到更逼近真实值的最小规模的下限,一旦 2 维约束覆盖数组的规模小于最小规模的下限,则其不可能存在. 本文的实验方法是,将禁忌边分解方法应用到现有的被测系统中,得到其 2 维约束覆盖数组最小规模的下限,将最小规模的下限与生成算法给出的 2 维约束覆盖数组的规模进行对比. 实验结果表明:禁忌边分解方法给出的最小规模下限可以用于评价现有算法生成的 2 维约束覆盖数组,有助于判断其是否真实存在.

关键词: 组合测试;约束覆盖数组;禁忌边分解;最小规模;下限

中图分类号: TP311.5 **文献标志码:** A **文章编号:** 0367-6234(2020)05-0017-06

Promotion of lower bounds of minimum sizes of 2-way constrained covering arrays

SHENG Yunlong¹, WEI Chang'an¹, JIANG Shouda¹, FU Yao¹, ZHAO Weizhi²

(1. Automatic Test and Control Institute, School of Electronics and Information Engineering, Harbin Institute of Technology, Harbin 150080, China; 2. Beijing University of Technology, Beijing 100000, China)

Abstract: When evaluating 2-way constrained covering arrays, the minimum sizes given by the existing methods are too large because the constraints are ignored. In order to obtain more accurate minimum sizes of 2-way constrained covering arrays and evaluate the 2-way constrained covering arrays generated by existing algorithms, a forbidden edge decomposing method was proposed to promote lower bounds of the minimum sizes of 2-way constrained covering arrays in this study. The graphs that describe the input configurations of the systems under test were decomposed into two subgraphs. Through calculating the sizes of the sub constrained covering arrays which cover all the vertices in the two subgraphs and the number of the remaining value combinations to be covered, the minimum sizes of the 2-way constrained covering arrays were promoted compared with the number of the value combinations to be covered. The lower bounds of the minimum sizes obtained from the proposed method were closer to the real values. If the sizes of the 2-way constrained covering arrays are less than the lower bounds of the minimum sizes, then it is impossible for them to exist. The proposed experimental method applied the forbidden edge decomposing method to the existing systems under test, then obtained the lower bounds of the minimum sizes of 2-way constrained covering arrays, and compared the lower bounds of the minimum sizes with the sizes of the 2-way constrained covering arrays given by the generation algorithms. The experimental results show that the lower bounds of the minimum sizes given by the forbidden edge decomposing method can be used to evaluate the 2-way constrained covering arrays generated by the existing algorithms, and are helpful to determine their existence.

Keywords: combinatorial testing; constrained covering array; forbidden edge decomposing; minimum size; lower bound

组合测试是一种重要的黑盒测试技术^[1-4],利用更少的测试数据量,可以发现更多的由参数之间的相互作用引起的系统故障. 通过对任意 t 个参数全部取值组合的覆盖,可以发现几乎 100% 的系统故障,因此又被称为“伪穷举测试”^[5].
在实际测试中,参数之间往往存在约束条件,限

收稿日期: 2019-09-23
作者简介: 盛云龙(1988—),男,博士研究生;
姜守达(1964—),男,教授,博士生导师
通信作者: 魏长安,weichangan@hit.edu.cn

制了参数之间的取值关系^[6-8]. 这种情况下, 组合测试的测试数据集被称为 t 维约束覆盖数组, 其可覆盖任意 t 个参数之间满足约束的取值组合. $t=2$ 的约束覆盖数组被称为 2 维约束覆盖数组, 因为其规模小, 所以最为常用. 为了提高实际测试的效率, 2 维约束覆盖数组的规模应尽可能的小, 最好找到理论上最小规模的 2 维约束覆盖数组. 然而找到规模最小的 2 维约束覆盖数组已被证明是 NP 完全 (Non-deterministic Polynomial Complete, NPC) 问题^[9-10], 因此为减少生成时间, 很多测试数据搜索算法被应用到寻找规模最小或近似最小的 2 维约束覆盖数组. 经典算法有 HSS^[11]、SA_SAT^[12]、mAETG_SAT^[13]、PICT、TestCover、IPOG^[14]、IPOG-F^[15]、CTWT^[16] 等. 前两个算法属于启发式算法, 其余算法属于贪心算法.

为了评价这些算法的性能, 需要将这些算法生成的 2 维约束覆盖数组的规模与理论上的最小值进行比较. 目前可以得出的理论最小值是任意两个参数之间满足约束的取值组合数的最大值, 然而这个值只表示了约束下需要覆盖的取值组合数, 并没有考虑约束的影响. 由于约束的影响, 2 维约束覆盖数组不仅要考虑需要覆盖的取值组合数, 还要考虑约束情况, 约束的存在使得原本可以在一条测试数据中覆盖的取值组合, 要被拆开放到多条测试数据中, 这样就增加了测试数据的数目, 导致 2 维约束覆盖数组的规模增加, 因此 2 维约束覆盖数组最小规模要考虑约束的影响.

本文提出一种禁忌边分解方法, 可以将描述被测系统输入配置的图分解成两个子图, 通过计算覆盖两个子图中全部顶点的子覆盖数组的规模和剩余需要覆盖的取值组合数, 与单纯计算需要覆盖的取值组合数相比, 提升了 2 维约束覆盖数组的最小规模. 提升的最小规模可以用于评价现有算法生成的 2 维约束覆盖数组, 比较生成的 2 维约束覆盖数组的规模与理论最小规模之间的差距, 同时最小规模有助于判断生成的 2 维约束覆盖数组是否真实存在.

1 基本概念

假设一个被测系统具有 k 个参数, 分别为 $P_1, P_2, \dots, P_k$, 每个参数分别对应 $v_1, v_2, \dots, v_k$ 个取值, 即对于任意参数 $P_i (1 \leq i \leq k)$ 其有 v_i 个取值, 具体表示为集合 $\{0, 1, \dots, v_i - 1\}$, 简记为 $[0, v_i - 1]$, 各个参数之间取值均相互独立. 记参数集合 $P = \{P_1, P_2, \dots, P_k\}$, 取值个数集合 $V = \{v_1, v_2, \dots, v_k\}$, 配置空间模型 (Configuration Space Model, CSM) $M = <$

$P, V >$, 组合测试根据被测系统的配置空间模型来构造测试数据集.

定义 1 (约束覆盖数组, 约束混合覆盖数组)^[17]: 设 A 是一个 $n \times k$ 的矩阵, 约束元组 $f = (a_1, a_2, \dots, a_k)$ 是一个 k 维向量, 用于限制取值组合的出现, $a_i \in [0, v_i - 1] \cup [x] (1 \leq i \leq k)$, 如果约束中没有指定第 j 个参数的取值, 则 $a_j = x$. 对任意一个出现在测试数据中, 且不会引起测试数据中出现约束元组的 t 维组合 $I = \{(P_{i_1}, a_{i_1}), (P_{i_2}, a_{i_2}), \dots, (P_{i_t}, a_{i_t})\}$, $a_{i_j} \in [0, v_{i_j} - 1] (1 \leq j \leq t)$, 如果 A 至少存在一行 r , 使得 $A[r, i_j] = a_{i_j}$ 且不覆盖约束元组, 则称 A 为约束覆盖数组 (Constrained Covering Array, CCA), 记为 $CCA(n; t, k, v; F)$, 其中 F 是约束元组 f 的全集.

没有覆盖约束元组的测试数据和测试数据集是满足约束一致性的, A 和 A 中的测试数据都是满足约束一致性的. 当参数的取值个数不同时, A 被称为约束混合覆盖数组 (Constrained Mixed Covering Array, CMCA), 记为 $CMCA(n; t, (v_1, v_2, \dots, v_k), F)$.

在记录 CCA 或者 CMCA 时, 可以把一些具有相同值域的项合并, 如 $CMCA(n; t, (v_1, v_2, \dots, v_k), F)$ 可以表示为 $CMCA(n; t, s_1^{p_1} s_2^{p_2} \dots s_r^{p_r}, F)$, 其中 $k = \sum_{i=1}^r p_i$.

定义 2 (带有禁忌边的覆盖数组)^[9] 设 A 是一个 $n \times k$ 的带有禁忌边的覆盖数组 (Covering Array with Forbidden Edges, CAFE), 表示为 $CAFE(n, G)$, 其中 $G = G_{(g_1, g_2, \dots, g_k)}$ 是参数取值构成的图, $g_i (1 \leq i \leq k)$ 是图中第 i 列顶点的集合, $E(G)$ 是 G 中的边集, 简记为 E 或 E_G . 对 A 中任意一列, 其取值范围为 $[0, |g_i| - 1]$. 对任意的不属于同一列的两个顶点 $a_i \in [0, |g_i| - 1]$ 和 $b_j \in [0, |g_j| - 1] (i \neq j)$, 如果 $\{a_i, b_j\} \notin E(G)$ 且满足约束一致性, 则 A 中一定存在一行 r , 使得 $A[r, i] = a, A[r, j] = b$.

E 中的边就是约束, 在图中被称为边或禁忌边. $CAFE(n, G)$ 是力度为 2 的约束覆盖数组. N 或 N_c 用来表示对应于图 G 的覆盖力度为 2 的带有禁忌边的覆盖数组的最小规模. 从图 G 中容易得出如下式所示关系式:

$$N_c \geq \max_{1 \leq i < j \leq k} (|g_i| |g_j| - |E_{ij}|). \quad (1)$$

式(1)中 E_{ij} 表示图 G 中第 i 列和第 j 列中满足约束一致性的边, N_c 的下限是图 G 中任意两列满足约束一致性的取值组合个数的最大值.

定义 3 (顶点子图) $G_{i,a}$ 是图 G 的一个子图, 移除了图 G 中第 i 列除了 a 之外的全部顶点 ($1 \leq i \leq k$), 并且移除了与第 i 列中顶点 a 构成边的顶点.

定义 4 (顶点带禁忌边的覆盖数组) A 是顶点子图 $G_{i,a}$ 的一个 $n \times k$ 的顶点带有禁忌边的覆盖数组, 顶点带有禁忌边的覆盖数组简记为 $\text{CAFE}(n, G_{i,a})$, 对顶点子图 $G_{i,a}$ 中的任意一个顶点 b_j , A 中一定存在一行 r 满足约束一致性且 $A[r, j] = b$.

定义 5 (两列的取值组合个数) $I_{i,j}(A_{G_{i,a}})$ 表示 $\text{CAFE}(n, G_{i,a})$ 中覆盖的第 i 列和第 j 列的取值组合个数. 如果取值组合重复出现, 则只记录一次.

2 基于禁忌边分解方法的约束覆盖数组最小规模下限的提升

式(1)在估计 N_G 的范围时, 只是根据两个参数之间满足约束的取值组合数进行估计, 并没有依据实际的约束的特点进行估计, 因此得出的 N_G 的范围较为宽泛, 通过对禁忌边进行分解可以更准确的估计 N_G 的范围.

通过对图中的边进行分解, 进一步分析组合约束对其余取值的限制关系, 可以更为准确的得到 $\text{CAFE}(n, G)$ 的最小规模 N_G 的下限, 式(2)所示为对 G 中的一条边 (a_i, b_j) 进行分解得到的 N_G 的下限.

$$N_G \geq \max_{1 \leq i \leq k, (a_i, b_j) \in E_{ij}} (|g_i| |g_j| - |E_{ij}| + N_{G_{i,a}} + N_{G_{j,b}} - I_{i,j}(A_{G_{i,a}}) - I_{i,j}(A_{G_{j,b}})). \quad (2)$$

如式(2)所示, (a_i, b_j) 是图 G 中的任意一条边, 基于边 (a_i, b_j) 中的两个顶点可以从图 G 中分出两个顶点子图 $G_{i,a}$ 和 $G_{j,b}$. $A_{G_{i,a}}$ 和 $A_{G_{j,b}}$ 是 $G_{i,a}$ 和 $G_{j,b}$ 对应的两个顶点带禁忌边的约束覆盖数组, $N_{G_{i,a}}$ 和 $N_{G_{j,b}}$ 是 $A_{G_{i,a}}$ 和 $A_{G_{j,b}}$ 的规模, $I_{i,j}(A_{G_{i,a}})$ 和 $I_{i,j}(A_{G_{j,b}})$ 是 $A_{G_{i,a}}$ 和 $A_{G_{j,b}}$ 中 i 和 j 两列参数的取值组合数.

式(2)的含义有如下两种解释:

1) 为了覆盖第 i 列和第 j 列参数的 $|g_i| |g_j| - |E_{ij}|$ 个取值组合, $\text{CAFE}(n, G)$ 规定一定包含顶点 a_i 与第 j 列参数的全部满足约束的取值组合和顶点 b_j 与第 i 列参数的全部满足约束的取值组合, 这两组参数取值组合分别在规模最小的 $A_{G_{i,a}}$ 和 $A_{G_{j,b}}$ 中被覆盖, 然而 $\text{CAFE}(n, G)$ 中仍有 $(|g_i| |g_j| - |E_{ij}| - I_{i,j}(A_{G_{i,a}}) - I_{i,j}(A_{G_{j,b}}))$ 个满足约束的取值组合需要覆盖, 加上已经生成的规模 $N_{G_{i,a}} + N_{G_{j,b}}$, 即得到 N_G 的下限.

2) 因为最终生成的覆盖数组中一定覆盖了包含顶点 a_i 与第 j 列参数的全部满足约束一致性的取值组合, 和顶点 b_j 与第 i 列参数的全部满足约束的取值组合, 所以规模最小的 $A_{G_{i,a}}$ 和 $A_{G_{j,b}}$ 一定存在于 $\text{CAFE}(n, G)$ 中, 原本 $\text{CAFE}(n, G)$ 的最小规模为 $|g_i| |g_j| - |E_{ij}|$, 但是为了覆盖其中的 $I_{i,j}(A_{G_{i,a}})$ 个参数取值组合, 需要增加的测试数据规模为 $N_{G_{i,a}} -$

$I_{i,j}(A_{G_{i,a}})$, 为了覆盖 $I_{i,j}(A_{G_{j,b}})$ 个参数取值组合, 需要增加的测试数据规模为 $N_{G_{j,b}} - I_{i,j}(A_{G_{j,b}})$, 所以为了覆盖 $|g_i| |g_j| - |E_{ij}|$ 个取值组合, 至少还要在其基础上增加 $N_{G_{i,a}} + N_{G_{j,b}} - I_{i,j}(A_{G_{i,a}}) - I_{i,j}(A_{G_{j,b}})$ 条测试数据.

以一个有 4 个参数的被测系统为例, 其第 1 个参数有 2 个取值(0 和 1), 其余参数有 3 个取值(0、1 和 2), 其中参数 2 和参数 3 的取值组合(0,0)、参数 2 和参数 4 的取值组合(2,2)、参数 3 和参数 4 的取值组合(1,1)是约束, 如图 1 所示.

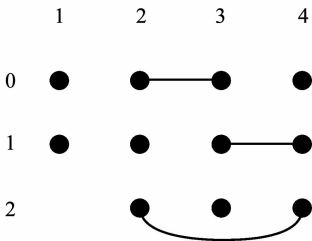


图 1 被测系统参数及取值配置对应图举例

Fig. 1 Example graph of the parameters and value configurations of the system under test

为其生成带有禁忌边的覆盖数组时, 按照式(1)的结论, 可以得出 $N_G \geq 9 - 1 = 8$. 然而在实际构造带有禁忌边的约束覆盖数组时, 由于约束的存在, 需要增加一定的测试数据才能覆盖 8 个取值组合, 其最小规模为 10, 如表 1 所示. 为了证明其最小规模为 10, 采用禁忌边分解方法首先对图进行分解, 然后计算需要覆盖的参数间的取值组合数.

表 1 带有禁忌边的覆盖数组举例

Tab. 1 Example of the covering array with forbidden edge				
序号	参数 1	参数 2	参数 3	参数 4
1	0	0	1	0
2	0	1	0	2
3	1	2	0	0
4	1	0	2	1
5	0	2	2	1
6	1	1	1	2
7	1	1	2	0
8	0	0	2	2
9	0	1	0	1
10	0	2	1	0

图 2 和图 3 分别是顶点子图 $G_{2,0}$ 和 $G_{3,0}$, 在 $G_{2,0}$ 中移除了与第 2 个参数顶点 0 同一列的其他取值, 同时移除了与顶点 0 构成边的其余参数的顶点, 参数 3 的顶点 0. 同理可得 $G_{3,0}$. 表 2 和表 3 分别是对应顶点子图 $G_{2,0}$ 和 $G_{3,0}$ 的 $\text{CAFE}(3, G_{2,0})$ 和

CAFE($3, G_{3,0}$), 记为 $A_{G_{2,0}}$ 和 $A_{G_{3,0}}$, 其规模都为 3. 因为每个顶点子图中, 参数 4 最多有 3 个取值, 所以覆盖这两个顶点子图中的全部顶点 (取值) 至少需要 3 条测试数据. 所以 $A_{G_{2,0}}$ 和 $A_{G_{3,0}}$ 都具有最小规模, 且覆盖的参数 2 和参数 3 的取值组合数 $I_{2,3}(A_{G_{2,0}})$ 和 $I_{2,3}(A_{G_{3,0}})$ 都是 2, 所以由式 (2) 可得 $N_C \geq 3 \times 3 - 1 + 3 + 3 - 2 - 2 = 10$, 从另外的两个边着手也可以得到相同的 N_C 的下限, 因此可以得出 N_C 的下限就是 10.

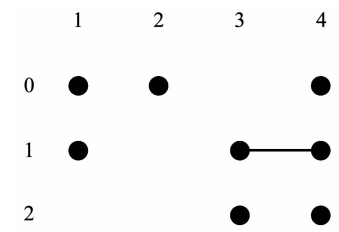


图 2 $G_{2,0}$
Fig. 2 $G_{2,0}$

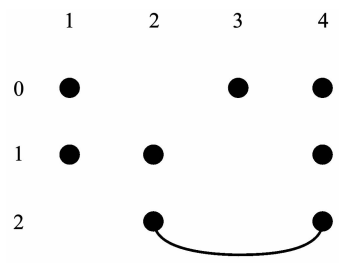


图 3 $G_{3,0}$
Fig. 3 $G_{3,0}$

表 2 CAFE($3, G_{2,0}$)
Tab. 2 CAFE($3, G_{2,0}$)

序号	参数 1	参数 2	参数 3	参数 4
1	0	0	1	0
2	1	0	2	1
3	0	0	1	2

表 3 CAFE($3, G_{3,0}$)
Tab. 3 CAFE($3, G_{3,0}$)

序号	参数 1	参数 2	参数 3	参数 4
1	0	1	0	0
2	1	2	0	1
3	0	1	0	2

根据对图中边进行分解计算 CAFE(n, G) 的最小规模下限的过程, 可以得出如下推论:

如果一个图满足下面 3 个条件:

1) 所有的列具有相同的顶点数, 即被测系统的每个参数具有相同的取值个数 $|g_1| = |g_2| = \cdots = |g_k| = g$;

2) 参数 i 和 j 之间只有一条边 $(a_i, b_j), a, b \in [0, g-1], i, j \in [1, k]$ 且 $i \neq j$;

3) 不存在任意顶点 $c_l, c \in [0, g-1]$ 且 $l \in [1, k]$, 使得 $(c_l, a_i), (c_l, b_j) \notin E_G$.

则有如下关系:

$$N_C \geq |g_i| |g_j| - 1 + |g_i| - (|g_i| - 1) + |g_j| - (|g_j| - 1) = |g_i| |g_j| + 1 = g^2 + 1$$

上述推论的证明过程与上例计算 N_C 的过程相同. 该推论可以用一个简单的例子来理解, 假设图 G 中每一列都有 g 个顶点, G 中没有边时 $N_C \geq g^2$. 假设存在一个带约束的禁忌覆盖数组 A_C 覆盖了全部参数的取值组合, 其规模为 g^2 , 当 G 中有一条边时, A_C 中一定存在至少一行违反了约束, 违反约束的行至少要被扩展成两行才能满足约束, 所以 CAFE(n, G) 的规模至少要增加 1, 因此, 带有一条边的图 G 满足 $N_C \geq g^2 + 1$.

3 试验结果与分析

本文利用禁忌边分解方法给出 5 个典型被测系统的 2 维约束覆盖数组的最小规模的下限, 典型的被测系统的配置如表 4 所示, 这些系统来自于文献 [11] 和 [16]. 现有算法生成的 2 维约束覆盖数组的规模如表 5 所示. 禁忌边分解方法给出的最小规模的下限如表 6 所示. 通过与生成的约束覆盖数组的规模进行比较, 可以看出除了 HSS 为第 2 个被测系统生成的规模 16 小于下限 17, 其余结果都大于等于给出的下限. 小于基于禁忌边分解方法给出的最小规模下限的约束覆盖数组一定不存在, 可通过实际的生成过程来说明.

表 4 被测系统的配置
Tab. 4 Configurations of the systems under test

序号	被测系统
1	$CCA(n; 2, 3^3, F\{(x, 2, 0)(x, 1, 0)(0, x, 1)(2, 0, x)(2, x, 2)(1, 2, 2)\})$
2	$CCA(n; 2, 4^3, F\{(0, 1, x)(2, x, 3)(3, 3, 0)(2, 1, x)\})$
3	$CCA(n; 2, 5^3, F\{(1, 1, x)(4, x, 2)(4, x, 4)(4, 3, 1)(4, 2, x)(1, 3, x)\})$
4	$CCA(n; 2, 6^3, F\{(3, 5, x)(x, 3, 4)(2, 0, x)(x, 1, 2)(3, x, 1)(x, 3, 1)(5, 4, 4)\})$
5	$CCA(n; 2, 7^3, F\{(x, 0, 5)(5, 5, 3)(4, x, 0)(6, 4, x)(1, 4, x)(6, 3, x)\})$

表 5 现有算法生成的 2 维约束覆盖数组的规模

Tab. 5 Sizes of the 2-way constrained covering arrays generated by the existing algorithms

方法	被测系统				
	1	2	3	4	5
HSS	10	16	26	36	51
SA_SAT	10	17	26	36	52
mAETG_SAT	10	17	26	37	52
PICT	10	19	27	39	56
TestCover	10	17	30	38	54
IPOG	11	17	28	39	55
IPOG-F	10	18	28	38	54
CTWT	10	17	27	38	52

表 6 禁忌边分解方法给出的最小规模的下限

Tab. 6 Lower bounds of the minimum sizes obtained by the forbidden edge decomposing method

被测系统	1	2	3	4	5
最小规模下限	10	17	25	36	50
生成规模	10	16	26	36	51

图 4 是第 2 个被测系统参数及取值配置对应的图,图 5 和图 6 是选取其中第 1 个参数的值 0 和第 2 个参数的值 1 构成的两个顶点子图. 对应这两个顶点子图,能得到表 7 和表 8 给出的 $\text{CAFE}(4, G_{1,0})$ 和 $\text{CAFE}(4, G_{2,1})$, 记为 $A_{G_{1,0}}$ 和 $A_{G_{2,1}}$, 其规模都为 4, 表中覆盖了第 1 个参数和第 2 个参数的 5 个取值组合,但是仍有 9 个取值组合没有被覆盖,如表 9 所示. 所以可得出 $N_c \geq 4 + 4 + 9 = 17$, 即第 2 个被测系统的约束覆盖数组规模的最小值不能小于 17. 违反禁忌边分解方法给出的最小规模的下限时,覆盖数组一定不存在,没有违反也不一定能够说明约束覆盖数组不存在,此时只有得到实际生成的约束覆盖数组才能进行验证.

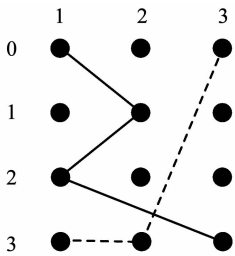


图 4 第 2 个被测系统参数及取值配置对应

Fig. 4 Graph of the parameters and value configurations of the 2nd system under test

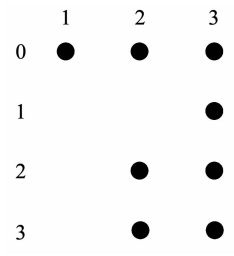


图 5 第 2 个被测系统的 $G_{1,0}$

Fig. 5 $G_{1,0}$ of the 2nd system under test

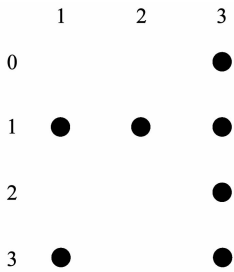


图 6 第 2 个被测系统的 $G_{2,1}$

Fig. 6 $G_{2,1}$ of the 2nd system under test

表 7 第 2 个被测系统的 $\text{CAFE}(4, G_{1,0})$

Tab. 7 $\text{CAFE}(4, G_{1,0})$ of the 2nd system under test

序号	参数 1	参数 2	参数 3
1	0	0	0
2	0	2	1
3	0	3	2
4	0	3	3

表 8 第 2 个被测系统的 $\text{CAFE}(4, G_{2,1})$

Tab. 8 $\text{CAFE}(4, G_{2,1})$ of the 2nd system under test

序号	参数 1	参数 2	参数 3
1	1	1	0
2	3	1	1
3	3	1	2
4	3	1	3

表 9 第 2 个被测系统的未被覆盖的取值组合

Tab. 9 Uncovered value combinations of the 2nd system under test

序号	参数 1	参数 2
1	1	0
2	1	2
3	1	3
4	2	0
5	2	2
6	2	3
7	3	0
8	3	2
9	3	3

4 结 论

本文提出了一种分析 2 维约束覆盖数组最小规模的禁忌边分解方法. 通过将图分解成两个顶点子图, 生成两个顶点带有禁忌边的覆盖数组, 计算其规模和剩余需要覆盖的取值组合数, 提升了 2 维约束覆盖数组的最小规模. 该方法能够得到提升的 2 维约束覆盖数组的最小规模, 更逼近真实值. 生成的最小规模可以用于评价现有算法生成的 2 维约束覆盖数组, 有助于判断其是否真实存在.

参考文献

- [1] KACKER R N, KUHN D R, LEI Yu, et al. Combinatorial testing for software: An adaptation of design of experiments [J]. Measurement, 2013, 46 (9): 3745. DOI: 10. 1016/j. measurement. 2013. 02. 021
- [2] 聂长海. 组合测试研究进展[J]. 中国科技论文, 2017, 12(20): 2391
NIE Changhai. The latest research development of combinatorial testing[J]. China Science Paper, 2017, 12(20): 2391. DOI: 10. 3969/j. issn. 1002 - 137X. 2010. 03. 001
- [3] KUHN D R, BRYCE R C, DUAN Feng, et al. Combinatorial testing: Theory and practice[J]. Advances in Computers, 2015, 99: 1. DOI: 10. 1016/bs. adcom. 2015. 05. 003
- [4] ZHANG Jian, ZHANG Zhiqiang, MA Feifei. Automatic generation of combinatorial test data[M]. Berlin and Heidelberg, Germany: Springer, Berlin, Heidelberg, 2014: 1. DOI: 10. 1007/978 - 3 - 662 - 43429 - 1
- [5] KUHN D R, WALLACE D R, GALLO A M. Software fault interactions and implications for software testing [J]. IEEE Transactions on Software Engineering, 2004, 30 (6): 418. DOI: 10. 1109/TSE. 2004. 24
- [6] AHMED B S, ZAMLI K Z, AFZAL W, et al. Constrained interaction testing: A systematic literature study[J]. IEEE Access, 2017, 5: 25706. DOI: 10. 1109/ACCESS. 2017. 2771562
- [7] YU Libin, DUAN Feng, LEI Yu, et al. Constraint handling in combinatorial test generation using forbidden tuples [C]// Proceedings of the 2015 IEEE Eighth International Conference on Software Testing, Verification and Validation Workshops. Washington DC, USA: IEEE Computer Society, 2015: 1. DOI: 10. 1109/ICSTW. 2015. 7107441
- [8] YAMADA A, BIERE A, ARTHO C, et al. Greedy combinatorial test case generation using unsatisfiable cores[C]//Proceedings of the 31st IEEE/ACM International Conference on Automated Software Engineering. New York, USA: ACM New York, 2016: 614. DOI: 10. 1145/2970276. 2970335
- [9] DANZIGER P, MENDELSON E, MOURA L, et al. Covering arrays avoiding forbidden edges[J]. Theoretical Computer Science, 2009, 410(52): 5403. DOI: 10. 1016/j. tcs. 2009. 07. 057
- [10] MALTAIS E, MOURA L. Finding the best CAFE is NP-hard [C]//Proceedings of the 9th Latin American Conference on Theoretical Informatics. Berlin, Germany: Springer-Verlag, 2010: 356. DOI: 10. 1007/978 - 3 - 642 - 12200 - 2_32
- [11] ALSEWARI A R A, ZAMLI K Z. Design and implementation of a harmony-search-based variable-strength t-way testing strategy with constraints support[J]. Information and Software Technology, 2012, 54(6): 553. DOI: 10. 1016/j. infsof. 2012. 01. 002
- [12] GARVIN B J, COHEN M B, DWYER M B. Evaluating improvements to a meta-heuristic search for constrained interaction testing[J]. Empirical Software Engineering, 2011, 16(1): 61. DOI: 10. 1007/s10664 - 010 - 9135 - 7
- [13] COHEN M B, DWYER M B, SHI Jiangfan. Constructing interaction test suites for highly-configurable systems in the presence of constraints: A greedy approach [J]. IEEE Transactions on Software Engineering, 2008, 34(5): 633. DOI: 10. 1109/TSE. 2008. 50
- [14] LEI Yu, KACKER R, KUHN D R, et al. IPOG: A general strategy for t-way software testing [C]//Proceedings of the 14th Annual IEEE International Conference and Workshops on the Engineering of Computer-Based Systems. Washington DC, USA: IEEE Computer Society, 2007: 549. DOI: 10. 1109/ECBS. 2007. 47
- [15] MICHAEL F, JIM L, YU Lei, et al. Refining the in-parameter-order strategy for constructing covering arrays [J]. Journal of Research of the National Institute of Standards and Technology, 2008, 113(5): 287. DOI: 10. 6028/jres. 113. 022
- [16] LI Longshu, CUI Yingxia, YANG Yun. Combinatorial test cases with constraints in software systems[C]// Proceedings of the 2012 IEEE 16th International Conference on Computer Supported Cooperative Work in Design. New York, USA: IEEE, 2012: 195. DOI: 10. 1109/CSCWD. 2012. 6221818
- [17] GALINIER P, KPODJEDO S, ANTONIOL G. A penalty-based Tabu search for constrained covering arrays[C]//Proceedings of the Genetic and Evolutionary Computation Conference. New York, USA: ACM New York, 2017: 1288. DOI: 10. 1145/3071178. 3071324

(编辑 苗秀芝)