

基于 K -均值聚类粒子群优化算法的组合测试数据生成

潘 烁*, 王曙燕, 孙家泽

(西安邮电学院 计算机学院, 西安 710061)

(* 通信作者电子邮箱 leyoyo426@gmail.com)

摘 要:在解决组合测试中的测试数据集生成问题时,粒子群优化算法(PSO)在待测数据量增加达到一定程度以后,出现迭代次数增加、收敛速度减慢的缺点。针对该问题,提出了一种应用于组合测试数据集生成问题的基于 K -均值聚类的粒子群优化算法。通过对测试数据集进行聚类分区域,增强测试数据集的多态性,从而对粒子群优化算法进行改进,增加各个区域内粒子之间的影响力。典型案例实验表明该方法在保证覆盖度的情况下具有一定的优势和特点。

关键词:组合测试;粒子群优化算法; K -均值聚类算法;测试数据

中图分类号: TP311.55 ; TP181 **文献标志码:** A

Test data generation based on K -means clustering and particle swarm optimization

PAN Shuo*, WANG Shu-yan, SUN Jia-ze

(School of Computer Science and Technology, Xi'an University of Posts and Telecommunications, Xi'an Shaanxi 710061, China)

Abstract: To solve the problem of the test data set generation in combinatorial test, if the software under test has a great many factors and values, the traditional Particle Swarm Optimization (PSO) will have large iteration times and slow convergence velocity. A test data set generation method based on K -means clustering algorithm and PSO has been proposed. The polymorphism of the test data set has been enhanced, though clustering and partitioning the test data set. And it makes PSO has been improved. The compact between the particles in each area has been promoted. Several typical cases show that this method has some merits while ensuring the coverage.

Key words: combinatorial test; Particle Swarm Optimization (PSO) algorithm; K -means clustering algorithm; test data

0 引言

软件测试作为软件质量保证的重要手段,在整个软件生命周期中占有重要的地位。而测试数据生成是软件测试过程中的重要环节,现代软件测试数据的数量庞大,若使用穷尽测试,则耗费大量的时间且效率低下。目前组合测试研究的焦点问题是生成数量少质量高的测试数据集问题,即根据具体待测软件,用尽量少的测试数据尽可能多地覆盖软件系统中的各个因素以及相应组合,以便在对软件系统的检测时发现系统中各个因素之间的相互作用对系统所产生的影响。

2002 年 Kuhn 等^[1]研究了组合测试的可用性发现,大约 70% 的软件故障是由一个或两个因素的相互作用引发的,因而研究两两组合覆盖测试具有重要的意义。近年来研究者使用贪心算法、代数分析法和启发式搜索算法解决组合测试数据生成问题。2006 年 Microsoft 发布了工具 PICT 是一种基于 one-test-at-a-time 策略的,可以有效地按照两两组合测试的原理进行测试数据的设计;而后由 Lei 等提出了逐因素扩展(in-parameter-order)的方式实现二维组合测试数据的生成,并且基于该方法开发了名为 PairTest 的二维组合测试数据生成工具^[2],该方法为以后的研究者进行多维的组合测试数据生成提供了新思路。2009 年,McCaffrey 将遗传算法以及蚁群算法应用于组合测试数据集生成问题^[3-4],将现代群体智能算法引入,对测试数据生成的时间效率和寻优能力上有了很大提高。2010 年查日军等^[5]提出组合测试数据生成的交叉熵与粒子群方法,该方法简单地使用了传统粒子群算法取得了一

定的效果。近年来研究者将多种算法与粒子群优化算法相结合,对粒子群算法进行改进,例如将粒子群算法与遗传算法结合,粒子群算法与模拟退火算法混合等等。这些算法对于传统粒子群优化算法的各个方面进行改进,以满足应用问题。本文将 K -均值聚类算法结合粒子群优化算法(Particle Swarm Optimization, PSO)应用于两两覆盖的组合测试数据集的生成,实验表明该方法对于组合测试数据集生成问题的迭代次数,以及收敛速度上具有优势和改善。

1 组合覆盖测试模型

假设影响待测软件(Software Under Test, SUT)的因素共有 n 个,形成有限集合 $F = (f_1, f_2, \dots, f_n)$,其中因素 f_i 在有限离散集合中包含 k_i 个可选取值,从而形成该因素的取值集合 $X_i (1 \leq i \leq n)$,不妨设 $X_i = \{X_{i1}, X_{i2}, \dots, X_{ik_i}\}$ 。称一个 n 元组 $test = (x_1, x_2, \dots, x_n) (x_1 \in X_1, x_2 \in X_2, \dots, x_n \in X_n)$ 为 SUT 的一条测试数据,相应的,称一个含有多个这样的 n 元组所构成的集合为 SUT 的一个测试数据集。

待测软件系统 SUT 中任意两个因素对可以用二元关系矩阵 $A = (a_{ij})_{K \times K}$ 表示,其中, K 表示待测系统的 n 个因素的全部取值总个数,即因素集合 F 的各个因素取值个数相加和,对各个取值进行编号为 $1, 2, \dots, K$, i, j 为各个取值的编号, a_{ij} 的值表示第 i 个离散值和第 j 个离散值覆盖对是否要覆盖, $a_{ij} = 1$ 表明第 i 个离散值和第 j 个离散值对需要覆盖, $a_{ij} = 0$ 表明第 i 个值和第 j 个值对不需要覆盖,不需要覆盖的原因表示是

收稿日期:2011-09-21;修回日期:2011-11-17。 基金项目:国家自然科学基金资助项目(61050003)。

作者简介:潘烁(1987-),女,河北鹿泉人,硕士研究生,主要研究方向:软件测试; 王曙燕(1964-),女,河南南阳人,教授,博士,CCF 会员,主要研究方向:软件测试、数据挖掘; 孙家泽(1980-),男,河南南阳人,讲师,博士研究生,主要研究方向:软件测试、智能信息处理。

同一因素内部的取值对,或者是组合对为已有组合对的逆转重复。

两两覆盖的组合测试数据集生成问题就是找到覆盖所有因素取值的两两组合对的规模最小的测试数据集,在上述模型中找到覆盖二元关系矩阵 $T = (t_{ij})_{p \times p}$ 所有两两组合对的规模最小的集合,以便在保证错误检测能力的前提下尽可能地降低测试成本。可以证明,两两覆盖的组合测试数据集生成问题是一个 NP-C 问题^[6]。在实际的测试中,人们一般都利用启发式算法、贪心算法和一些数学代数等方法近似求解。近几年,研究者将几种仿生物学的算法应用于生成两两组合测试数据集。本文在粒子群优化算法的基础上结合模糊 K -均值聚类算法将算法应用于组合测试数据集生成问题。算法在每次迭代过程中首先通过 K -均值算法把粒子群体分成若干个子群体,然后粒子群中的粒子根据其个体极值和子群中的最优粒子更新自己的速度和位置值。采用一次产生一个测试数据集的方法,从整体评价一个测试数据集的优劣来指导生成测试数据集。

2 基于 K -均值的粒子群优化算法

K -均值算法是根据函数准则进行分类的聚类算法^[7],这种算法先选择 K 个聚类中心,然后根据聚类准则对 K 个中心反复修改。

虽然 K -均值算法的每步迭代都沿着好的方向进行,但这种基于梯度下降的 K -均值算法本质上是一种局部搜索算法,容易陷入局部极小值,对于聚类样本数量较大的情况更明显。而且 K -均值算法所能找到的最优解对初始值也有很大的依赖性,如初始分类情况对最终的聚类效果有很大的影响。而基于种群操作的粒子群优化算法初始为均匀分布在解空间中的若干可能解,具有很强的全局搜索能力,而基本粒子群优化算法虽然能够收敛到全局最小值附近,但是其效率不高,并且迭代次数要求高。而且由于粒子群优化算法固有的缺点,在进化过程中无可避免地产生了退化的可能,本文将两者相结合,利用 K -均值聚类将粒子群的中粒子划分为多个子群,然后使用粒子群优化算法,增加粒子更新对各自粒子所属区域的依赖程度,从而增强了粒子群优化算法的全局寻优能力,加快了算法的进化速度,提高了收敛精度。

算法描述如下:

1) 设聚类中心 K 个,总粒子数为 $n(K < n)$ 。首先任选 K 个粒子为 K 个聚类中心,记为 $Z_1(1), Z_2(1), \dots, Z_K(1)$ 。粒子群表示为 $X = (X_1, X_2, \dots, X_n)$, 相应的飞行速度表示为 $V = (V_1, V_2, \dots, V_n)$ 。 X_i 和 V_i 分别为第 i 个粒子的位置和速度。

2) 随机选取粒子群中的 K 个粒子作为聚类中心。

3) 计算每个粒子 X_i 到 K 个聚类中心的欧氏距离,并按最近邻规则聚类。具体计算如下:若:

$$\|X_i - Z_j(1)\| = \min\{\|X_i - Z_k(1)\|, k = 1, 2, \dots, K\} \quad (4)$$

则:

$$X_i \in S_j(1) \quad (5)$$

根据聚类准则重新计算各类聚类中心。设 $S_j(1)$ 类有 n_j 个样本,新聚类中心为 $Z_j(2)$, 则:

$$Z_j(2) = \frac{1}{n_j} \sum_{X_i \in S_j(1)} X_i; j = 1, 2, \dots, K$$

第 $M+1$ 次计算:若第 M 次计算得到的聚类中心为

$Z_1(M), Z_2(M), \dots, Z_K(M)$ 。根据最近邻规则对每个粒子进行分类。若:

$$\|X_i - Z_j(M)\| = \min\{\|X_i - Z_k(M)\|, k = 1, 2, \dots, K\}$$

则 $X_i \in S_j(M)$ 。

重新计算各类的聚类中心,其中 S_j 类的聚类中心按式 (6) 计算:

$$Z_j(M+1) = \frac{1}{N_j} \sum_{X_i \in S_j(M)} X_i; j = 1, 2, \dots, K \quad (6)$$

若 $Z_j(M+1) = Z_j(M)$ 对所有 $j = 1, 2, \dots, K$ 成立,则算法结束,否则回到第 2) 步,重新对所有粒子进行分类。直到聚类中心不变,停止聚类分区过程。

4) 根据式 (7)、(8) 更新粒子群中各类的局部最优 $gBest_k(k = 1, 2, \dots, K)$, 每个粒子的个体最优位置 $pBest_i(i = 1, 2, \dots, N)$:

$$pBest_i(t+1) = \begin{cases} X_i(t+1), & X_i(t+1) \geq pBest_i(t) \\ pBest_i(t), & X_i(t+1) < pBest_i(t) \end{cases} \quad (7)$$

$$gBest_j(t+1) = \max(pBest_i(t+1))$$

其中:

$$X_i \in S_j(M); j = 1, 2, \dots, K \quad (8)$$

5) 根据粒子群优化算法中的式 (9)、(10) 更新每个粒子的速度 V_i 和位置 X_i :

$$V_i^{t+1} = wV_i^t + c_1r_1(pBest_i^t - X_i^t) + c_2r_2(gBest_i^t - X_i^t)$$

其中若:

$$X_i \in S_j(K) (j = 1, 2, \dots, K), \text{ 则 } k = j \quad (9)$$

$$X_i^{t+1} = X_i^t + V_i^{t+1} \quad (10)$$

其中: c_1, c_2 为两个学习因子, r_1, r_2 为两个均匀分布在 (0,1) 间的随机数, $i = 1, 2, \dots, n, w$ 为惯性权重。

6) 判断当前粒子是否为满意解,若为满意解,则停止;否则返回步骤 3)。

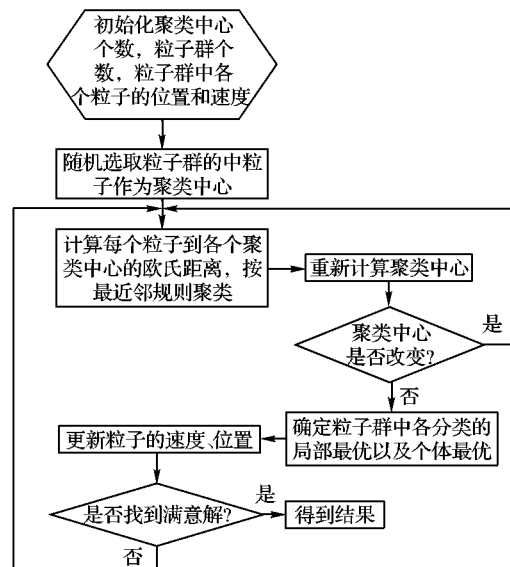


图 1 基于 K -均值聚类的粒子群优化算法流程

3 基于 K -PSO 的组合测试数据生成

针对基本粒子群的效率不高迭代次数要求高的问题,本文使用 K -PSO 算法用于测试数据集生成问题,避免粒子朝向单一粒子进行更新操作,增强粒子的多态性,使用通过聚类将粒子群分区,然后分别对每个区域的“子群”进行更新迭代的操作。由于在算法中进行了粒子群的分区,使得每个粒子对

于所属区域的更新的参考认知增大,这样更容易找到全局最小值附近,同时减少了迭代次数。

基于K-均值聚类的粒子群优化算法首先根据K-均值算法把粒子群分成K个“子群”,求出每个“子群”的最优位置 $sp_i (i = 1, 2, \dots, K)$ (K 为聚类区的个数),然后粒子群中的粒子根据其个体极值和每个“子群”中的最优位置 sp_i 更新自己的速度和位置^[10]。

将K-均值聚类的基本步骤和粒子群优化的思想相结合,可得基于K-均值聚类粒子群算法的两两覆盖组合软件测试数据集生成过程描述:

1) 对待测软件系统 SUT,在粒子群中每个粒子代表一个生成的测试数据。确定参数:聚类类别数 K ,群体规模 N ,且 $2 \leq K < N$,学习因子 c_1, c_2 ,以及最到迭代代数 G_{\max} ,影响系统的因素;

2) 初始化时,随机生成 N 个粒子群位置 $X_i^0 = (x_{i1}^0, x_{i2}^0, \dots, x_{id}^0) (i = 1, 2, \dots, N)$ 和其速度 $V_i^0 = (v_{i1}^0, v_{i2}^0, \dots, v_{id}^0) (i = 1, 2, \dots, N)$, d 表示待测软件系统 SUT 的因素个数。

3) 根据K-均值聚类算法,从已有的粒子群中随机选取 K 个聚类中心。将聚类中心作为一个测试数据集(一个例子代表一个测试数据)用 $T = \{ \langle t_{11}, t_{12}, \dots, t_{1d} \rangle, \langle t_{21}, t_{22}, \dots, t_{2d} \rangle, \dots, \langle t_{q1}, t_{q2}, \dots, t_{qd} \rangle \}$ 表示。其中 t_{ij} 表示生成的测试数据集中第 i 测试数据的第 j 个测试因素的取值,即 $t_{ij} \in P_j$; q 表示要生成的测试数据集中的测试数据的数量($q = K$), $P_j = \{p_{j1}, p_{j2}, \dots, p_{jm}\}$ 表示第 j 个测试因素的有 m 个取值,分别是 $p_{j1}, p_{j2}, \dots, p_{jm}$ 。

4) 根据K-均值算法式(4)、(5)计算所有粒子的所属区域,并根据式(6)计算新的聚类中心 $Z_j(M+1) (j = 1, 2, \dots, K)$,若 $Z_j(M+1) = Z_j(M)$ 对所有 $j = 1, 2, \dots, K$ 成立,则转入5),否则回到4),重新对所有粒子进行分类。

5) 根据所确定的聚类区域分别计算每个区域的 $gBest_k (k = 1, 2, \dots, K)$,每个粒子的 $pBest_i (i = 1, 2, \dots, N)$;并且根据式(9)、(10)更新粒子群的最优 $pBest, gBest$;

6) 根据粒子群优化算法中的式(7)、(8)更新每个粒子的速度 V 和位置 X ;

7) 判断是否满足收敛条件,满足停止;不满足,返回4)。这里的收敛条件即为找到的粒子组成的集合对两两因素组合的覆盖度达到标准或者迭代到一定代数。

4 实例研究与分析

为了测试基于K-均值聚类粒子群优化算法的求解两两覆盖组合测试数据集生成问题的性能,我们进行实例分析。对于粒子群的速度更新式(7)的参数,本文采用文献[11]推荐,取 $c_1 = c_2 = 1.49, w = 0.729$ 。

实验1 测试数据集规模比较

本文的算法K-PSO和现有的两两覆盖组合软件测试数据集生成方法AETG, PAIRTEST, NetWork, PSST, SA, GA, ACA, CE, PSO实验比较结果如表1所示。

表格待测软件系统SUT中测试数据集规模解释:以问题 $(5^3 \times 4^4 \times 3^1 \times 2^2)$ 为例,表示待测软件系统SUT共有10个因素,其中有3个因素的取值个数为5个,有4个因素的取值个数为4个,有1个因素的取值个数为3个,有2个因素的取值个数为2个。

表1 不同算法测试数据集规模比较

算法	测试数据个数				
	3^4	3^{13}	$3^{12} \times 4^5$	$4^1 \times 3^{39} \times 2^{35}$	$5^3 \times 4^4 \times 3^1 \times 2^2$
AETG ^[5]	9	15	31	28	31
PAIRTEST ^[5]	9	19	29	29	37
NetWork ^[5]	9	19	26	49	34
PSST ^[5]	9	20	28	30	33
SA ^[5]	9	16	NA	21	NA
GA ^[5]	9	17	NA	27	NA
ACA ^[5]	9	17	NA	27	NA
PSO ^[5]	9	17	29	32	33
K-PSO	9	16	27	27	32

从表1中看出K-PSO方法生成的测试数据集和其他方法生成的测试数据集规模总体相当,在某些情况下生成的数据规模较小,如 $3^{12} \times 4^5$ 和 $4^1 \times 3^{39} \times 2^{35}$ 。对每组数据进行50次实验,其中60%以上的结果都可以达到优值,约35%的结果可以达到与优值比较偏差1到3测试数据。

实验2 算法迭代次数比较

该待测软件系统SUT中测试数据集规模为 $(4^1 \times 3^1 \times 2^2)$,即有13个待测因素,每个因素的取值个数都为3。可以假设某待测软件系统SUT中的二元关系矩阵 $A = (a_{ij})_{11 \times 11}$ 表示如图2,可以看出这个数据集的两两组合对有44对。

0	0	1	1	1	1	1	1	1	1	1
0	0	1	1	1	1	1	1	1	1	1
0	0	0	0	0	0	1	1	1	1	1
0	0	0	0	0	0	1	1	1	1	1
0	0	0	0	0	0	1	1	1	1	1
0	0	0	0	0	0	1	1	1	1	1
0	0	0	0	0	0	0	0	0	1	1
0	0	0	0	0	0	0	0	0	1	1
0	0	0	0	0	0	0	0	0	1	1
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0

图2 实验待测软件系统的二元关系矩阵

使用基于K-均值粒子群优化算法的测试数据集生成方法对该问题求解执行50次,初始值每次随机产生,50次要求生成12个测试数据的结果中39次覆盖了所有的两两覆盖,6次生成一个组合没有覆盖到,4次有两个组合没有被覆盖到,1次有3个组合没有覆盖到。每次生成的测试数据集的具体内容可能不一样,但个数都达到了最少,如图3所示。

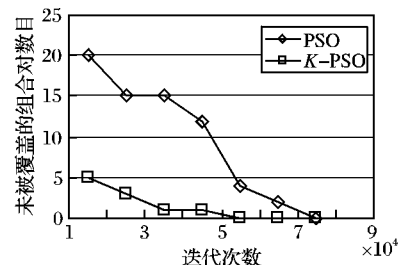


图3 算法测试数据集迭代比较

从图3可以看出,PSO算法的测试数据集在60000至70000次的迭代中就找到了最优集合。而将K-均值聚类算法与PSO算法相结合所产生的测试数据集在40000~50000次的迭代中就找到了最优集合。并且由于算法在前10000次迭代中就可以将覆盖对的组合数目缩小至5以内,也就是全部

(下转第1175页)

间隔较大,致使三维模型不能完全反映坡体和滑坡的真实形状。另外,本文采用文献[9]中提出的基于GIS空间数据库的三维边坡稳定分析方法对同样的数据进行了分析计算实验,两种方法计算结果如表1所示。

表1 两种方法计算情况比较

方法	计算结果	计算时间/s
文献[9]方法	0.42	3.58
本文方法	0.38	1.24

实验表明,在相同数据精度的前提下,采用文献[9]中的方法计算得到的结果和本文方法基本一致,但在时间消耗上是本文方法的3倍。上述结果说明本文中提出的近似计算方法不仅是一种可靠的评估方法,而且可以有效提高系统的运行效率。

3 结语

本文提出了一种基于DEM数据的坡面稳定性三维分析方法。该方法设定初始滑动面为球面或椭球面,从而根据球体或椭球体方程来构建滑体模型,并在其中搜寻稳定性系数最小的滑面为危险滑面,确定危险区域。计算过程中则通过DEM网格对滑体进行剖分,对每一个剖面进行二维计算,然后积分求和,得到近似三维结果,这样的计算过程避免了对三维滑体的整体求解过程,简化了三维滑体的计算步骤。应用结果表明该方法可以方便地研究多种可能情况下的稳定性,计算简捷快速,结果有效可靠,为滑坡灾害分析提供了一种新的方法。由于分析结果的精度和DEM的取样间隔成反比(间隔越小,精度越高),因此要取得好的分析结果,还需要大

比例尺的DEM数据(DEM取样间隔在1m左右)支持。

参考文献:

- [1] 王根龙, 门玉明, 陈志新, 等. 基于等分圆弧滑面土质边坡的稳定性极限分析方法[J]. 地球科学与环境学报, 2007, 29(2): 178 - 182.
- [2] 王根龙, 伍法权, 李巨文. 折线型滑面边坡稳定系数计算的极限分析上限解[J]. 水文地质工程地质, 2007, 34(1): 62 - 65.
- [3] 郭明伟, 李春光, 葛修润, 等. 基于矢量和分析方法的边坡滑面搜索[J]. 岩土力学, 2009, 30(6): 1775 - 1781.
- [4] 周海清, 刘东升, 陈正汉, 等. 基于传递系数法的滑面指标反算方法的研究[J]. 地下空间与工程学报, 2010, 6(6): 1161 - 1167.
- [5] 张有天. 岩石高边坡的变形与稳定[M]. 北京: 中国水利水电出版社, 1999: 12 - 13.
- [6] HOVLAND H J. Three-dimensional slope stability analysis method [J]. Journal of the Geotechnical Engineering Division, 1977, 103(9): 971 - 986.
- [7] CHEN Z, MORGENSTERN N R. Extension to the generalized method of slices for stability analysis [J]. Canadian Geotechnical Journal, 1983, 20(1): 104 - 119.
- [8] LAM L, FREDLUND D G. A general limit equilibrium model for three-dimensional slope stability analysis [J]. Canadian Geotechnical Journal, 1993, 30(6): 905 - 919.
- [9] 谢谟文, 江崎哲郎, 周国云, 等. 基于GIS空间数据库的三维边坡稳定性分析[J]. 岩石力学与工程学报, 2002, 21(10): 1494 - 1499.
- [10] 谢谟文, 江崎哲郎, 周国云, 等. 基于边坡单元的三维滑坡灾害评价的GIS方法[J]. 岩石力学与工程学报, 2003, 22(6): 969 - 976.

(上接第1167页)

两两组合对的1/10左右。这样,使用这个方法,如果要求数目的组合测试数据集少于完全覆盖所有两两组合对的测试数据集数目时,就可以在更早的时间结束程序,提高算法效率。

5 结语

通过以上分析和实例验证表明,本文提出的基于K-均值聚类粒子群优化算法的测试数据集生成方法与已有的组合测试数据集生成方法相比,本方法的粒子群采用随机方式初始化,测试数据集生成结果与初始值无关;将K-均值算法与粒子群优化算法相结合,避免了粒子向着迄今为止找到的单一最优粒子吸引,从而在一定程度上保持了粒子的多样性,减少迭代次数。此外,方法中以多个子群的方式进行迭代更新使得粒子群中的粒子容易摆脱局部极值,算法的收敛速度加快。

参考文献:

- [1] KUHN D R, REILLY M J. An investigation of the applicability of design of experiments to software testing [C]// Proceedings of the Annual NASA/IEEE Software Engineering Workshop (SEW). Los Alamitos: IEEE Press, 2002: 91 - 95.
- [2] LEI Y, TAI K C. In-parameter-Order: A test generation strategy for pairwise testing [C] // HASE'98: The 3rd IEEE International Symposium on High-Assurance Systems Engineering. Washington, DC: IEEE Computer Society, 1998: 254 - 261.
- [3] McCaffrey J D. Generation of pairwise test sets using a genetic

algorithm [C] // Proceedings of the 2009 33rd Annual IEEE International Computer Software and Applications Conference. [S. l.]: IEEE, 2009: 625 - 628.

- [4] McCaffrey J D. Generation of pairwise test sets using a simulated bee colony algorithm [C]// IRI'09: Proceedings of the 10th IEEE International Conference on Information Reuse & Integration. Piscataway: IEEE, 2009: 115 - 120.
- [5] 查日军, 张德平, 聂长海, 等. 组合测试数据生成的交叉熵与粒子群算法比较[J]. 计算机学报, 2010, 33(10): 1896 - 1908.
- [6] COLBOURN C J, COHEN M B, TURBAN R C. A deterministic density algorithm for pairwise interaction coverage [C]// SE'04: Proceedings of IASTED International Conference on Software Engineering. Innsbruck: [s. n.], 2004: 345 - 352.
- [7] 李晶皎, 赵丽红, 王爱侠. 模式识别[M]. 北京: 电子工业出版社, 2010: 90 - 93.
- [8] KENNEDY J, EBERHART R. Particle swarm optimization [C]// Proceedings of IEEE International Conference on Neural Networks. Perth: IEEE Service Center, 1995: 1942 - 1948.
- [9] SHI Y, EBERHART R C. A modified particle swarm optimizer [EB/OL]. [2011 - 06 - 20]. <http://dsp.szu.edu.cn/psa/ispso/download/a%20modified%20psa.pdf>.
- [10] 高鹰, 谢胜利, 许若宁, 等. 基于聚类的多子群粒子群优化算法[J]. 计算机应用研究, 2006, 23(4): 40 - 41.
- [11] CLER M, KENNEDY J. The particle swarm explosion, stability, and convergence in a multidimensional complex space [J]. IEEE Transactions on Evolutionary Computer, 2002, 6(1): 58 - 73.