

两两组合覆盖测试数据自动生成集成工具的设计与实现^{*}

周晓宇^{1,2} 聂长海^{1,2} 徐宝文^{1,2,3} 陈火旺^{3,4}

(东南大学计算机科学与工程系 南京210096)¹ (江苏省软件质量研究所 南京210096)²

(武汉大学软件工程国家重点实验室 武汉430072)³ (国防科技大学计算机学院 长沙410073)⁴

摘要 两两组合覆盖测试方法是一种科学有效的软件测试方法,相应的测试数据生成算法是人们研究的一个重点问题,本文在人们已有的研究基础上,结合我们在这个方面的研究成果,设计实现了一个两两组合覆盖测试数据自动生成工具,所产生的测试数据集与同类工具相比具有一定的特点和优势。

关键词 软件测试,组合覆盖,测试数据生成

Design and Implement of a Tool for Pairwise Testing Data Generation

ZHOU Xiao-Yu^{1,2} NIE Chang-Hai^{1,2} XU Bao-Wen^{1,2,3} CHEN Huo-Wang^{3,4}

(Department of Computer Science & Engineering, Southeast University, Nanjing 210096)¹

(Jiangsu Institute of Software Quality, Nanjing 210096)²

(State Key Laboratory of Software Engineering, Wuhan University, Wuhan 430072)³

(School of Computers, National University of Defense Technology, Changsha 410073)⁴

Abstract Pairwise testing is practical and effective for various types of software systems, and it is a key problem that the test data generation method for the pairwise testing. This paper introduces the design and implement of a tool for the pairwise test data generation, which is based on the foundation of the former and our recent research results. The experiments in using this tool show that our tool has good properties and merits compared with the related tool.

Keywords Software testing, Combination coverage, Test data generation

1 引言

组合覆盖是一种重要的软件测试方法,这种方法充分考虑了系统中各种因素以及各种因素之间的相互作用可能对系统产生的影响,可以根据实际需要,用尽可能少的测试用例次数尽可能多地覆盖一些影响系统的因素,同时这少数测试的结果能够反映全面测试的内在规律,具有代表性。这种方法对于那些由系统中某些因素相互作用而导致的软件故障具有较强的检测能力。

组合覆盖方法根据覆盖程度的不同可以区分为单因素覆盖、两两组合覆盖、三三组合覆盖等。随着覆盖程度的提高,相应需要的测试用例规模也呈指数迅速增长。综合考虑软件测试的成本、时间、各种因素的组合覆盖程度、测试用例的规模等因素,人们最常用的组合覆盖测试方法是两两组合覆盖方法。

两两组合覆盖方法已经在软件测试领域得到了成功的应用,人们应用这种方法对软件系统进行测试时,发现了很多传统测试方法难以发现的错误。所以,这种方法受到了人们很大的重视。但两两组合覆盖方法的测试数据生成一直是人们研究的重要课题,至今还没有得到很好的解决。

我们在人们已有工作的基础上,结合我们在这个方面的研究结果,开发实现了一个用于两两组合覆盖方法的测试数据自动生成工具,这个工具与国外同类产品相比,在很多方面

具有更好的性能。

2 相关研究工作

人们一开始利用组合覆盖方法进行软件测试时,主要是将正交实验设计方法应用于软件测试,取得了比较好的效果^[1~5]。正交实验设计方法是一种有效的测试用例选择方法,它可以实现对各参数两两组合的等概率覆盖。然而这种方法依赖于正交表,而正交表的构造还存在很多未解决的难题,特别是对于混合型的正交表,目前还没有比较好的构造方法,所以这给正交试验设计在软件测试中的应用带来了较大的局限。通常,在软件测试中,测试用例只需要达到对各参数两两组合覆盖,并不要求等概率^[12];而且正交试验设计产生的测试用例数量往往远远多于两两组合覆盖产生的测试用例数量。

目前,人们关于两两组合覆盖方法的研究主要集中在两两组合覆盖测试数据生成方面。D. M. Cohen 与 S. R. Datal 等提出了一种基于两两组合覆盖的测试数据启发式生成方法,开发了一个用于两两组合覆盖测试用例的自动生成系统: AETG 系统,并申请了专利,在互联网上售卖这种服务^[6,7]。Y. Lei 和 K. C. Tai 利用贪心算法,提出一种基于参数顺序的渐进扩充的两两组合覆盖测试数据生成方法,也开发实现了相应的测试用例自动生成系统: PAIRWISE TEST 系统^[8,9]。这些方法各有各的优点,但都不能解决所有的问题。作

^{*} 本研究得到国家自然科学基金(60373066)、江苏省科技攻关项目(BE2001025)、江苏省三三三人才基金、武汉大学软件工程国家重点实验室开放基金、江苏省计算机信息处理技术重点实验室(苏州大学)开放基金(KJS02009、T8118011)资助。周晓宇 讲师,在职博士生,主要从事程序设计语言、软件分析及软件测试方面的教学与科研工作。聂长海 讲师,博士,主要从事软件工程和软件测试技术、模糊信息处理、神经网络等方面的教学与科研工作。徐宝文 教授,博士,博士生导师,主要从事程序设计语言、软件工程、并行与网络软件等方面的教学与科研工作。

为启发式方法的一种重要补充, N. Kobayashi 和 T. Tsuchiya 等提出了一种代数方法, 在某些情况下其效果要比启发式方法和贪心算法好^[10]。我们也提出了一些算法, 其效果不逊于以上方法, 并具有一些很好的性质。

3 设计与实现

两两组合覆盖测试数据生成问题是一个 NP 完全问题, 由于所有的 NP 完全问题都还没有多项式时间算法, 因此关于这个问题的求解只能利用近似算法、启发式方法、贪心算法等进行近似求解。而且解决这个问题的各种方法具有不同的优点, 有些方法针对一些具体问题可以产生最优解。我们将这些方法集成到我们的工具中, 每次针对具体的需求, 分别生成相应的两两组合覆盖测试数据集, 经过比较, 输出最小的测试数据集(见图1)。

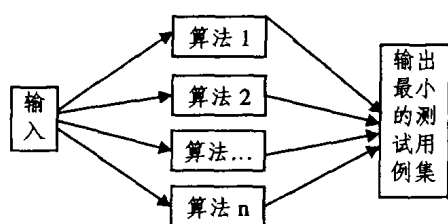


图1 两两组合覆盖测试数据生成工具设计框架

输入部分主要需要输入待测软件系统参数的个数, 每个参数的取值个数。有些情况下, 如果用户需要事先指定一些测试用例, 就把这些事先指定的测试用例按照指定的格式进行输入, 以便在此基础上生成一组最小的两两组合覆盖的测试数据集。

目前我们在吸收已有研究的基础上, 结合我们对这个问题的研究结果, 设计实现了五个算法, 这五个算法互为补充, 生成一个最小的测试数据集。第一个算法实现的是 D. M. Cohen 等在文[6,7]中提出的贪心算法, 该算法每次构造一条测试数据使之最大限度地覆盖没有被覆盖的参数间的两两组合对, 从而产生两两组合覆盖的最小测试数据集; 第二个算法实现了 Y. Lei 等在文[8,9]中提出的基于参数顺序的渐进扩充的两两组合覆盖测试数据生成方法; 第三个算法是对 N. Kobayashi 等提出的代数方法的实现^[10]。

我们将两两组合覆盖测试数据生成问题建立成这样一个模型, 如图2所示: 如果有四个参数, 每个参数有三个取值的待测软件系统生成两两组合覆盖测试数据时, 相当于在图2所示的网络中, 寻求从第一层(每一列为一层)到最后一层最少路线, 实现对网络中不同层上的任意两个结点之间的路径覆盖。基于这个模型, 我们提出了一个有效的两两组合覆盖测试数据生成算法4。对于一些具体情况, 我们研究出一些有效的算法, 例如对于每个参数取值个数同为某素数个时, 用图2所示的模型可以产生正交表, 这个正交表是实现两两组合覆盖的最小测试数据集^[14,15]。

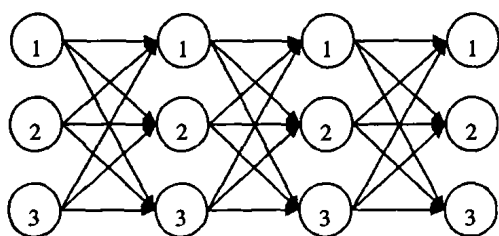


图2

算法5是对于每个参数都取两个值的情况我们研究出的一个有效算法, 可以生成规模相当小的测试数据集, 这个算法只对这种情况有效。在这个工具中我们充分考虑到关于这个问题的认识还处于发展阶段, 当有新的算法产生时可以集成到这个工具中去^[16]。

4 试验结果的比较

以下列出我们开发的算法工具与目前同类算法工具的效果比较, 主要与 AETG 和 PAIRTEST 两个工具进行比较, 结果显示这些工具各有所长, 效果相当, 分别见表1和表2。

表1 测试数据生成工具与 PAIRTEST、AETG 系统的比较

待测系统	S1	S2	S3	S4	S5	S6
PAIRTEST	9	19	36	29	15	218
AETG	9	15	41	28	10	180
Ourtool	9	19	42	30	10	225

其中, S1表示有4个参数, 每个参数有3个值; S2表示有13个参数, 每个参数有3个值; S3表示有61个参数(15个4值, 17个3值, 29个2值参数); S4表示有75个参数(1个4值, 39个3值, 35个2值参数); S5表示有100个参数, 每个2个值; S6表示有20个参数, 每个10个值。

表2 测试数据生成工具与 PAIRTEST 系统的效果比较

参数个数	10	20	30	40	50	60	70	80	90	100
PAIRTEST	31	40	46	49	52	57	60	62	62	66
Ourtool	28	40	40	40	40	40	52	52	52	52

表2中每个参数有4个取值, 第一列表示当待测系统有10参数, 每个参数有四个取值时, 用 PAIRTEST 产生31个测试用例, 而我们的工具只产生28个测试用例。从表2中可以看到我们的工具在有些方面与 PAIRTEST 系统相比, 具有明显的优点。

以上关于 AETG 和 PAIRTEST 的数据都是来自文[6~9]。

5 进一步研究工作

目前关于两两组合覆盖测试数据生成的现有算法尽管取得了很好的效果, 但还存在很多问题, 首先这些算法产生的测试数据都只是最优值的一个近似, 所以需要进一步研究性能更好的算法。

其次, 对于任何一个具体的测试需求, 客观上存在一个规模最小的两两组合覆盖测试数据集, 例如, 设某个待测软件系统有 n 个参数, 每个参数有 t_i 个取值, 不妨设 $t_1 \geq t_2 \geq \dots \geq t_n$, 它的最小两两组合覆盖测试数据集的规模 S 一定满足: $t_1 \times t_2 \leq S \leq t_1 \times t_2 \times \dots \times t_n$ 。即如何在完全组合情况下的 $t_1 \times t_2 \times \dots \times t_n$ 个可选数据中选择能够对各参数进行两两组合覆盖的最少的一组测试数据。研究这样一个最小两两组合覆盖测试数据集的构造理论将具有十分重要的意义。对于一些重要的软件系统, 如果测试代价很高时, 即使精简掉一个两个测试数据, 效益也是十分明显的。

结论 在软件测试过程中, 穷尽测试既不可能, 也没有必要, 因此, 如何精选少量的测试用例, 对系统进行有效的测试, 是软件测试研究中的关键课题。选择好的测试用例不仅能减少软件测试工作量, 降低软件测试成本, 而且在不降低软件测试质量的前提下, 提高软件开发的速度, 使系统尽快参与市场竞争。因此, 两两组合覆盖方法是一种科学有效的软件测试方

(下转第154页)

多数情况下,球员 A 可以根据计划将球带到球门25米以内的位置去(可以射门的距离)。具体数据如表1所示。我们可以看到系统具备相当的推理规划能力。

实验2:测试系统在实时环境下的执行效率情况。

该实验通过采样和统计在完整比赛中智能体决策时平均消耗的时间来评估系统在实时环境下的执行效率。实验结果如表2所示。

我们可以看到系统在70%的情况下能够在40ms 以下作出决策,97.5%的情况下能够在100ms 以下。这保证了系统能够在实时环境下有效地工作(RoboCup 仿真平台 Soccer Server 设定的决策周期为100ms)。

实验3:测试系统在动态不确定环境中的整体性能。

我们从去年参加 RoboCup 2002 仿真机器人足球比赛的球队中挑选了一些不同风格球队作为我们的对手,以此来评估该模型在动态、不确定环境中的表现。表3列出了该部分的实验结果。

由实验数据可以看出,该系统模型能够较好地适应像 RoboCup 这样的动态实时领域。

表1

最终距球门距离	目标	0-15m	15-25m	25-35m	35m 以上
达成次数	5	36	28	19	12

表2

平均时间(ms)	0-10	10-40	40-70	70-90	100以上
占决策的百分比	23.7%	46.8%	17.2%	9.8%	2.5%

表3

对手	实验场次	每均平均进球数	每场平均被进球数
Robolog2002	20	6.8	0.0
UvA trilearn 2002	20	1.2	0.8
YowAI2002	20	7.9	0.0
HelliRespina 2002	20	2.5	0.1

(上接第119页)

法,关于这种方法的测试数据自动生成工具的研究具有很重要的意义。

参考文献

- Heller E. Using Design of Experiment Structures to Generate Test Cases. In: Proc. 12th Int'l Conf. Testing Computer Software, ACM, New York, 1995. 33~41
- Mandl R. Orthogonal Latin Squares: An Application of Experimental Design to Compiler Testing. Communications of the ACM, 1985, 28(10): 1054~1058
- Brownlie R, Prowse J, Phadke M. Robust Testing of AT&T PMX/StarMail using OATS. AT&T Technical Journal, 1992, 71(3): 41~47
- Phadke M S. Quality Engineering Using Robust Design. Prentice Hall, Englewood Cliffs, NJ, 1989
- Taguchi G. System of Experimental Design, Quality Resources, 1987. Translation of Jikken keikakuho, Maurzen Co., Tokyo, 1976
- Cohen D M, et al. The AETG System: An Approach to Testing Based on Combinatorial Design. IEEE Trans. on Software Engineering, 1997, 23(7): 437~444
- Cohen D M, et al. The Combinatorial Design Approach to Auto-

总结以及工作展望 实验表明,本文所提出的混和式的智能体结构模型在一定程度上能够在实时反应和规划推理之间取得较好的折衷,因而可以满足动态不确定环境下的智能体决策要求。然而当前我们的工作主要还是集中于单个智能体的决策推理,在未来的工作中,应该进一步完善在这个结构上的多智能体主体的合作问题。

参考文献

- Bratman M E. Intentions, Plans, and Practical Reasons. Harvard University Press, 1987
- Rao, Georgeff M. BDI Agents from Theory to Practice: [Technical Note 56]. AAIL, April 1995
- Rao A S, Georgeff M P. Modeling rational agents within a BDI-architecture. In: Proc. of the Second Intl. Conf. on Principles of Knowledge Representation and Reasoning (KR'91), Morgan Kaufmann, 1991. 473~484
- Kitano H, et al. oboCup: The Robot World Cup Initiative. In: Proc. of the IJCAI-95 Workshop on Entertainment and AI/ALife, 1995
- Kitano H, et al. The Robocup synthetic agent challenge 97. In: Proc. of the Fifteenth Intl. Joint Conf. on Artificial Intelligence, San Francisco, CA, Morgan Kaufman, 1997
- Nilsson N J. Shakey the robot: [Technical Report 223]. SRI International, 1984
- Wooldridge M, Jennings N R. Intelligent agents: Theory and practice. Knowledge Engineering Review, 1995, 10(2)
- Brooks R A. Intelligence Without Reason. In: Proc. IJCAI-91', Sydney, Australia, 1991
- Stuart R. Rationality and intelligence. In: Proc. of the 14th Intl. Joint Conf. on Artificial Intelligence, 1995. 950~957
- Chen Xiaoping. On Deliberative Adaptability. In: Proc. of 2002' National Conf. on Theoretical Computer Science. Also in: Computer Science, 29(9): 49~51
- matic Test Generation. In: IEEE Software Sep. 1996. 83~87
- Lei Y, Tai K C. In-Parameter-Oder: A Test Generation Strategy for Pairwise Testing: [Technical Report TR-2001-03]. Dept. of Computer Science, North Carolina State Univ, Raleigh, North Carolina, Mar. 2001
- Tai K C, Lei Y. A Test Generation Strategy for Pairwise Testing. IEEE Trans. on Software Engineering, 2002, 28(1): 109~111
- Kobayashi N, Tsuchiya T, Kikuno T. A New Method for Constructing Pair-wise Covering Designs for Software Testing. Information Processing Letters, 2002, 81(2): 85~91
- Cohen D M, et al. New Techniques for Designing Qualitatively Independent Systems. J. Combin. Designs, 1998, 6(6): 411~416
- Williams A W, Probert R L. A Practical Strategy for Testing Pair-wise coverage of Network Interfaces. In: Proc. 7th Internat. Symp. Software Reliability Engineer. Nov. 1997. 246~254
- Cohen D M, et al. The Automatic Efficient Tests Generator. In: Fifth Int'l Symposium on Software Reliability Engineering, IEEE, 1994. 303 ~ 309
- Xu Baowen, Nie Changhai, Shi Qunfeng, Lu Hong. An Algorithm for Automatically Generating Black-Box Test Cases. Journal of Electronics, 2003, 20(1): 74~77
- 聂长海, 徐宝文. 基于接口参数的黑箱测试用例自动生成算法. 计算机学报, 2004, 27(3): 382~388