

## 探索性方法在面向故障软件测试中的应用

The application of exploratory method in fault oriented software testing

(湖南大学) 谢经纬 吴昊

XIE Jing-wei WU Hao

**摘要：**传统的面向故障的测试方法存在限制条件高精确程度与低误报率无法兼得的瓶颈效果，而高误报率直接导致了测试成本的增加和效率的低下。本文在面向故障的测试框架下，深入研究了探索性软件测试方法，对该测试过程进行了量化处理，提取出一系列指标和效应函数，作为所采用的用于优化的遗传算法中的迭代条件，进而寻找出在有限测试成本内的最佳检查点组合，最终实现了既定精确程度的条件下测试成本的优化。

**关键词：**探索性测试；面向故障；遗传算法；检查点

**中图分类号：**TP311

**文献标识码：**A

**Abstract:** There is a bottleneck effect between high accuracy of restricted condition and low false alarm rate in traditional testing way of fault oriented. The high false alarm rate directly leads to high cost and low efficiency. In this paper, under the frame of fault oriented, exploratory testing has been researched and quantized, some indexes and functions has been given, as the iterative condition for Genetic Algorithm, which is used for optimizing. Then, optimal combination of alarm points in limited cost is revealed, task of optimizing in established accuracy of restricted condition is completed.

**Key words:** Exploratory Testing; Fault Oriented; Genetic Algorithm; Alarm Point

**1引言：**软件测试是是软件工程的重要组成部分，是保证软件质量、高软件可靠性的关键。软件测试工作的好坏决定着软件产品质量的好坏。近年来，随着软件应用范围的扩大，软件复杂度的提高，以及软件设计技术的不断发展，软件开发规模越来越大，处理的问题愈来愈复杂。软件系统的可靠性也变得更为重要，传统的软件测试技术和方法以及测试工具已无法满足大型的、复杂的软件测试需要。软件测试已成为当前软件技术研究的重点和难点。

探索性软件测试是软件测试领域比较前沿的理论，尤其适用于那些要求在短时间内发现被测软件一些重要缺陷或事先没有能够进行详细测试设计的情况。利用探索性测试，能显著提高软件测试的效率。

面向故障的软件测试方法是指测试是针对某些故障的，而不是传统意义上的针对的是整个软件。因为面向故障的方法在测试之前把测试目标定位到一类具体的故障，所以这些故障具有类似的检测难度。

已经有很多文献把遗传算法应用到软件测试中并取得了较好的结果，但是这些应用大多是基于路径覆盖，条件覆盖以及测试数据生成的。在探索性测试方面，J'org Denzinger把遗传算法应用到系统冗余（错误）行为测试中效果也较明显，他的测试倾向于系统测试，把时间和待测个体（这些个体被看作agent）的动作构建成一个序偶，进而用遗传算法迭代寻找出冗余错误的动作。

传统的面向故障的测试方法存在限制条件高精确程度与低误报率无法兼得的瓶颈效果，而高误报率直接导致了测试成本的增加和效率的低下。本文在面向故障的测试框架下，深入研究了探索性软件测试方法，对该测试过程进行了量化处理，提取出一系列指标和效应函数，作为所采用的用于优化的遗传算法中的迭代条件，进而寻找出在有限测试成本内的最佳检查点组合，最终实现了既定精确程度的条件下测试成本的优化。

**2探索性测试：**探索性测试的创始人James Bach对探索性测试的定义为：了解被测软件，设

计测试用例，执行测试同时进行的软件测试技术。也就是说不进行事先计划和设计的一种特殊类型的测试，由有经验的测试人员根据实际情况产生诸如“我这么测结果会怎样？”的灵感来进行测试，这一方式往往能帮助在测试设计之外发现更多的软件错误。

探索性测试往往是测试人员依据测试任务进行主动的、探索性的测试，这就要求探索性测试的实施具备比传统测试更强的测试设计能力，他要根据测试的实际情况，来及时的设计出测试用例，只不过这个测试不需要形成文档而已，但是要给出测试的结果报告，此外，还需要很强的观察能力，思维能力和调动各种资源的能力，如他人的帮助，工具的使用等，特点概括如下

(1)探索性测试强调测试设计和测试执行的“同时”性。这个“同时”是相对于传统软件测试过程中的“先设计，后执行”来说的。

(2)测试工程师通过测试来不断学习被测试系统。

(3)探索性测试的重点是创造。

运用探索性测试对软件进行测试，首先通过阅读能找到的软件需求和开发文档，学习了解软件的运行环境、了解测试任务、配置测试环境等。准备这些测试计划的同时，根据已经掌握的情况，对被测试系统实施初次的测试用例设计。利用设计出来的测试用例，进行初次的探索性软件测试执行。测试的执行结果，一方面反馈到设计好的测试用例中去，对已有的测试用例进行重新修订；另一方面与原先学习到的系统作比较，加深对原来界定不清问题的理解，进一步的加深对原系统的了解。之后，又可以进一步利用对系统的学习了解成果，设计出新的测试用例，执行测试。如此反复，直至完成原先设定的测试目标。

**3面向故障的软件测试：**面向故障的软件测试方法是指测试是针对某些故障的，而不是传统意义上的针对的是整个软件。因为程序是软件的具体表现，所以为了便于测试，其定义的故障模型必须落实到程序上，也就是说，故障是程序中某个或某几个程序的错误。定义在程序中的故障必须满足下列几个条件：

(1)故障必须是和实际是对应的。也就是说，实际软件中存在大量这种故障，而且所提出的故障模型能覆盖大多数实际存在的故障。

(2)故障必须能引起错误。一旦该故障被执行且产生的变异能传播到程序中的某个输出，则程序的结果必然是错误的，或者说程序的行为是错误的。

(3)故障的个数是可以容忍的。故障的个数一般和程序的规模是线性关系，个数太多，难以实现，以至无法实用。

#### 4探索性方法的建模和量化：

Andy Tinkham和Cem Kaner, J.D., Ph.D.分析了影响探测型测试人员测试效率的因素，包括过去的经验，个人的技能，知识构架，思维方式，和对待测软件系统的了解程度等。这些因素都会或多或少地影响测试人员的执行测试的质量。

本文构建基于探索型测试的模型，首先定义效用值V，然后把影响因素数量化之后映射到V，用V来描述一个**检查点**出现故障的可能性，这个值是测试人员从自身的经验技能及对系统的理解程度出发做出的判断，是一个主观上的效用值，用来决策一个检查点是否应当经行测试。即： $V=F(E,P,K)$ ；

V: Value of testing necessary; E: experience ; P: personnel situation ; K: knowledge of the SUT

(注：检查点：具体确定待测故障后，依据故障检查规则自动或手动找出的可能出现此类故障的地方。)

模型中E表示过去经验，P表示个人的一些情况，涵盖了个人的思维技能知识等方面因素，K表示他对系统的了解程度。在一个测试人员即将开始一次探索性测试时，他的经验和个人情况是相对既定的，之所以把两者独立起来，因为在诸多因素中，经验所占的比重是最大的。但是，对系统的理解程度，是随着学习和测试的不断深入而变化的，这样使得每个检查点的

效用值在测试过程中不断变化。各个因素具体的数量化方法，随着所采用的函数F而不同；而F的选择应当在面向具体故障时给予确定。

由于采用遗传算法进行测试，需要给出编码规则和适应度函数。(1)编码是先进进行代码解析，根据给定规则，找出可能出错的检查点，各位取值0或1，0表示出错可能性较小不许测试，1则相反，是最后的测试点。由于对软件质量要求不同，所给出的规则可能有不同的严格程度。把所有的检查点放在一起，就得到了一个n位的串，作为染色体串。我们的目标就是对这个串优化，寻找出有限测试成本内的检查点组合。(2)适应度函数： $T = \sum_{i=1}^n V_i \times t_i$ ， $t_i$

是编码串中的位值，取值为0或1； $V_i$ 是各位的测试必要值。

在模型中，采用遗传算法行探索性测试。在寻找最佳组合时，设 I是给定测每轮计划测试次数，即测试成本，我们的目标是

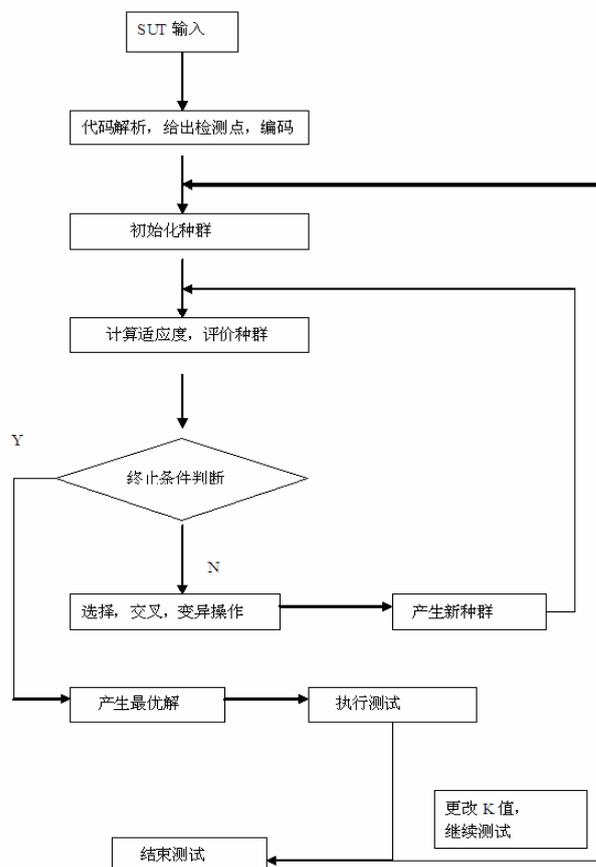
Max  $T = \sum_{i=1}^n V_i \times t_i$

$$\text{Max } T = \sum_{i=1}^n V_i \times t_i$$

$$\text{s. t. } \sum_{i=1}^n t_i \leq I$$

这样该问题就退化为0-1背包问题，可以有多种方法给出近似或确定解，但是为了更好的模拟人类思维进行探索型测试，遗传算法的定向性和随即性是最佳选择。

### 5测试流程：



## 6实验及结果:

实验采用了 PC—lint 代码检测工具，针对非法计算故障，对一个用 C++编写的大约 2 万行代码的程序做了两组测试实验。当 PC—lint 给出警报点时，一般测试方法要对所有警报点进行测试，存在测试成本太高的情况。根据本文所给出的基于探索性测试的方法，可以在保证测试质量的基础上，减少测试成本。下图是实验数据统计

试验一

测试工具	警报点	检查点	故障点	测试成本
PC-lint	27	27	9	27
实验算法	27	10	8	10

进行探索性试验时，测试人员的经验知识背景等方面会影响测试的效果，下面实验结果反应了两个经验不同的试验者测试结果的差异。

试验二

试验者	警报点	检查点	故障点	测试成本
试验者一	25	10	5	10
试验者二	25	10	8	10

PC—lint 工具给出的警报点存在误报和漏报的情况，但是一般已能涵盖大部分故障点，所以上述试验不考虑漏报的情况。

**7 结束语:** 探索性测试要求充分发挥测试人员的个人能力，把自身的经验技能及对系统的学习认知作为参考，灵活地经行测试设计和执行。本文模型把这些方面量化并以此构造适应度函数，理论上能够寻找低代价高覆盖的测试组合。但是在实践中，首先要解决的是检查点生成的任务，这需要一定的经验和理论指导，也是一个很有意义的研究点。另外，遗传算法有一定的缺陷，比如收敛过早等，可以把改进的算法应用到该模型中。

**本文作者创新点:** 采用量化处理的方法对探索性测试进行研究，并结合遗传算法对故障测试进行优化，为软件测试提供了一种新的思路和方法。

参考文献:

- [1] 练荣政. 一个基于探索性软件测试理论的测试用例生成系统的研究与实现 北京交通大学硕士学位论文 2008. 6
- [2] Jörg Denzinger. Exploratory testing for unwanted behavior using evolutionary learning techniques [M]. Technical Report 2007-868-20, 2007, June 8
- [3] 叶焰锋, 叶俊民. 数组越界的故障模型及其检测方法研究 [J]. 微计算机信息, 2007, 11-1: 145-147.
- [4] 宫云战. 一种面向故障的软件测试新方法 [J]. 装甲兵工程学院学报, 2004. 3:21-25
- [5] Cem Kaner, James Bach. The Nature of Exploratory Testing. <http://www.testingeducation.org>. 2004.

**作者简介:** 谢经纬(1983-), 男(汉族), 河北籍, 湖南大学硕士研究生, 研究方向为软件测试。吴昊(1968-) 男(汉族), 湖南籍, 湖南大学硕士生导师, 博士, 研究方向为软件理论。

**Biography:** XIE Jing-wei ( 1983- ), male ( han ), He Bei Province, Master( Hu Nan University ), major in software testing.

(410082 湖南长沙 湖南大学计算机与通信学院)谢经纬 吴昊  
通讯地址: 410082 湖南长沙 湖南大学南校区14舍424室 谢经纬