

收件记录

日期：
收件人：

中国科学院软件研究所 计算机科学重点实验室开放课题基金 申 请 书

申请人姓名：孙昌爱

所在单位：北京科技大学

申请课题：面向并发程序的蜕变测试技术研究

申请金额：12 万元人民币

年限起止：2019 年 1 月 ~ 2020 年 12 月

填报日期：2018 年 3 月 30 日

中国科学院软件研究所
二零零四年五月制

填 报 说 明

1. 填写申请书前，请先阅读“开放课题基金管理办法”和“开放课题基金申请指南”。表格的各项内容，要实事求是，逐条认真填写。表达要明确、严谨，字迹要清晰。外来语要同时用中文和原文表达。第一次出现的缩写词，须注出全称。
2. 请用A4纸以正反面打印，并于左侧装订成册。各栏空格不够时，请自行加页。
3. 申请书一式三份，由所在单位学术委员会签署意见后，投送计算机科学重点实验室。
4. 中级科研人员申请开放课题，需有高级职称的科研人员作为推荐人。
5. 通讯地址：北京市海淀区中关村南四街四号，中国科学院软件研究所，计算机科学国家重点实验室，邮政编码：100190。来函请注明“开放课题”。
6. 办公电话：010-62661616
传真号码：010-62661627
Email地址：zli@ios.ac.cn。

申请人及课题组情况

申请人	姓名	孙昌爱		性别	男		出生年月	1974年7月	
	民族	汉		职称	教授		身份证号码	32092319740718093x	
	通信地址、 邮编	北京科技大学 计算机与通信工程学院 100083							
	联系电话	13641386936			电子邮箱	casun@ustb.edu.cn			
项目组	总人数	高级	中级	初级	博士后	博士生	硕士生	其他	
	9	2	0	0	0	2	5	0	
成员姓名	职称/学位	性别	年龄	身份证号码		工 作 单 位		签名	
陈宗岳	教授/博士	男	70	H0967432501		澳大利亚斯文本大学			
付安	博士生	男	25	13010519930925 0938		北京科技大学			
王真	博士生	女	24	13112219940318 2449		北京科技大学			
代贺鹏	硕士生	男	25	41282419930301 683X		北京科技大学			
李盟	硕士生	男	24	41282819940427 0170		北京科技大学			
朱维忠	硕士生	男	24	45033119940926 1237		北京科技大学			
唐锦	硕士生	男	23	34082319951105 6119		北京科技大学			
李好纤	硕士生	女	23	36010219951111 5821		北京科技大学			

一、申请者正在承担的其它研究项目情况

项目名称	起始日期	负责或参加	来 源	进展或完成情况
模型驱动的 SOA 软件测试与监控技术研究	2016.1~ 2018.12	负责	北京市自然科学基金	在研。本课题已经资助发表国际期刊论文 3 篇，国际会议论文 2 篇，国内一级学报论文 2 篇；在审国际期刊论文 3 篇、在审国际会议论文 2 篇；授权国家发明专利 2 项；计算机软件著作权 3 项；培养研究生 6 名；基本完成课题预期指标。
面向航空嵌入式软件的变异测试框架与优化技术研究	2016.10~ 2018.9	负责	中国航空科学基金	在研。本课题已经资助发表国际期刊论文 1 篇、国际会议论文 1 篇；培养研究生 3 名；在审国际期刊论文 3 篇、在审国际会议论文 2 篇；授权国家发明专利 1 项，申请国家发明专利 1 项；计算机软件著作权 3 项；全国软件原型竞赛优秀奖 1 项；基本完成课题预期指标。

二、申请人主要学术经历（包括大学以上学历，工作经历及论文、成果等）

孙昌爱，博士，教授，博士生导师，现为北京科技大学计算机与通信工程学院院长助理、软件与安全研究所所长。2002年12月毕业于北京航空航天大学计算机学院（硕博连读），获计算机软件与理论博士学位；1997年7月年毕业于北京科技大学计算机科学与技术系，获计算机及应用学士学位。2003.3-2003.7，香港理工大学助理研究员；2004.3-2006.5，澳大利亚斯文本大学任博士后；2006.5-2007.5，荷兰格罗宁根大学任博士后；2007.9-2009.4，北京交通大学计算机与信息技术学院讲师；2009.5-2014.6，北京科技大学计算机与通信工程学院副教授；2014.7-现在，北京科技大学计算机与通信工程学院教授。其间，2013.3-2014.2，美国普渡大学计算机系访问学者；2017.1-2017.2，澳大利亚斯文本大学访问教授；曾担任北京科技大学计算机系副主任、软件工程系主任。

长期从事软件测试、程序分析、服务计算等方向的研究与教学，主持完成国家自然科学基金、北京市自然科学基金、教育部博士点新教师基金、教育部留学回国人员科研启动基金等十余项研究课题，参与完成了欧盟研究理事会资助的第六框架计划项目、澳大利亚自然科学基金项目等海内外研究课题近二十项。在《ACM Transactions on the Web》、《IEEE Transactions on Services Computing》、《Journal of Systems and Software》、《Information and Software Technology》、《Software: Practice and Experience》、《Software Quality Journal》、《The Computer Journal》、《计算机学报》、《软件学报》等国内外重要学术刊物和国际会议上发表论文80余篇，SCI/EI收录70余篇次、他引1700余次；申请国家发明专利6项（已授权4项）、登记计算机软件著作权18项、出版译著3部；完成北京科技大学校级教育教学改革项目3项、获北京科技大学教育教学成果奖一等奖1项、二等奖2项；获得北京市高等教育教学成果奖二等奖1项；培养研究生30余名，其中6人获“北京科技大学优秀硕士论文”、4人次获计算机与通信工程学院“学术之星”称号、8人获得研究生国家奖学金、3人获得“北京市优秀毕业生”称号；获得中国计算机学会主办的软件原型竞赛二等奖1项、三等奖1项、优秀奖1项，中国计算机学会主办的服务创新大赛一等奖1项、二等奖1项；曾在美国、荷兰、澳大利亚、德国、香港多所著名大学从事合作研究或学术访问，入选北京市优秀人才培养资助计划（2012）。

现为国际电气和电子工程师协会高级会员（IEEE Senior Member）、中国计算机学会高级会员、中国电子学会高级会员、中国计算机学会服务计算专业委员会常务委员、中国计算机学会软件工程专业委员会委员、中国电子学会青年科学家俱乐部成员、边缘计算产业联盟咨询委员会专家。担任12th IEEE International Conference on Ubiquitous Intelligence and Computing (UIC 2015)、Asia-Pacific Services Computing Conference (APSCC 2015) (Special Track)、2008 International Symposium on Trusted Computing (TrustCom 2008) 等国际会议程序委员会主席，担任8th IEEE International Conference on Service Oriented Computing and Applications (SOCA 2015) 国际会议研讨会主席，担任“2014 中荷双边国际研讨会”、“2015 年服务软件的测试与分析研讨会”大会主席，担任国际研讨会 ICSE International Workshop on Metamorphic Testing (ICSE MET 2017) 组委会委员，担任软件工程与服务计算领域国际、国内会议学术程序委员会委员80余次。

近期与本项目相关的主要论著:

- [1]C. Sun, Y. Zhao, L. Pan, H. Liu, T.Y. Chen. Automated Testing of WS-BPEL Service Compositions: A Scenario-Oriented Approach, *IEEE Transactions on Services Computing* (DOI: 10.1109/TSC.2015.2466572), in press.
- [2]孙昌爱, 郭新玲, 张翔宇, 陈宗岳. 一种基于数据流分析的冗余变异体识别技术, 计算机学报, 刊出中 (录用日期: 2018年1月24日)
- [3]C. Sun, Y. Ran, C. Zhen, H. Liu, D. Towey, X. Zhang. Fault localization for WS-BPEL program based on Predicate Switching and Program Slicing, *Journal of Systems and Software*, Elsevier, 2018, 135:191-204.
- [4]C. Sun, L. Pan, Q. Wang, H. Liu, X. Zhang. An Empirical Study on Mutation Testing of WS-BPEL Programs, *The Computer Journal*, Oxford University Press, 2017, 60(1):143-158.
- [5]C. Sun, C. Fan, Z. Wang, H. Liu. duReg: A Path-Aware Mutation Analysis Guided Approach to Regression Testing, *Proceedings of the 12th IEEE/ACM International Workshop on Automation of Software Testing* (AST 2017), collocated with ICSE 2017, May 20-28, 2017, pp.59-64.
- [6]C. Sun, F. Xue, H. Liu, X. Zhang. A Path-aware Approach to Mutant Reduction in Mutation Testing, *Information and Software Technology*, Elsevier, January 2017, 81(1):65-81.
- [7]D. Towey, Y. Dong, C. Sun, T.Y. Chen, Metamorphic Testing as a Test Case Selection Strategy, *Science China: Information Sciences*, 2016, 59(5):050108:1-050108:2.
- [8]C. Sun, Y. Zhao, L. Pan, X. He, D. Towey. A Transformation-based Approach to Testing Concurrent Programs using UML Activity Diagrams, *Software: Practice and Experience*, Wiley, 2016, 46(4):551-576.
- [9]C. Sun, G. Wang, Q. Wen, D. Towey, T.Y. Chen. MT4WS: An Automated Metamorphic Testing System for Web Services. *International Journal of High Performance Computing and Networking*, Inderscience Press, 2016, 9(1-2):104-115.
- [10]C. Sun, Y. Liu, Z. Wang, W.K. Chan. μ MT: A Data Mutation Directed Metamorphic Relation Acquisition Methodology, *Proceedings of the First International Workshop on Metamorphic Testing* (MET 2016), collocated with ICSE 2016, IEEE Computer Society, May 14-22, 2016. pp.12-18.
- [11]C. Sun, Y. Zhai, H. Liu. Evaluating and Comparing Fault-based Testing Strategies for General Boolean Specifications: A Series of Experiments. *The Computer Journal*, Oxford University Press, 2015, 58 (5): 1199-1213.
- [12]C. Sun, Z. Wang, G. Wang. A Property-based Testing Framework for Encryption Programs. *Frontiers of Computer Science*, Springer, 2014, 8(3):478-489.
- [13]孙昌爱, 王冠. MujavaX:一个支持非均匀分布的变异生成系统, 计算机研究与发展, 2014, 51(4): 874-881.
- [14]C. Sun, Y. Zhai, Y. Shang, Z. Zhang. BPELDebugger: An effective BPEL-specific fault localization framework. *Information and Software Technology*, Elsevier, 2013, 55(12): 2140-2153.
- [15]C. Sun, G. Wang, B. Mu, H. Liu, Z. Wang, T.Y. Chen. A Metamorphic Relation-Based Approach to Testing Web Services without Oracles, *International Journal on Web Service Research*, 2012, 9(1): 51-73.

申请课题介绍

一、研究的意义、预期成果、应用前景及国内外发展综述

1. 研究的意义及国内外发展综述

近年来，多核计算日益普及，并发程序广泛用来提升此类计算系统的资源利用效率。并发程序中存在多个并发执行的流程，流程之间通常显式或隐式地共享一些存储空间，并且它们执行的次序不确定[1]。并发流程之间不确定的相互作用与影响的情形通常简称为执行交错(Interleaving)。执行交错不仅易于导致并发程序在设计阶段引入新的缺陷，而且给并发程序的分析与测试带来了挑战。如何有效地检测并发程序中潜藏的缺陷、提高并发程序的可靠性已经成为并发程序开发领域的一个亟待解决的重要问题。

软件测试是一种广泛采用的软件质量保证手段[2]，通过运行有限的测试用例，比较测试用例的输出与预期输出是否一致来检测软件中潜藏的故障。与传统的顺序执行程序相比，并发程序的测试工作更具挑战性。由于流程之间执行的次序不确定，每一种流程执行序列对应一个交错模式。在某些交错模式下指令的执行顺序和开发人员的预期不同或者并发程序输出的结果与预期不同，这样的交错模式也称为交错性质。并发程序中常见的交错性质包括：（1）**数据竞争**：来自不同线程的两个操作（至少有一个是写操作）并发地访问没有同步机制保护的数据项；（2）**原子违反**：某一流程中应具有原子性的一系列操作与来自其它流程的操作出现交错；（3）**死锁**：同步机制无限期地阻止一些流程的继续执行。由于上述交错性质，并发程序测试在测试用例生成、交错性质分析、测试结果判定等方面面临新的问题与挑战。

近年来，围绕并发程序的测试用例生成、执行交错分析、测试结果判定问题，研究者提出了一系列的并发程序测试技术[1]。部分代表性研究工作包括：

- （1）**并发程序的测试用例生成方面**：测试用例生成是软件测试的一个重要步骤，直接影响软件测试的故障检测能力。并发程序的测试用例生成技术主要有：（1）**随机生成方法** [3]：适用于规模较小的并发程序，能够揭示一些简单的并发程序故障类型；（2）**顺序测试用例并发化处理方法**：从已有的测试用例集中选择与并发操作相关的测试用例，然后将这些测试用例构建成并发测试用例。
- （2）**并发程序执行交错检测方面**，已有技术可以分为**基于故障的交错执行检测技术**和**交错空间探索技术**两大类。其中，**基于故障的交错执行检测技术**主要关注待测并发程序的执行交错是否存在交错性质，即通过分析执行轨迹识别操作间的同步约束关系，然后判断当前执行交错是否符合交错性质的条件；**交错空间探索技术**探索待测并发程序中可能出现的所有交错模式组成的交错空间。

依据不同的交错性质类型，**基于故障的交错执行检测技术**可以分为：

- **面向数据竞争的交错执行检测技术**：首先分析给定执行流程中访问共享内存变量的执行次序，然后判断是否发生了数据竞争或者预测可能出现数据竞争的交错。并发程序中流程执行某个代码片段或者执行某个指令称为事件，Lamport 等人提出

针对并发程序中多个流程中不同事件排序的 happens-before 分析方法[4]，该方法首先对并发程序中的事件进行排序，然后分析排序后的事件是否可能发生数据竞争。由于需要分析每个执行流程的信息、以及所有的执行流对每个共享数据项的访问信息，该方法存在时间开销大的缺点。为了改进 happens-before 方法的性能，Flanagan 等人提出了在运行时保存额外信息的动态 happens-before 方法[5]，该方法仅记录共享数据最近一次写操作，因此减少了分析的对象，有效地降低了时间开销。Marino 等人提出对部分源程序进行采样分析的轻量级探测技术[6]，仅仅分析部分程序而不是整个程序，从而降低了时间开销。基于共享变量经常在没有持锁的情况下被初始化这一观察，Savage 等人提出基于锁集分析的数据竞争检测方法[7]，首先为每一个待测并发程序的共享变量 v 创建一个锁集合 $C(v)$ ，该集合初始时包含所有可能的锁对象；当流程访问 v 时，将 $C(v)$ 与当前流程持有锁形成的集合 $T(l)$ 的交集作为新的 $C(v)$ 。如果 $C(v)$ 为空集，则表明共享变量 v 没有被锁保护，流程之间可能出现数据竞争。Shacham 等人将动态锁集分析与模型检测结合，提出一种共享内存程序的数据竞争检测方法[8]，首先通过动态锁集分析捕获锁约束信息，然后利用模型检测产生可选择的执行交错，最后执行识别的交错来揭示故障。Sheng 等人开发了基于锁集分析的并发程序的数据竞争检测工具[9]，可以选择性捕获并发程序中同步操作和内存访问，有助于降低锁集分析技术的时间开销。

- **面向原子违反的执行交错检测技术：**利用执行交错串行化验证程序的正确性。根据具有原子性的区域不同，相关技术可以分为以代码为中心与以数据为中心的原子违反检测技术。**(1) 以代码为中心的原子违反检测技术：**根据软件规格说明书确定具有原子性的代码区域，然后在识别的代码区中检测是否可能出现违反原子性的执行交错。一个方法（或一个代码块）具有原子性当且仅当这个方法在执行过程中不被其它流程的事件交错。Flanagan 等人基于上述理论提出了 Atomizer 方法[[10]，首先通过执行待测程序得到并发程序的事件序列，然后根据定义的规则改变事件发生的次序，如果最终某个方法（或代码块）的所有事件没有被其它流程的事件交错，则不存在原子违反；否则，检测到原子违反。在 Atomizer 方法基础上，研究者提出了一系列的原子违反检测改进方法：Lu 等人提出一种根据待测程序的结构以及执行信息推断原子代码块方法[11]，有效缓解了无法从软件规格说明书中获得原子代码块的问题；Flanagan 等人依据原子代码块的规格说明分析可能引起原子违反的操作，忽略不相关的操作[12]，有助于减少故障误报情形与降低时间开销；**(2) 以数据为中心的原子违反检测技术：**该方法首先定义常见的不符合原子集访问的交错场景模式，然后检查在并发执行流程是否存在上述模式。如果存在，则表明存在原子违反故障。原子集表示程序中原子化更新的变量，而定义在原子集上的操作称为工作单元[13]。在原子集与工作单元等概念基础上，Vaziri 等人定义了 11 种原子违反故障类型[14]。类似的，Hammer 等人识别了 11 种可能导致原子违反的数据访问交错场景，提出了相应的自动检测算法[15]。Lai 等人利用 happens-before 分析方法推断 Hammer 等人识别的可能导

致原子违反的数据访问交错场景 [16]。

- **面向死锁的执行交错检测技术：**死锁的主要原因是共享资源访问次序不正确。例如，两个线程共享两个锁对象并且这两个线程访问这两个锁的顺序相反。Havelund 基于上述观察提出一种运行时分析算法 GoodLock [17]。该算法在程序运行时记录每一个流程访问不同锁对象的次序并构建锁树，程序终止后，成对比较锁树并判断对应的执行流之间是否可能出现死锁。由于锁树忽略了程序的细节，只记录每一个流程访问锁对象的次序，因此可能存在故障误报的情形；在 GoodLock 方法基础上，Joshi 等人提出了一种面向并发程序的两步骤死锁动态分析方法[18]，首先构建锁树并分析程序中可能潜在的死锁，然后执行预测造成死锁的交错，该方法可以有效降低 GoodLock 的误报率；为了提高 GoodLock 方法的性能，Samak 等人提出一种面向锁树的修剪方法[19]、利用顺序执行的程序属性构造死锁检测的并发程序用例生成方法[20]。

依据探索交错空间的不同方式，**交错空间探索技术**可以分为：

- **面向执行交错的压力测试：**传统的压力测试通过逐渐增加负载观察与确定目标系统的性能瓶颈。在并发程序中，压力测试反复执行同一个测试用例集，旨在检查并发程序是否可线性化或版本变化是否对并发程序的性能产生影响。线性一致性是并行对象行为正确性的一个条件模型，指并发程序执行的历史中存在可线性化点 P 的执行模型，即对于某个操作在程序的调用和返回之间的某个点 P，系统中并发运行的所有其他线程都能感知（保证某个线程中操作的调用与返回满足 FIFO 条件）。Fonseca 等人提出一种基于线性一致性的并发故障检测方法[21]，首先针对某个测试用例的每一次并发执行结果与所有可能的线性化执行结果进行比较，如果存在某次并发执行结果与任意线性化执行都不相同，则存在语义并发故障；然后捕获每次并发执行的程序状态，通过比较不同测试用例下程序状态的差异检测潜在的并发故障。Pradel 等人提出了一种不同版本并发程序的性能比较技术 [22]。
- **并发程序执行交错穷尽探索：**旨在探索待测并发程序中所有的交错模式。随着程序规模扩大和并发流程数量的增加，交错模式的数量呈指数式增长，探索所有可能的交错模式需要大量的资源。一方面，合适的测试用例集可以减小交错空间。Nistor 等人提出了面向对象并发程序的随机测试方法 Ballerina [23]，首先针对待测对象中某个方法，随机创建两个并发访问的线程，然后基于线性一致性检测并发执行中是否存语义并发故障。Pradel 等人将并发程序在非线性化执行中出现的崩溃和死锁作为并发故障触发的充分条件[24]，有助于降低 Ballerina 方法的故障误报数。另一方面，通过符号执行或者模型检测探索交错空间。Sen 等人提出一种增量的执行交错探索方法[25]，首先利用符号执行产生测试用例，执行测试用例并记录覆盖的分支节点，然后继续产生测试用例覆盖尚未覆盖的分支。
- **并发程序执行交错覆盖准则：**利用待测并发程序的内部信息识别交错空间的一个子集。覆盖标准利用共享内存中的数据流关系、同步代码块的访问次序限制要探索的程序代码量。Wang 等人提出一种高风险执行交错覆盖准则，首先动态记录共享变

量的访问次序，然后选择执行高风险的交错模式（具有较大概率揭示并发故障的执行交错）[26]。Hong 等人提出了同步代码区的覆盖准则，在程序执行过程中构建线程模型，动态捕获并记录线程访问共享内存以及同步代码块的操作，然后执行尚未被覆盖的同步代码块[27]。

- (3) **并发程序的测试结果判定：**并发程序的测试结果判定关注两方面的内容[1]：(1) 待测程序在执行过程中是否出现符合某种故障类型（数据竞争、原子违反、死锁等）的执行交错；(2) 待测并发程序执行结果是否与预期结果相同。已有的面向并发程序的测试技术侧重执行过程的交错性质的检测问题（1）；当测试预期不存在时，已有并发测试技术则无法适用（2）。**如何解决并发程序的测试预期问题方面的研究工作鲜有报道。**

由于并发程序执行的不确定性，并发程序的测试预期问题不仅存在，而且更加突出。针对测试预期问题，已有的技术包括 N-version 测试[28]、断言[29]、机器学习[30]、假设检验[31]、蜕变测试[32]。其中，蜕变测试依据待测软件的蜕变属性获取蜕变关系，执行原始测试用例（采用测试用例生成技术获得）与衍生测试用例（根据蜕变关系获得），检查对应的输出是否违反蜕变关系。如果违反了某种蜕变关系，则表明存在故障。不难看出，**蜕变测试无需测试预期，因此有效地缓解了测试预期问题，并在多个领域得到成功的应用[33]**。近年来，研究者在蜕变测试理论与应用等方面取得了丰富的研究成果。部分代表性工作如下：

- (1) **蜕变关系识别与复合方面：**Chen 等人提出了一种基于范畴划分的蜕变关系识别方法[34]；Zhang 等提出一种基于搜缩的蜕变关系推理方法[35]；Su 等人提出了一种蜕变关系动态识别方法[36]；Sun 等人提出一种数据变异指导的蜕变关系获取方法[37]；Gotlieb 等人提出了一种蜕变关系验证框架[38]，对于给定的程序的某个蜕变关系植入故障，利用约束逻辑编程技术生成满足植错后的蜕变关系的测试用例集，如果测试用例集检测出故障，则表明植错后的蜕变关系是错误的；Liu 等人提出一种蜕变关系复合方法，通过对多个蜕变关系的复合形成新的蜕变关系[39]。
- (2) **蜕变测试原始测试用例生成方面：**Batra 等人提出了一种基于遗传算法的测试用例生成方法，旨在最大化覆盖程序路径[40]；Dong 等人采用符号执行提取蜕变关系并生成相应的测试用例[41]；Chen 等人比较了特殊值法与随机方法产生的原始测试用例对蜕变测试效率的影响[42]。
- (3) **蜕变测试与其它测试技术结合：**Xie 等人将蜕变测试与基于频谱分析的故障定位技术结合，提出了无需预期的故障定位方法[43]；Dong 等人将蜕变测试与遗传算法相结合，在搜索过程中使用蜕变关系，将原始测试用例和衍生测试用例都视为候选方案，加速达到某种覆盖标准[44]。
- (4) **蜕变测试在不同领域中的应用：**Sun 等人提出了面向 Web 服务的蜕变测试方法[45-46]，首先从 Web 服务的 WSDL 描述文档中提取蜕变关系，然后根据 WSDL 文档随机产生测试用例并且运用蜕变关系得到相应的衍生测试用例；Chan 等人提出了一种面向 SOA 的蜕变测试方法，将离线测试通过的测试用例作为原始测试用例，在线测试用例作为衍生测试用例[47-48]；Mayer 等人将蜕变测试应用于图像处理程序[49]；Kuo 等人

将蜕变测试应用于图象处理程序时检测出一个真实故障[50]；Tse 等人尝试将蜕变测试应用于上下文敏感的中间件软件[51]；Chan 等人将蜕变测试应用于能量感知的无线传感器网络应用软件[52]；Sun 等人将蜕变测试成功地用于几类典型加密程序的测试[53]。

课题组在国家自然科学基金面上项目“面向 SOA 软件的蜕变测试技术研究”、中科院软件所计算机科学重点实验室开放课题“面向 Web 服务及其组合的蜕变测试技术与框架研究”等资助下，系统地研究了蜕变测试应用于 SOA 软件的关键问题，取得如下主要成果：1) 建立了面向 SOA 的通用蜕变测试框架[45]；2) 提出了一种基于数据变异的蜕变关系获取方法[37]；3) 提出了一种改进的 METRIC 方法 METRIC*，在基于范畴划分的蜕变关系识别框架 METRIC 中引入软件的输出行为；4) 设计了一种蜕变关系通用描述语言 MRDL，开发了基于 MRDL 的衍生测试用例生成算法和测试结果判定算法[46]；5) 设计与实现迭代蜕变测试算法，改进蜕变测试中原始测试用例生成效率[54]；6) 通过将符号执行与约束求解相结合，提出一种基于符号执行的原始测试用例生成技术；7) 依据测试用例覆盖的程序路径，提出一种基于路径距离的原始测试用例优先级排序方法；8) 提出基于马尔可夫链和基于奖惩机制的两种适应性分区控制策略，将反馈控制引入到蜕变测试过程，改进蜕变测试的故障检测效率[55]；9) 研制出一个面向 Web 服务的蜕变测试支持系统 MT4WS [56] [57]。除了上述蜕变测试方面的研究工作，课题组还在本课题相关的并发测试和变异测试方面积累了丰富的经验。在并发测试方面：1) 提出一个基于变换的并发程序测试用例生成方法，使用变换技术解决了活动图中存在的非结构化问题，根据不同的并发覆盖准则按需生成不同的测试用例集，生成的测试用例集的大小是可控的，开发了相关支持工具[58]；2) 针对服务组装 BPEL 程序的工作流特性、存在并发、多分支等特点，提出一种基于场景的 BPEL 测试用例自动生成技术[59]。在变异测试方面：1) 提出了一种 BPEL 程序的变异测试框架[60]，开发相应的支持工具[61]；研究了如何降低面向 BPEL 程序的变异测试的开销问题，从变异算子之间的包含关系、变异算子优先级、二阶变异三个方面，提出一组面向 BPEL 程序的变异测试的优化策略[62]；2) 从不同角度改进变异测试技术，包括：提出一种分布感知的变异测试模型[63]；利用程序的结构信息，提出一种路径感知的变异体精简技术[64]；从减少冗余变异体角度，提出一种基于数据流分析的冗余变异体识别技术[65]；从加快变异测试速度角度，提出了一种轻量级的并发变异测试方法。

本课题将在蜕变测试、并发程序测试、变异测试等上述研究工作的基础上，开展面向并发程序的蜕变测试技术研究。预期研究成果提供一种新型的并发程序软件测试理论，为开发可靠的并发程序提供测试理论支持。

2. 预期成果

- (1) 面向并发程序的蜕变测试框架、方法与支持工具方面，发表高水平学术论文 3-4 篇，其中 SCI 检索国际期刊论文至少 1 篇。
- (2) 面向并发程序的蜕变测试技术演示系统：采用多个并发程序实例演示与验证提出的蜕变测试框架与方法，提交技术报告 1 份。
- (3) 登记计算机软件著作权 1~2 项。

(4) 培养研究生 5~7 人。

3. 应用前景

随着大数据、云计算、高性能计算等普及与发展，人们对软件性能要求愈来愈高，并发程序设计技术将受到更多的关注。一方面，并发程序设计有助于解决计算机系统资源充分利用、提升计算速度；另一方面，并发程序的行为具有不确定性，测试人员往往不容易得到测试预期或者获得测试预期的代价很高。与传统程序测试相比，并发程序的测试更加困难。本课题旨在探索如何在预期不存在的情形下仍然对并发程序进行有效的测试，如何解决蜕变测试应用于并发程序是关键问题。本课题预期成果可为开发可靠的并发程序提供有效的测试方法与工具支持，对于高性能计算、云计算、操作系统等多个领域的并发程序的质量保证有重要的理论意义与实践价值。

参考文献

- [1] F. Bianchi, A. Margara, M. Pezze. A Survey of Recent Trends in Testing Concurrent Software Systems. *IEEE Transactions on Software Engineering*, IEEE Computer Society, 2017, 1(2):1-2.
- [2] G. J. Myers, C. Sandler, T. Badgett. The Art of Software Testing. John Wiley and Sons, 2011.
- [3] R. Hamlet. Random Testing. *Encyclopedia of Software Engineering*. John Wiley and Sons, Inc., 2002.
- [4] L. Lamport. Time, Clocks, and the Ordering of Events in a Distributed System. *Communications of the ACM*, ACM Press, 1978, 21(7): 558-565.
- [5] C. Flanagan, S. N. Freund. FastTrack: Efficient and Precise Dynamic Race Detection. *Proceedings of the 30th ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'09)*. ACM Press, 2009, pp. 121-133.
- [6] D. Marino, M. Musuvathi, S. Narayanasamy. LiteRace: Effective Sampling for Lightweight Data-Race Detection. *Proceedings of the 30th ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'09)*. ACM Press, 2009, pp. 134-143.
- [7] S. Savage, M. Burrows, G. Nelson. Eraser: A Dynamic Data Race Detector for Multithreaded Programs. *ACM Transactions on Computer Systems*, ACM Press, 1997, 15(4): 391-411.
- [8] O. Shacham, M. Sagiv, A. Schuster. Scaling Model Checking of Data-races Using Dynamic Information. *Journal of Parallel and Distributed Computing*, Elsevier, 2007, 67(5): 536-550.
- [9] T. Sheng, N. Vachharajani, S. Eranian. RACEZ: A Lightweight and Non-Invasive Race Detection Tool for Production Applications. *Proceedings of the 33rd International Conference on Software Engineering (ICSE'11)*. ACM Press, 2011, pp. 401-410.
- [10] C. Flanagan, S. N. Freund. Atomizer: A Dynamic Atomicity Checker for Multithreaded Programs. *Proceedings of the 31st Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL'04)*. ACM Press, 2004, pp. 256-267.
- [11] S. Lu, J. Tucek, F. Qin. AVIO: Detecting Atomicity Violations via Access Interleaving Invariants. *Proceedings of the 12th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS'06)*. ACM Press, 2006, pp. 37-48.
- [12] C. Flanagan, S. N. Freund, J. Yi. Velodrome: a sound and complete dynamic atomicity checker

- for multithreaded programs. *Proceedings of ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'08)*, ACM Press, 2006, pp. 334-345.
- [13]L. Wang, S. D. Stoller. Runtime Analysis of Atomicity for Multithreaded Programs. *IEEE Transactions on Software Engineering*, 2006, 32(2): 93-110.
- [14]M. Vaziri, F. Tip, J. Dolby. Associating Synchronization Constraints with Data in an Object-Oriented Language. *Proceedings of the 33rd Annual ACM SIGPLAN – SIGACT Symposium on Principles of Programming Languages (POPL'06)*. ACM Press, 2006, pp. 334-345.
- [15]C. Hammer, J. Dolby, M. Vaziri. Dynamic Detection of Atomic-Set-Serializability Violations. *Proceedings of the 30th International Conference on Software Engineering (ICSE'08)*, IEEE Computer Society, 2008, pp. 231-240.
- [16]Z. Lai, S. C. Cheung, W. K. Chan. Detecting Atomic-Set Serializability Violations in Multithreaded Programs through Active Randomized Testing. *Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering (ICSE'10)*. IEEE Computer Society, 2010, pp. 235-244.
- [17]K. Havelund. Using Runtime Analysis to Guide Model Checking of Java Programs. *Proceedings of the 7th International SPIN Workshop on SPIN Model Checking and Software Verification (SPIN'00), Co-located with the 22nd International Conference on Software Engineering (ICSE'00)*, Springer, 2000, pp. 245-264.
- [18]P. Joshi, C. S. Park, K. Sen. A Randomized Dynamic Program Analysis Technique for Detecting Real Deadlocks. *Proceedings of the 30th ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'09)*, ACM Press, 2009, pp. 110-120.
- [19]M. Samak, M. K. Ramanathan. Trace Driven Dynamic Deadlock Detection and Reproduction. *Proceedings of the 19th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming (PPoPP'14)*, ACM Press, 2014, pp. 29-42.
- [20]M. Samak, M. K. Ramanathan. Multithreaded Test Synthesis for Deadlock Detection. *Proceedings of the 2014 ACM International Conference on Object Oriented Programming Systems Languages and Applications (OOPSLA'14)*, ACM Press, 2014, pp. 473-489.
- [21]P. Fonseca, C. Li, R. Rodrigues. Finding Complex Concurrency Bugs in Large Multi-Threaded Applications. *Proceedings of the 6th European Conference on Computer Systems (EuroSys'11)*, ACM Press, 2011, pp. 215-228.
- [22]M. Pradel, M. Huggler, T. R. Gross. Performance Regression Testing of Concurrent Classes. *Proceedings of the International Symposium on Software Testing and Analysis (ISSTA'14)*, ACM Press, 2014, pp. 13-25.
- [23]A. Nistor, Q. Luo, M. Pradel. Ballerina: Automatic Generation and Clustering of Efficient Random Unit Tests for Multithreaded Code. *Proceedings of the 34th International Conference on Software Engineering (ICSE'14)*, IEEE Computer Society, 2014, pp. 727-737.
- [24]M. Pradel, T. R. Gross. Fully Automatic and Precise Detection of Thread Safety Violations. *Proceedings of the 33rd ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'12)*, ACM Press, 2012, pp. 521-530.
- [25]K. Sen, G. Agha. Automated Systematic Testing of Open Distributed Programs. *Proceedings of the 9th International Conference on Fundamental Approaches to Software Engineering*

(FASE'06), Springer, 2006, pp. 339-356.

- [26]C. Wang, M. Said, A. Gupta. Coverage Guided Systematic Concurrency Testing. *Proceedings of the 33rd International Conference on Software Engineering (ICSE'11)*, ACM Press, 2011, pp. 221-230.
- [27]S. Hong, J. Ahn, S. Park. Testing Concurrent Programs to Achieve High Synchronization Coverage. *Proceedings of the ACM SIGSOFT International Symposium on Software Testing and Analysis (ISSTA'12)*, ACM Press, pp. 210-220.
- [28]S. S. Brilliant, J. C. Knight, P. E. Ammann. On the Performance of Software Testing Using Multiple Versions. *Proceedings of the 20th international symposium on fault-tolerant computing (FTCS'90)*, IEEE Computer Society, 1990, pp. 408–415.
- [29]K. Y. Sim, C. S. Low, F. C. Kuo. Eliminating Human Visual Judgment from Testing of Financial Charting Software. *Journal of Software*, 2014, 9(2): 298-312.
- [30]W. K. Chan, S. C. Cheung. PAT: A Pattern Classification Approach to Automatic Reference Oracles for the Testing of Mesh Simplification Programs. *Journal of Systems and Software*, 2009, 82(3), 422–434.
- [31]R. Guderlei, J. Mayer. Statistical Metamorphic Testing Testing Programs with Random Output by Means of Statistical Hypothesis Tests and Metamorphic Testing. *Proceedings of the 7th international conference on quality software (QSIC'07)*, IEEE Computer Society, 2007, pp. 404–409.
- [32] T. Y. Chen, S. C. Cheung, S. M. Yiu. Metamorphic Testing: A New Approach for Generating Next Test Cases. Technical Report HKUST-CS98-01, Department of Computer Science, Hong Kong University of Science and Technology, Hong Kong, 1998.
- [33]S. Segura, G. Fraser, A. B. Sanchez, A. R. Cortes. A survey on metamorphic testing. *IEEE Transactions on software engineering*, 2016, 42(9): 805-824.
- [34]T. Y. Chen, P. L. Poon, X. Xie. METRIC: Metamorphic Relation Identification Based on the Category-Choice Framework. *Journal of Systems and Software*, 2016, 116: 177-190.
- [35]J. Zhang, J. Chen, D. Hao. Search-Based Inference of Polynomial Metamorphic Relations. *Proceedings of the 29th ACM/IEEE International Conference on Automated Software Engineering (ASE'14)*, ACM Press, 2014, pp. 701-712.
- [36]F. H. Su, J. Bell, C. Murphy. Dynamic Inference of Likely Metamorphic Properties to Support Differential Testing. *Proceedings of the 10th International Workshop on Automation of Software Test (AST'15), Co-located with the 37th IEEE International Conference on Software Engineering (ICSE'15)*, IEEE Computer Society, 2015, pp. 55-59.
- [37]C. Sun, Y. Liu, Z. Wang, W.K. Chan. μ MT: A Data Mutation Directed Metamorphic Relation Acquisition Methodology, *Proceeding of the First International Workshop on Metamorphic Testing (MET 2016)*, collocated with ICSE 2016, IEEE Computer Society, 2016, pp.12-18.
- [38]A. Gotlieb, B. Botella. Automated Metamorphic Testing. *Proceedings of the 27th Annual International Conference on Computer Software and Applications (COMPSAC'03)*, IEEE Computer Society, 2003, pp. 34-40.
- [39]H. Liu, X. Liu, T. Y. Chen. A New Method for Constructing Metamorphic Relations. *Proceedings of the 12th International Conference on Quality Software (QSIC'12)*, IEEE

Computer Society, 2013, pp. 59-68.

- [40]G. Batra, J. Sengupta. An Efficient Metamorphic Testing Technique Using Genetic Algorithm. *Proceedings of 5th International Conference on Information Intelligence, Systems, Technology and Management (ICISTM'11)*, Springer, 2011, pp. 180-188.
- [41]G. Dong, T. Guo, P. Zhang. Security Assurance with Program Path Analysis And Metamorphic Testing. *Proceedings of the 4th International Conference on Software Engineering and Service Science (ICSESS'13)*, IEEE Computer Society, 2013, pp. 193-197.
- [42]T. Y. Chen, F. C. Kuo, Y. Liu. Metamorphic Testing and Testing with Special Values. *Proceedings of the 5th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD'04)*, 2004, pp. 128-134.
- [43]X. Xie, W. E. Wong, T. Y. Chen, B. Xu. Metamorphic Slice: An Application in Spectrum-Based Fault Localization. *Information and Software Technology*, 2013, 55(5): 866-879.
- [44]G. Dong, S. Wu, G. Wang, T. Guo, Y. Huang. Security Assurance with Metamorphic Testing and Genetic Algorithm. *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT'10)*, IEEE Computer Society, 2010, pp. 397-401.
- [45]C. Sun, G. Wang, B. Mu. Metamorphic Testing for Web Services: Framework and A Case Study. *Proceedings of the 9th IEEE International Conference on Web Services (ICWS'11)*, IEEE Computer Society, 2011, pp. 283-290.
- [46]C. Sun, G. Wang, B. Mu, H. Liu, Z. Wang, T. Y. Chen. A Metamorphic Relation-Based Approach to Testing Web Services without Oracles. *International Journal of Web Services Research*, 2012, 9(1): 51-73.
- [47]W. K. Chan, S. C. Cheung, K. R. Leung. Towards a Metamorphic Testing Methodology for Service-Oriented Software Applications. *Proceedings of 5th International Conference on Quality Software (QSIC'05)*, IEEE Computer Society, 2005, pp. 470-476.
- [48]W. K. Chan, S. C. Cheung, K. R. Leung. A Metamorphic Testing Approach for Online Testing of Service-Oriented Software Applications. *International Journal of Web Services Research*, 2007, 4(2): 61-72.
- [49]J. Mayer, R. Guderlei. On Random Testing of Image Processing Applications. *Proceedings of the 6th International Conference on Quality Software (QSIC'06)*, IEEE Computer Society, 2006, pp. 85-92.
- [50]F. C. Kuo, S. Liu, T. Y. Chen. Testing a Binary Space Partitioning Algorithm with Metamorphic Testing. *Proceedings of the ACM Symposium on Applied Computing (SAC'11)*, ACM Press, 2011, pp. 1482-1489.
- [51]T. H. Tse, S. S. Yau. Testing Context-Sensitive Middleware-Based Software Applications. *Proceedings of the 28th Annual International Computer Software and Applications Conference (COMPSAC'04)*. IEEE Computer Society, 2004, pp. 458-466.
- [52]W. K. Chan, T. Y. Chen, S. C. Cheung, T. H. Tse, Z. Zhang. Towards the Testing of Power-Aware Software Applications for Wireless Sensor Networks. *Proceedings of the 12th Ada-Europe International Conference on Reliable Software Technologies (ICRST'07)*, Springer, 2007, pp. 84-99.

- [53]C. Sun, Z. Wang, G. Wang. A Property-based Testing Framework for Encryption Programs. *Frontiers of Computer Science*, Springer, 2014, 8(3): 478-489.
- [54]C. Sun, Z. Wang, Q. Wen, P. Wu, T.Y. Chen. An Empirical Study on Iterative Metamorphic Testing for Web services. *International Journal on Software Tools for Technology Transfer* (under review, submitted on 24 March 2018).
- [55]C. Sun, H. Dai, H. Liu, T.Y. Chen, K. Cai, Adaptive Partition Testing. *IEEE Transactions on Computers* (under review, submitted on 29 January 2018).
- [56]C. Sun, G. Wang, Q. Wen, D. Towey, T. Y. Chen. MT4WS: An Automated Metamorphic Testing System for Web Services. *International Journal of High Performance Computing and Networking*, 2016, 9(1): 104-115.
- [57]孙昌爱, 付安, 张媛钰, 张在兴. MT4WS: 面向 Web 服务的蜕变测试系统, 第八届中国计算机学会服务计算学会会议“软件服务创新大赛”二等奖, 中国计算机学会服务计算专业委员会, 2017 年 9 月.
- [58]C. Sun, Y. Zhao, L. Pan, X. He, D. Towey. A Transformation-based Approach to Testing Concurrent Programs using UML Activity Diagrams, *Software: Practice and Experience*, 2016, 46(4): 551–576.
- [59]C. Sun, Y. Zhao, L. Pan, H. Liu, T.Y. Chen. Automated Testing of WS-BPEL Service Compositions: A Scenario-Oriented Approach, *IEEE Transactions on Services Computing*, in press (DOI: 10.1109/TSC.2015.2466572).
- [60]C. Sun, L. Pan, Q. Wang, H. Liu, X. Zhang. An Empirical Study on Mutation Testing of WS-BPEL Programs, *The Computer Journal*, Oxford University Press, 2017, 60(1): 143-158.
- [61]孙昌爱, 潘琳, 赵彦. μ BPEL: 一个面向 BPEL 服务组装程序的变异测试系统, “第三届全国软件研究成果原型竞赛”二等奖, 中国计算机学会软件工程专业委员会, 2015 年 11 月.
- [62]孙昌爱, 王真, 潘琳. 面向 WS-BPEL 程序的变异测试优化技术, 计算机研究与发展, 在审 (2017 年全国服务计算会议推荐论文).
- [63]孙昌爱, 王冠. MujavaX: 一个支持非均匀分布的变异生成系统, 计算机研究与发展, 2014, 51(4): 874-881.
- [64]C. Sun, F. Xue, H. Liu, X. Zhang. A Path-aware Approach to Mutant Reduction in Mutation Testing, *Information and Software Technology*, Elsevier, 2017, 81(1):65-81.
- [65]孙昌爱, 郭新玲, 张翔宇, 陈宗岳. 一种基于数据流分析的冗余变异体识别技术, 计算机学报, 2018, 刊出中 (录用日期: 2018 年 1 月 24 日).

二、研究采取的技术路线与方法

1. 研究内容

在并发程序测试、蜕变测试等领域前沿研究成果基础上，围绕并发程序测试中测试结果判定与高效的测试用例生成等关键问题开展研究，探索面向并发程序的蜕变测试框架与优化技术。主要研究内容如下：

(1) 面向并发程序的蜕变测试框架

软件测试的主要步骤包括测试用例生成、测试执行与测试结果判定。测试结果判定中的预期问题是软件测试领域尚未有效解决的开放问题。由于并发程序中并发流程执行的不确定性，测试结果判定更加困难，在某些情况下甚至无法确定测试预期。现有的并发程序测试技术通常假设已存在测试预期，当并发程序不存在测试预期时，这些技术的有效性则受到了很大的限制。本课题探索无需测试预期的并发程序新型测试方法。蜕变测试利用待测程序存在的某种内在属性（蜕变属性）生成测试用例（包括原始测试用例和衍生测试用例），通过原始测试用例与衍生测试用例对应的输出是否满足某种关系（蜕变关系）进行测试结果判定。**蜕变测试无须知道某个测试用例的预期输出，因此可以有效缓解测试预期问题。将蜕变测试应用于并发程序时，首先需要建立面向并发程序的蜕变测试框架。**研究如下问题：

- **并发程序的编程模型与故障特点分析：**并发程序设计引入了多线程、消息传递、Fork/Join等同步概念，不同线程访问公共变量时存在约束限制。支持并发的程序设计语言（如C、Java、BPEL等）的并发机制实现方式不同，相应地，并发相关的故障类型与特点也不同。
- **并发程序的故障检测机制：**测试用例生成是传统软件测试的关键步骤。并发程序测试中，为了增加触发某种交错性质的概率，不仅需要生成具有揭错能力的测试用例，还需要触发特定的执行交错。并发程序的故障检测不仅要关注测试用例生成方式，还要考虑如何尽可能地覆盖不同的执行交错。

(2) 面向并发程序的蜕变关系获取方法

识别蜕变关系是蜕变测试的核心问题。并发程序测试中即便执行了可能揭示故障的测试用例，也只有在特定的执行交错下才能检测到并发故障。因此，需要在各种交错模式的基础上，**提取同时覆盖待测程序内在的蜕变属性与不同执行交错模式的蜕变关系。**研究如下问题：

- **如何有效识别包含执行交错属性的蜕变关系：**由于并发程序执行的特点，并发故障只有在特定的执行交错下才能被揭示。因此，识别的蜕变关系满足衍生测试用例的执行交错尽量不同于原始测试用例的执行交错。
- **如何表示与选择包含执行交错属性蜕变关系：**与常规程序相比，并发程序的蜕变关系需要包含执行交错信息，需要解决的一个重要问题是如何描述此类新型蜕变关系；对于复杂的并发程序可能存在若干条蜕变关系，需要解决的另外一个重要问题是如何在不影响故障检测能力的情形下约减所需的蜕变关系。

(3) 面向并发程序的高效蜕变测试用例生成

原始测试用例与蜕变关系是影响蜕变测试的故障检测能力两个主要因素。在蜕变关系识别与选择的基础上，通过采用原始测试用例选择策略进一步提升并发程序的蜕变测试效率。面向并发程序的测试用例生成方法中，较为常见的是随机测试用例生成，该方法具有简单易用的特点，但没有利用并发程序的特点与执行交错信息，因而并发故障检测能力有待进一步提高。研究如下问题：

- **面向并发程序的高效原始测试用例生成：**增加测试用例集的多样性有助于更有效地检测软件故障。为了提高蜕变测试的故障检测效率，探索面向执行交错的并发程序覆盖准则以及生成满足不同覆盖准则的测试用例生成方法。
- **面向并发故障的蜕变测试用例集充分性分析：**采用多种度量指标（如F度量、T度量和P度量）评价满足不同覆盖率标准的蜕变测试用例集的故障检测有效性。

（4） 面向并发程序的蜕变测试技术的验证与有效性评估

以多个开源并发程序实例为研究对象，采用变异分析方法模拟可能的并发故障，评估面向并发程序的蜕变测试技术的可行性与有效性。

2. 技术路线与方法

针对并发程序的不确定性、测试预期问题更加突出，本课题探索将蜕变测试应用于并发程序的关键问题，包括面向并发程序的蜕变测试框架、面向并发程序的蜕变关系获取、面向并发程序的测试用例生成与充分性分析、面向并发程序的蜕变测试工具原型研制。拟采取的技术路线与方法如下：

（1） 通过并发程序的编程模型与蜕变测试原理相结合，构建面向并发程序的蜕变测试框架。

具体来说：（A）在并发程序的编程模型与故障检测方面，依据执行交错的通信方式，并发程序分为消息传递类型和共享内存类型。两种类型的并发程序中都存在不确定的执行交错，相应的并发故障类型可以归纳为不同的交错模式，包括数据竞争、原子违反、死锁。针对检测上述并发故障类型，测试用例的执行应能够驱动相应流程、并且覆盖不同的执行交错；（B）在构建面向并发程序的蜕变测试框架时，首先需要依据被测并发程序的蜕变属性识别蜕变关系，然后进行交错特征分析，在满足交错执行覆盖的基础上进行原始测试用例与衍生测试用例生成，分别执行原始与衍生测试用例，依据定义的蜕变关系验证测试结果。

（2） 根据并发故障的特点，获取并发程序的蜕变关系时，不仅需要满足待测程序的蜕变属性，还需覆盖不同的执行交错模式。具体来说：（A）在识别覆盖交错模式的蜕变关系方面，首先依据规格说明书定位并发相关操作，然后识别并发操作的蜕变属性并提取相应的蜕变关系，采用程序结构分析并在蜕变关系中进一步融合执行交错的相关信息，确保原始测试用例或者衍生测试用例的执行符合某种交错性质；（B）在并发程序的蜕变关系选择方面，探索基于某种交错性质的蜕变关系评价指标，在量化的评价指标基础上进一步开发贪婪方法、最大熵方法等蜕变关系选择策略，优先选择能够揭示并发故障类型的执行交错情形较多的蜕变关系。

- (3) 在并发程序的原始测试用例生成方面，定义新型交错覆盖准则并依据覆盖准则选择测试用例。具体来说：(A) 定义执行交错的覆盖度准则：随着线程数量以及程序规模的增加，执行交错数量可能存在“组合爆炸”问题。为了减少覆盖执行交错的数目，依据常见的交错性质（数据竞争、原子违反、死锁）提出相应的覆盖准则，使得测试用例的执行不仅驱动并发流程的执行，同时还覆盖不同的执行交错；(B) 对并发测试用例的充分性度量方面，采用F度量、T度量和P度量指标评价满足不同覆盖率标准的蜕变测试用例集的故障检测有效性。
- (4) 采用实例研究与变异分析相结合的方式，验证与评估面向并发程序的蜕变测试的可行性与有效性：(A) 为了模拟更多的并发故障类型，采用变异分析技术对常见并发算法程序集生成变异体集合。通过扩展面向传统Java程序的变异测试工具MuJava，支持并发程序的变异体生成；(B) 针对每一个程序实例的变异体集合，运用提出的面向并发程序的蜕变测试技术，统计反映故障检测能力的度量值，计算变异得分；(C) 进一步采用伯克利大学研制的并发程序集RADBench为研究对象（RADBench包含了Mozilla SpiderMonkey、Mozilla NSPR、Memcached、Apache Web引擎、Google浏览器等多种并发程序，并隐藏10个真实的并发故障），评估提出的蜕变测试技术的有效性。

三、拟解决的关键问题与创新之处

1. 拟解决的关键问题

- (1) 并发程序的执行方式与传统顺序程序的执行方式有很大的区别，相应地，并发故障不同于传统顺序程序的故障类型。面向并发程序的蜕变测试框架构建拟要解决的关键问题是**如何理解并发程序中故障类型与故障触发机制**。
- (2) 识别待测软件的蜕变属性是提取蜕变关系的前提。对于并发程序而言，不仅应考虑待测软件本身的蜕变属性, 还应满足特定的执行交错性质。面向并发程序的蜕变关系获取拟解决的关键问题是**如何识别、表示与选择包含交错属性的蜕变关系**。
- (3) 当蜕变测试应用于大规模复杂并发程序时，执行交错的数量非常庞大。由于只有满足特定的交错模式的测试用例才能检测出并发故障，在测试资源受限的情形下探索所有的执行交错是不可能的。面向并发程序的高效蜕变测试用例生成需要解决的关键问题是**如何定义执行交错的并发程序覆盖准则与生成满足不同覆盖准则的测试用例集**。
- (4) 采用实例研究与变异分析相结合的方式评估蜕变测试技术有效性时，需要大量模拟并发故障的并发程序变异体。人工生成变异体不仅消耗大量时间资源，而且易于出错。已有的变异测试工具（如MuJava）不支持模拟并发故障的变异算子，需要解决的关键问题是**如何研制支持并发故障的变异体生成工具**。

2. 本项目的创新之处

本课题研究面向并发程序的蜕变测试技术，预期研究成果将为开发可靠的并发程序提供测试的理论与工具支持。本项目的创新之处在于：

- (1) **针对并发程序突出的测试预期问题，提出面向并发程序的蜕变测试框架：**由于并发程序中并发流程执行的不确定性，测试结果判定更加困难，在某些情况下甚至无法确定测试预期。当并发程序不存在测试预期时，现有的并发程序测试技术无法适用。本课题探索符合并发程序特点的蜕变测试框架，利用待测并发程序的蜕变属性进行测试用例生成与测试结果判定，在测试预期不存在的情形下仍能对并发程序进行有效的测试。
- (2) **提出符合并发程序特点的蜕变关系识别与选择方法：**传统程序的蜕变关系仅仅依赖于规格说明。提出的面向并发程序的蜕变关系识别与选择方法融合了基于规格说明的蜕变属性与基于程序执行的交错覆盖，不仅满足规格说明中的蜕变属性, 还覆盖满足能够揭示某种交错性质的执行交错属性。
- (3) **提出面向并发程序的高效蜕变测试用例生成方法：**在传统的蜕变测试中，原始测试用例生成可以使用现有的测试用例生成技术。在并发程序的试中，测试用例的执行需要驱动流程的执行，同时还需覆盖不同的执行交错。通过定义新型交错覆盖准则及依据覆盖准则选择测试用例，提出一种面向并发程序的蜕变测试用例生成新方法，有助于改进面向并发程序的蜕变测试的故障检测能力。

四、与实验室研究工作的关系（请列出拟开展合作的实验室研究人员）

本课题拟与国家重点实验室的吴鹏博士展开合作。吴博士在并发理论、模型检验与测试领域有多年研究经历并取得丰富成果。特别地，吴博士在并发程序测试、蜕变测试方向均有论文发表。与本课题相关的论文如下：

- **Peng Wu**. Iterative Metamorphic Testing. *Proceedings of 29th Annual International Computer Software and Applications Conference (COMPSAC 2005)*, IEEE computer Society, Edinburgh, UK, 26- 28 July 2005, pp. 19-24.
- **Peng Wu**, Xiaochun Shi, Jiangjun Tang, Huimin Lin, T. Y. Chen. Metamorphic Testing and Special Case Testing: A Case Study. *Journal of Software*, 2005, 16(7): 1210-1220.
- **Peng Wu**, Huimin Lin. Model-Based Testing of Concurrent Programs with Predicate Sequencing Constraints. *Proceedings of the 5th International Conference on Quality Software (QSIC 2005)*, IEEE computer Society, Melbourne, Australia, 19-20 Sep. 2005, pp. 3-10.
- **Peng Wu**, Huimin Lin, Model-Based Testing of Concurrent Programs with Predicate Sequencing Constraints. *International Journal of Software Engineering and Knowledge Engineering*, 2006, 16(5): 727-746.
- **Peng Wu**, Daguang Liu, Huimin Lin. Symbolic Test Generation Using a Temporal Logic with Constrained Events. *Proceedings of the 19th IFIP International Conference on Testing of Communicating Systems and the 7th International Workshop on Formal Approaches to Testing of Software (TESTCOM/FATES 2007)*, Tallinn, Estonia, 26-29 June 2007.
- Han Yue, **Peng Wu**, T. Y. Chen, Yi Lv. Input-driven Active Testing of Multi-threaded Programs. *Proceedings of 22nd Asia-Pacific Software Engineering Conference (APSEC 2015)*, IEEE computer Society, New Delhi, India, 1-4 Dec. 2015, pp.246-253.
- Zhenya Zhang, **Peng Wu**, Yu Zhang. Localization of Linearizability Faults on the Coarse-grained Level. *Proceedings of 29th International Conferences of Software Engineering and Knowledge Engineering (SEKE 2017)*, Pittsburgh, USA, 5-7 July 2017, pp.272-277.

吴博士在并发程序测试、蜕变测试方面的经验与知识，将为本课题的顺利进展提供支持帮助。

五、合作研究人员意见（请列出合作基础以及具体的合作计划）

孙昌爱教授长期致力于软件测试方向的研究工作，与蜕变测试的提出者 T. Y. Chen 教授有长期合作关系。近年来，项目申请人在服务软件的蜕变测试方向进行了大量的研究工作，完成了国家自然科学基金、北京市自然科学基金课题多项，取得了丰硕的研究成果。

申请人与本人在并发程序的蜕变测试方面已经进行了深入讨论和交流，本课题研究方向的确立与本实验室的研究内容相关，合作基础扎实。并发程序设计有助于缩短程序执行时间，提高计算机系统的效率，在高性能计算、云计算、操作系统等多个领域有着广泛的应用。并发程序的测试是软件测试领域的一个难点。本课题旨在研究面向并发程序的新型测试技术，具有重要的理论意义与应用价值。

针对并发程序的不确定性，测试结果难以判定，测试预期更加突出的问题，本课题提出采用蜕变测试的方法缓解并发程序测试预期问题，规划了相应的研究内容与技术路线。本课题的预期目标合理，研究内容具体，技术路线清晰，预期成果可以完成。

六、以往承担实验室开放课题的情况

说明：请列出在开放课题基金支持下，且署名计算机科学国家重点实验室的，已发表（或已被录用）论文及鉴定或获奖的成果；并提供论文或成果说明的电子版或复印件，也请附上录用函及鉴定或获奖证书的复印件

项目名称：面向 Web 服务及其组合的蜕变测试技术与框架研究

课题编号：SYSKF 1105

年限起止：2011 年 1 月 ~ 2012 年 12 月 (已结题)

A. 标注开放课题资助并署名计算机科学国家重点实验室的论文（9 篇）

- Chang-ai Sun, Guan Wang, Zuoyi Wang. A Property-based Testing Framework for Encryption Programs. *Frontier of Computer Science*, Springer, 2014, 8(3):478-489.
- Chang-ai Sun, Yi Meng Zhai, Yan Shang, Zhenyu Zhang. BPELDebugger: An effective BPEL-specific fault localization framework. *Information and Software Technology*. Elsevier, 2013, 55(12): 2140-2153.
- Chang-ai Sun, Guan Wang, Baohong Mu, Huai Liu, Zhaoshun Wang, T.Y. Chen. A Metamorphic Relation-Based Approach to Testing Web Services Without Oracles, *International Journal on Web Service Research*, 2012, 9(1): 51-73.
- 孙昌爱, 程庆顺. 基于故障的布尔表达式测试技术综述. *计算机科学*, 2013, 40(3):16-23.
- 孙昌爱. 基于约束的软件失效域识别与特征分析, *软件学报*, 2012, 23(7): 1688-1701.
- Chang-ai Sun, Guan Wang, Kai-Yuan Cai, T.Y. Chen. Towards Dynamic Random Testing for Web Services, *Proceedings of 36th Annual IEEE International Computer Software and Application Conference (COMPSAC 2012)*, IEEE Computer Society, July 16-July 20, 2012, Izmir, Turkey. pp.164-169.
- Chang-ai Sun, Guan Wang, Kai-Yuan Cai, T.Y. Chen. Distribution-aware Mutation Analysis, *Proceedings of 9th IEEE International Workshop on Software Cybernetics (IWSC 2012)*, collocated with *COMPSAC 2012*, IEEE Computer Society, July 16-July 20, 2012, Izmir, Turkey. pp.170-175.
- Chang-ai Sun, Yan Shang, Yan Zhao, T.Y. Chen. Scenario-Oriented Testing for BPEL Service Compositions, *Proceedings of 12th International Conference on Quality Software (QSIC 2012)*, IEEE Computer Society, 27-29 August 2012, Xi'an, China. pp.171-174.
- Chang-ai Sun, Yi Meng Zhai, Yan Shang, Zhenyu Zhang. Toward Effectively Locating Integration-Level Faults in BPEL programs, *Proceedings of 12th International Conference on Quality Software (QSIC 2012)*, IEEE Computer Society, 27-29 August 2012, Xi'an, China, pp.17-20.

B. 登记计算机软件著作权3项

- 孙昌爱, 王冠. 面向 Web 服务的蜕变测试软件 MT4WS V1.0, 登记号：2012SR128924；登记日期：2012.12.20.
- 孙昌爱, 赵彦. 基于变换的 UML 活动图模型测试路径生成工具 TestGen V1.0, 登记号：2012SR105186；登记日期：2012.11.8.
- 孙昌爱. 面向布尔表达式的高效测试用例生成软件 BEAT-WEB V1.0, 登记号：2012SR023267；登记日期：2012.3.26.

C. 研究生培养情况（9名）

- 程庆顺：硕士学位论文题目为“面向通用布尔规格说明的 MUMCUT 的性能分析与改进”，2011 年 12 月毕业。
- 王冠：硕士学位论文题目为“面向 Web 服务的蜕变测试技术与工具研究”，2012 年 12 月毕业（北京科技大学优秀硕士生学位论文，研究生国家奖学金获得者，北京科技大学计通学院“学术之星”）。
- 薛铁恒：硕士学位论文题目为“基于 VxBPEL 的适应性服务组装平台的研究与实现”，2012 年 12 月毕业（北京市优秀硕士研究生，北京科技大学优秀硕士生学位论文，国家奖学金获得者，北京科技大学计通学院“学术之星”）。
- 尚岩：硕士学位论文题目为“基于 Aspect 的服务组装中事务集成与管理的研究与实现”，2012 年 12 月毕业（北京科技大学优秀硕士生学位论文）。
- 赵敏：硕士学位论文题目为“面向 Web 服务的可信性度量技术研究”，2012 年 12 月毕业。
- 温情：硕士学位论文题目为“面向 BPEL 服务组装的蜕变测试技术与工具研究”，2014 年 1 月毕业。
- 赵彦：硕士学位论文题目为“基于场景的 BPEL 测试用例自动生成技术与工具研究”，2014 年 1 月毕业（国家奖学金获得者）。
- 翟忆蒙：硕士学位论文题目为“面向 BPEL 程序的故障定位技术和工具研究”，2014 年 1 月毕业（国家奖学金获得者）。
- 王可：硕士学位论文题目为“服务组装中的可变性设计与管理技术及其支持平台研究”，2014 年 1 月毕业。

D. 开放课题部分资助的学术交流情况

应邀在如下国际研讨上交流开放课题资助的研究成果：

- Chang-ai Sun. Techniques and Tools for Adaptive and Reliable SOA Software. An invited talk on International Workshop on Software Cybernetics, 22-23 September 2011, Beijing, China.
- Chang-ai Sun. Metamorphic Testing for Web Services: A Framework and Case Studies. An invited talk on International Workshop on Evolving Critical Software Systems (IWECS 2011), 6-7 Dec. 2011, Shanghai, China.

在如下国际会议或全国学术会议宣读开放课题资助的研究成果：

- IEEE 9th International Conference on Web Services (*IEEE ICWS 2011*), July 4-9, 2011, Washington D.C., USA.
- 36th Annual IEEE International Computer Software and Application Conference (COMPSAC 2012), July 16-July 20, 2012, Izmir, Turkey.
- 12th International Conference on Quality Software (QSIC 2012), Society, 27-29 August 2012, Xi'an, China.

参加的全国学术会议：

- 2011 年全国软件与应用学术会议(NASAC 2011), 2011 年 10 月 27 - 29 日，吉林长春市.
- 2011 年中国计算机学会服务计算学术会议 (CCF NCSC 2011), 2011 年 8 月 18-19 日，山东济南市.
- 2012 年全国软件与应用学术会议(NASAC 2012), 2012 年 10 月 20 -22 日，江苏南京.
- 2012 年中国计算机学会服务计算学术会议 (CCF NCSC 2012), 2012 年 8 月 16-18 日，陕西西

安.

国际学术交流情况:

- 应 Fevzi Belli 教授和 Marco Aiello 教授的邀请, 访问德国帕德博恩大学 (University of Paderborn) 和荷兰格罗宁根大学 (University of Groningen) 1 个月, 交流面向 Web 服务与组装的蜕变测试技术与工具方面的研究进展。
- 邀请澳大利亚斯文本大学 T.Y. Chen 教授、Jun Han 教授来华学术交流 2 次, 就软件测试、服务组装等主题进行深入地探讨, 分别做题为“Diversity, Diversity and Diversity in Test Case Selection”、“Adaptive Business Processes for Multi-tenant Cloud/SaaS Applications”学术报告。

七、申请经费及预算

说明: 申请经费主要包括科研业务费^(注一)、材料费^(注二)、其他^(注三)等, 预算需合理、详细, “计算依据”需列出费用名称及金额, 预算中不得包含固定资产类的设备购置费。

1. 申请总金额: 12万元人民币。

2. 总预算:

名称	金额 (元)	计算依据 (细化预算项目)
科研业务费	7.0	参加国内学术会议或调研 4 次, 每次 0.5 万元, 包括会议注册、交通、住宿等; 出国参加国际会议 2 次, 每次 2 万元左右, 包括会议注册、来回机票、住宿等; 文章发表版面费、资料文献与信息检索费 1 万元。
材料费	1	复印、打印、日常办公用品、实验室耗材等 (每月 200 元左右)。
其他	4	按学校规定, 项目组织管理费 0.6 万元; 补贴直接参与项目的博士生和硕士生 3.4 万元 (每人月 200 元左右)。
合计	12 元人民币	

其他要说明的问题:

无。

(注一): 科研业务费, 指进行课题研究、业务活动所用开支。主要包括机时费、网络费; 会议费、差旅费、学术交流费、调研费; 报告及论文的印刷费、版面费; 图书资料费等。

(注二): 材料费, 指科研所需材料的购置费。主要包括存储、刻录设备; 软件工具; 打印、复印材料; 其他易耗材料等。

(注三): 其他, 主要包括协作费、管理费等。协作费指支付给外单位协作者的费用; 管理费指项目组织实施所用开支。

八、申请人所在单位学术委员会意见

并发程序设计广泛用来解决计算机系统资源充分利用、提升计算速度。并发程序的行为具有不确定性，通过有效的测试保证并发程序的正确性是软件工程领域亟待解决的难题之一。本课题研究面向并发程序的蜕变测试技术，属于国际前沿基础性研究。本课题建立在申请者近几年来丰硕的科研成果与实践经验基础之上，在理论上具有重要意义，并有广阔的应用前景。

申请者学风严谨、在软件测试领域具有较深的造诣，已经有多年坚实的研究基础和丰硕的研究成果。

鉴此，同意申报。

学科负责人：

申请人所在单位盖章：

如有推荐人，推荐人意见：

推荐人签字：

推荐人职称：

所在单位：

与申请人关系：

申请人保证以上所填内容属实并签字：

申请人所在单位领导签字：

申请日期： 年 月 日

单位盖章：

九、计算机科学重点实验室学术委员会审批意见

学术委员会主任签字：

年 月 日