文章编号:1008-0570(2010)12-1-0161-03

# 软件故障诊断技术综述

*A Survey of the Technique for Software Fault Diagnosis*

(湖南工业大学) 王志兵 李长云
WANG Zhi-bing LI Chang-yun

**摘要:** 软件系统的日益复杂及广泛应用,使其可用性、可靠性和可维护性等方面面临许多新的挑战。快速的对系统中出现的故障进行检测和定位,对于不断建立一个高可靠的系统和缩小平均修复时间意义重大。简要介绍了软件故障诊断中的基本概念,较系统地总结了故障检测与定位技术的研究进展和成果。

**关键词:** 软件故障诊断; 软件故障检测; 软件故障定位
**中图分类号:** TP311 **文献标识码:** A

**Abstract:** The increasingly complicated software system and its wide application have presented many new challenges to practicability, reliability and reparability of the system. Whether the fault that emerges in the system can be quickly detected and localized has great significance for continuously building a highly reliable system and reducing the average repairing time. After a brief introduction to the basic concepts in software fault diagnosis, this paper systematically summarizes the development and the achievements of the research into the technique for software fault detection and localization.

**Key words:** software fault diagnosis; software fault detection; software fault localization

技术创新

## 引言

高可靠和复杂的系统非常依赖于其采用的软件的可靠性,一个未能及时而正确检测到的软件故障可能造成整个系统的失效、瘫痪,甚至导致巨大的灾难性后果。例如,1996年6月,欧洲"阿丽亚娜"号航天飞机因导航系统的计算机软件出现故障,致使航天飞机坠毁,造成了数亿美元的巨大损失;2005年4月,软件失灵、继而导航失误,导致耗资1.1亿美元的NASA自主交会任务DART实验失败。一方面,软件系统的日益复杂及故障诊断在诸如飞行控制、化工、发电厂等重要领域的应用需求使得软件故障诊断技术激发了国外内研究者的兴趣;另一方面,随着现代科学技术水平的日益提高,软件系统的规模越来越大、功能越来越复杂,人们对其的可用性、可靠性和安全性等可信性质给予了更高的期望和要求。

理论研究表明,现有的技术途径不能完全消除软件故障问题。目前,提高软件质量,消除软件故障主要依赖软件过程改善技术,但是这些技术都有一定的局限性。软件测试在一定程度上可以消除软件故障,改善系统可靠性,但不能保证复杂系统代码执行对于所有可能输入都是正确的,无法做到穷尽式测试。软件模型检查等形式化验证技术针对的常常是模型,而不是实际的软件运行;即使在需求阶段,软件性质能够通过形式化技术进行验证,软件故障也可能在软件生命周期的实现或者维护阶段被引入,系统运行环境也可能导致软件故障。因此,如何提高系统的可靠性,防止和杜绝影响系统正常运行的故障的发生和发展已成为一个重要的有待解决的问题,软件故障诊断技术也就因此而有着极为重要的意义。

王志兵:硕 士

基金项目:项目申请人:李长云;项目名称:分布式软件运行时诊断技术研究;颁发部门:国家自然科学基金(CX2009B200)项目名称:开放环境下的软件动态演化研究(60773110)中国博士后科学基金(20080440216);湖南省自然科学基金(09JJ6087);湖南省研究生创新基金;湖南省教育厅科研项目(09C325).

## 1 软件故障诊断

软件故障诊断是判定软件是否存在错误、缺陷和故障,以及分析推断错误、缺陷和故障的位置和原因的过程,是保证软件质量和提高软件可靠性的重要手段。软件故障诊断是根据软件(包括程序、数据和文档)的静态表现形式和动态运行状态信息查找故障源,并确定采取相应决策的一门技术。软件故障诊断突出了诊断的目的性,即寻找和发现软件故障症状而进行诊断。

软件故障诊断的过程包括故障检测、故障定位和故障排除等几个阶段。软件故障检测是软件故障诊断的第一步,通过静态检查、动态运行等方法获取软件中的各种信息,获得可能出现软件故障的征兆,识别软件是否正常运行或存在故障,并为软件故障定位提供依据。软件故障定位,是指根据软件故障检测提供的能反映软件状况的征兆或特征参数的变化情况,或与某故障状态参数(模式)进行比较,并进一步收集软件的历史和使用信息,识别软件是否正常运行或存在故障,复现软件故障过程,诊断软件故障的性质和程度、产生原因或发生部位,确定缺陷,为纠正缺陷、排除软件故障做好准备。软件故障排除是指当诊断出软件中存在缺陷,决定纠正缺陷、排除故障的办法,包括修改程序代码、数据或软件文档等。

在本文中我们主要关注软件故障检测和故障定位两个方面。

### 1.1 软件故障检测

软件故障检测是软件故障诊断的前提,只有当检测出软件出现故障后才能进行故障定位工作。软件故障检测是指采集软件系统运行参数,获取系统状态信息,从系统可测量或不可测量变量的估计中,判断被诊断系统是否发生了故障。软件故障检测

是通过研究故障与其症状之间的关系来判断被检测软件的状态。由于软件的复杂性以及故障证据可能是模棱两可的、不一致的和不完全的，故障与征兆之间的关系很难用精确的数学模型来表示。现有的软件故障检测主要采用测试、形式化验证及运行时验证等技术手段。

(1)测试。软件测试是为了发现程序中的错误而执行程序的过程,通过定义各种测试充分性,可以提高我们对被测软件质量的信心,但无法回答系统一定没有错误这样一类问题,因而无法从根本上确保系统的可靠性。传统的测试作为软件开发的一个必要组成部分,一般包括单元测试与集成测试,主要用于软件开发阶段的错误发现,属于一种静态的检测方式。与程序切片、形式化验证等技术相结合的测试手段现在得到了较好的应用。

(2)形式化验证。形式化验证则通过基于数学的形式化方法,揭示系统的不一致性、歧义性和不完备性,可以从某一个角度回答系统一定没有错误这样一类问题，提高我们对系统可靠性的信心,往往被应用于一些关键领域的软件系统。由于其验证对象一般为系统的模型，所以形式化验证一般也属于一种静态检测技术。文献提出利用定理证明的手段来保证软件产品可靠性和正确性。定理证明以逻辑推理为基础,根据公理和形式推演规则来验证设计实现是否满足要求,由于需要人工干预,能处理的软件规模较小。Clarke EM 等率先将模型验证技术引入软件故障诊断领域，用某种形式化方法说明系统应拥有的性质和应满足的属性,通过对模型进行检测,判断软件系统是否拥有这些期望的属性。软件模型验证技术可以分为对系统说明 (需求分析)的模型检测和针对源代码的模型检测。针对于不同的应用，已经开发出来了一些较为成熟的软件模型检测工具，其中比较典型的有前面提到的 SPIN、加州大学伯克利分校的 BLAST 系统和 MOPS 原型系统、卡内基梅隆大学开发的 MAGIC 系统、美国国家航空航天局开发的 JPF、微软的 SLAM 和 Zing 项目。

(3)运行时验证。鉴于现代分布式软件系统具有松散聚合及开放动态等特性，以及软件故障除了由软件本身的错误和缺陷造成外,也可能由运行环境所导致,因此不仅需关注开发期的静态检测,还需注重运行期的在线检测。运行时验证简单地说可以看作软件测试和形式化方法的结合，它是一种保证程序可靠性的轻量化方法，其基本思想是收集软件运行中的相关信息并与软件需求规约进行比照,如不满足预定义的相关需求,则提示检测到故障;或者在软件中设定一些阀值,当软件运行过程中超过阀值设定的范围时,则提示检测到故障。目前的运行时验证技术主要有基于 AOP 的技术、基于截获器的技术、基于构件包装器的技术及基于框架的技术四种。其中基于框架的技术是目前的热点研究领域,如 Monitoring-oriented programming (MoP)、Java with Assertions (Jass)、Java PathExplorer (JPaX)、Java Run-time Timing-constraint Monitor(JRTM)、基于程序运行形式化分析的软件故障监控模型等框架。

1.2 软件故障定位

故障定位是软件故障诊断中的关键问题，它的难点是如何在大量的相关故障事件中找到故障源。国内外学者针对故障定位问题,进行了深入的研究,提出了很多行之有效的理论和方法,它们都来源于计算机科学的不同领域,包括人工智能、图形理论等。故障定位技术一般归为三类。第一类是基于人工智能的技术,包括专家系统、神经网络、决策树。专家系统又可以细分为基于规则的系统、基于模型的系统和基于范例的系统。第二类是模式穿越技术。这种技术使用形式化的方法描述分布系统实体之间的关系。故障会依照这种关系在实体间传播,通过探测这些关系,从发生故障的实体开始回溯,即可找到其它相关故障的实体和原因实体。第三类是故障传播模型。这种模型企图在故障之间建立一种因果关系,通过这种因果关系,根据现象和传播模型推测故障原因,也可以用这种模型来预测故障的发生,故障传播模型包括贝叶斯网络、因果关系图和依赖关系图等。在软件故障定位领域目前得到应用的的主要技术是基于模型的定位技术、模式穿越技术等。

当软件发生故障时,需要进行故障定位,深入代码的内部找出故障模块和故障代码。在软件故障定位中一般根据是否需要系统内部结构及行为知识分为白盒技术和黑盒技术[20]。白盒技术需要系统内部构件结构及行为知识，基于模型的故障定位技术是一种典型的白盒技术。在基于模型的定位推理中,它建立系统的形式化模型,从运行时监测器获得运行信息,通过逻辑推理定位故障源。模型反映了实际系统的结构和行为,由于基于模型的方法能够反映系统底层的详细细节，因此这种方法可以用来解决一些新出现的故障；它的诊断知识可以被组织成可扩展的形式,而且容易升级和模块化。不过,由于模型的知识难以获得,这种诊断需要详细的系统底层知识,对于不同的目标系统,它们的模型基本都不相同;同时由于软件系统规模越来越庞大、复杂,系统模型也变得十分复杂,模型的维护将更加困难。易昭湘等提出的基于代码检测的软件故障定位方法也属于白盒定位技术,获取软件发生故障时的模块运行序列,分析出软件故障可疑模块集,在此基础上对故障模块进行代码的分类检测。

黑盒定位技术由于不需要系统内部构件结构及行为知识,得到了更加广泛的应用。基于程序谱的定位技术是一种重要的黑盒技术。基于程序谱的故障定位一般不依赖于输入结构知识,而依赖于程序中能够和源码某个位置关联的程序执行特征,这些特征的整体可被称为程序谱。程序谱特征化程序行为、或者提供了程序行为的信号,如路径谱、分枝谱等,它可为故障定位提供重要知识。Tarantual 和 AMPLE 都是利用程序谱来进行故障定位。黑盒定位中用到的技术还包括动态程序切片技术及近邻模型技术等。

## 2 典型的软件故障诊断系统

伴随软件技术发展的同时,出现了一些软件诊断系统,代表性的包括:PinPoint、Tarantual 和 AMPLE 等。

美国斯坦福大学和加州伯克利分校的研究人员认为系统的故障和操作失误是不可改变的事实。因此,他们不再试图降低系统的失效率,而是致力于设计一种能够从故障中恢的系统。他们开发了基于面向复原计算(Recovery Oriented Computing,ROC)思想的故障诊断系统 Pinpoint。Pinpoint 是 J2EE 平台上的一个根原因分析框架,目标是大型、动态的因特网服务,比如邮件服务和搜索引擎。系统由客户请求跟踪、失效检测及数据聚类分析三个主要部分组成,当用户访问某个服务时,客户请求跟踪部件跟踪所有参与本次服务的构件,并进行记录;失效检测部件检测本次服务是否成功；数据聚类分析部件分析和比较请求成功和请求失败过程中，都有哪些构件参与了该过程的,Pinpoint 可以找出产生故障的最可疑构件。PinPoint 应用这种诊断方法的前提是假设系统的故障都是单故障,即同时只有一个故障发生,它很难排除多故障的情况。

技术创新

Tarantual 是针对 C 语言程序开发的一个故障定位工具,是应用基于程序谱故障定位技术的系统。Tarantual 提供一个图形化的用户接口,用彩色索引解释相似系数的计算值,用于直观化显示可疑的程序语句。通过在测试套件中执行有故障的程序,Tarantual 能提供测试过程中程序所有参与语句的可视化映射,参与语句被用从红色到黄色再到绿色的连续光谱表示,若一个语句参与的测试是高成功率的,则该语句用明亮的绿色表示,反之若一个语句参与的测试是高失败率的则用明亮的红色表示。Tarantual 仅仅包含一个故障定位部件,需要依赖外部错误检测来分类正常或错误的运行。

AMPLE(分析方法模式去定位错误)是一个针对面向对象软件的故障诊断系统,AMPLE 通过对收集到的方法调用序列的分析进行故障定位。AMPLE 诊断是在类层次,计算的系数也是来源于类收集的证据,并不能识别可疑的方法调用次序。AMPLE 通过一个滑动窗口特征化一个方法调用序列为多个调用子序列,通过比较一次失败序列和多次成功序列中各方法调用的权重定位故障类。

## 3 结论与展望

由于松散聚合、开放动态和行为复杂等特性,软件系统诊断面临着一系列的挑战,主要表现为:故障形式多种多样,其本质也有很大差别;故障证据有时是不一致和不完整的,可能牵涉到多个分布的实体;开放动态性给软件系统的时效性、可靠性和持续可用性带来了新的考验;运行环境可能导致软件故障,需对其充分考虑;软件处于动态演化过程中,难以及时捕获故障等等。软件故障与软件的运行有紧密的联系,因此需要根据软件的运行过程进行软件故障诊断。为此,在软件系统中嵌入特定的模块,监控软件的运行状态,实时记录软件运行状态和数据。在软件故障发生时,依据记录的数据采用形式化方法、统计分析等方法分析出软件系统中模块的运行序列、数据输入和数据输出,进行软件故障诊断。

参考文献

[1]王怀民,唐扬斌,尹刚,等.互联网软件的可信机理. 中国科学 E 辑:信息科学[J],2006,36(10):1156-1169.

[2]单锦辉,徐克俊. 软件故障诊断探讨[J]. 北京化工大学学报,2007,34(SI):5-8.

[3]张迎周,徐宝文. 一种新型形式化程序切片方法[J]. 中国科学 E 辑:信息科学[J], 2008,38(2):161-176.

[4]孙继荣,李志蜀,王莉,等. 程序切片技术在软件测试中的应用[J]. 计算机应用研究, 2007,24(5):210-213.

[5]徐宝文,聂长海,史亮,等. 一种基于组合测试的软件故障调试方法[J]. 计算机学报, 2006,29(1):132-137.

[6]Fenkam P, J azayeri M , Reif G. On met hodogies for const ructing correct event-based applications [C] //Proc of t he 3rd Int Workshop on Dist ributed Event-Based Systems.New York USA, 2004 : 38-42.

[7]Clarke E M , Grumberg O, Peled D A. Model checking [M]. Cambridge, Massachusetts: The MIT Press,1999.

[8]Armin Biere. Tutorial on Model Checking Modelling and Verification in Computer Science [C]. In Proc. of the 3rd international conference on algebraic biology (AB'08'), Volume 5147 of LNCS. Springer,2008.

[9]Bodden E. A lightweight LTL runtime verification tool for Java [C]. Proc of OOPSLA 2004. 2004: 306-307.

[10]L Mariani, M Pezze. Technique for verifying component-based software [J]. Electronic Notes in Theoretical Computer Science , 2005, 116 (1): 17-30.

[11]GAO J, ZHU Y, SH IM S, et al. Monitoring software components and component-based software [C].Proc of the 24 th IEEE Annual International Computer Software and Applications Conference. Taiwan , 2000: 403-412.

[12]LI Jun. Monitoring and characterization of component-based systems with global causality capture [C]. Proceedings. 23rd International Conference on Distributed Computing Systems, 2003. 422-431.

[13]F. Chen and G. Ro?u, Towards Monitoring-Oriented Programming: A Paradigm Combining Specification and Implementation[J]. Electronic Notes in Theoretical Computer Science, vol. 89, Elsevier, 2003.

[14]D. Bartetzko, Jass-Java with Assertions [C]. Proc. of RV'01, Paris, France, Jul. 2001.

[15]K. Havelund and G. Ro?u, Monitoring Programs using Rewriting[C]. Proc. of the 16th IEEE Int'l Conf. on Automated Software Engineering. 2001:135-143.

[16]M. Kim, S. Kannan, I. Lee, et al. Java-MaC: A run-time assurance approach for Java programs [J]. Formal Methods in System Design, 2004:129-155.

[17]刘彦斌,朱小冬. 基于 Multi-agent 的实时系统运行故障监控研究[J]. 微计算机信息, 2006,10-1:224-226.

[18]S.Katker,M.Paterok. Fault isolation and event correlation for integrated fault management[C]. In IM'97, 1997:583 - 596.

[19]M Steinder, AS Sethi. A survey of fault localization techniques in computer networks[J]. Science of Computer Programming, 2004: 165-194.

[20]P Zoeteweij, R Abreu, AJC Van Gemund. Software Fault Diagnosis[C]. 19th IFIP International Conference on Testing of Communicating Systems, Tallinn, Estonia, June 2007.

[21]J. de Kleer and B. C. Williams. Diagnosing multiple faults[J]. Artif. Intell., 32(1):97 - 130, 1987.

[22] 易昭湘,慕晓冬,赵 鹏,等.基于代码检测的软件故障定位方法[J].计算机工程, 2007,33(12), 82-83.

[23]R. Abreu, P. Zoeteweij, R. Golsteijn, et al, A Practical Evaluation of Spectrum-based Fault Localization [J]. Journal of Systems and Software (JSS), Elsevier, 2009.

[24]刘彦斌,朱小冬.基于双轨迹差异分析法的软件故障定位[J]. 计算机工程, 2007,33(9): 43-45.

[25]J. A. Jones and M. J. Harrold. Empirical evaluation of the tarantula automatic fault-localization technique[C]. In Proceedings of the 20th IEEE/ACM international Conference on Automated software engineering, NY,USA, 2005: 273 - 282. ACM Press.

[26]A. Jones, M. J. Harrold, and J. Stasko. Visualization of test information to assist fault localization [C]. In Proceedings of the 24th International Conference on Software Engineering, Orlando, Florida, USA, May 2002: 467 - 477. ACM Press.

[27]H. Agrawal, R. A. DeMillo, and E. H. Spafford. Debugging with dynamic slicing and backtracking [J]. Software Practice and Experience, 23(6):589 - 616, 1993.

[28]M. Renieris and S. P. Reiss. Fault localization with nearest neighbor queries[C]. In Proceedings of the 18th IEEE International Conference on Automated Software Engineering, Montreal, Canada, October 2003.

(下转第 211 页)

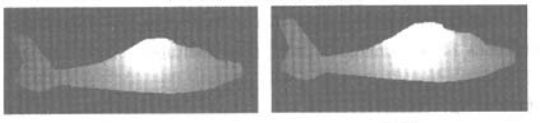
技术创新

计算出每个表面片的灰度后,按光能辐射度法,计算出每个表面片各个顶点的灰度,最后用 Gouraud Shading 绘制出每个表面片。

3.2 图像模拟结果

以某型武装直升机为例(如图 2 所示),整个空中直升机目标红外图像的生成采用 Visual Studio 2008 加 OpenGL 图形库设计在 Windows 平台上。采用适当的图形绘制方法得到直升机红外辐射图(如图 3 所示),计算条件为春季正午,金属综合导热系数为 35W/m.K,表面材料的辐射率为 0.9,接受波段为 $8\sim12\mu m$,根据直升机飞行速度和发动机排气管温度曲线,取两点见表 1。

图 2 某型武装直升机



(a)平飞 160 (b)平飞 250

图 3 不同状态下模拟仿真结果

由图 3 所示,前机身和机尾部分,其温度较低,和环境温度相接近;发动机舱盖区域及其周围区域较高。显然,结果是符合实际情况的。首先,机头和机尾离发动机较远,受到发动机影响较小,主要受环境影响;发动机舱盖区域由发动机的热作用引起,发动机舱盖区域周围区域由蒙皮表面热扩散引起。

表 1 飞行速度与排气管温度样本数据

| 任务状态 | 速度（$km/h$） | 燃气发生器转速(%) | 排气温度（K） |
|---|---|---|---|
| 平飞 | 160 | 88.0 | 813 |
| 平飞 | 250 | 98.0 | 953 |

图 3 中,通过比较(b)和(a)可知,随着直升机飞行速度的加快,发动机舱盖区域及其周围区域温度明显升高,范围也有所增加,而前机身和机尾部分基本不变,这是因为速度加大,发动机的外壳温度升高,热作用增强,因此,受它们影响的机体蒙皮范围增加,温度相应也变高。

## 4 结论

从红外辐射和热传导基本定律出发, 综合考虑目标实际结构、太阳辐射、天空长波辐射、空气对流及周围环境辐射等因素的影响,以某型武装直升机为代表建立起了红外热传导模型,通过对内热源向表面热传递、面片间热扩散以及空气对流效应在飞机蒙皮相应部位作用机理的进一步分析, 确定内热源向外表面传递热能的主要形式, 计算出到达热平衡时的表面温度场分布,建立了一个含有内热源的直升机红外辐射成像模型。然后采用辐射度方法绘制出直升机在平飞两个速度状态下的红外图像。本文较好地模拟了由内热源引起的直升机目标的红外图像,真实感较强。

本文创新点: 本文建立了一个含有内热源的直升机红外辐射成像模型, 成功解决了飞行模拟器中夜航制导训练的视景显示问题,具有很高的军事价值。

参考文献

[1]姚涛,李一凡.场景红外成像仿真原理和应用[J].计算机仿真,2004,21(1):96-98.

[2]章熙民,任泽霈,梅飞鸣.传热学[M].北京:中国建筑工业出版社,2007.104-106.

[3]HONG Hyun-ki,HAN Sung-hyun,HONG Gyoung-pyo,eta1. Simulation of reticle seekers using the generated thermal images [C]//Circuits and System,1996.IEEE Asia Pacific Conference,1996: 183-186.

[4]于伟杰,彭群生.基于红外物理与传热学的空中飞行目标红外成像模型[J].系统仿真学报,1998,l0(6):7-12.

[5]王学伟,张卫国,沈同圣.飞机目标动态红外图像的计算机生成[J].红外与激光工程,1999,28(2):21-24.

[6]沈同圣,严和平,周晓东.海洋作战环境动态红外图像的计算机仿真[J]. 红外与激光工程,1998,27(4):9-13.

[7]高嘉,方宁,王宝发,宁焕生.复杂目标红外辐射特性可视化仿真研究[J]. 微计算机信息, 2009, 6-1: 185-186.

**作者简介:**赵永(1985-),男,硕士研究生,主要研究方向:红外成像仿真; 姚连钰(1961-),男,硕士,副教授,空军航空大学军事仿真技术研究所,主要研究方向:视景仿真。

**Biography:**ZHAO Yong (1985-) , male ,graduate student , major in IR imaging simulation ;Yao Lian-yu (1961-),male, master, associate professor, Military Simulation Technology Institute , Air Force Aviation University, major in scene simulation.

**(130022 吉林长春 空军航空大学军事仿真技术研究所) 赵 永 姚连钰 李松维 陈 蕾**

**(Military Simulation Technology Institute Air Force Aviation University, Changchun Jilin 130022, China) ZHAO Yong YAO Lian-yu LI Song-wei CHEN Lei**

**通讯地址:(130022 吉林省长春 南湖大路 2222 号空军航空大学研究生队) 赵 永**

(收稿日期:2010.05.28)(修稿日期:2010.08.28)

(上接第 163 页)

[29]Chen M,E.Kieiman,E.Fratkin,et al. PinPoint:Problem Detemrination in Lagre, Dynamic, Intenret Services [C].Poreeedings of the Intenrational Conefreneeon DePendbale Systems and Networks (IPDS Track),Washington D.C,2002.

[30]D Patterson, A Brown, P Broadwell, et al. Recovery Oriented Computing (ROC): Motivation, Definition, Techniques[R]. University of California at Berkeley, CA, USA. 2002

**作者简介:**王志兵(1974-),男,硕士,实验师,主要研究领域为可信软件。

**Biography:**WANG Zhi-bing (1974- ), male, master; major researching aspect: trusted software.

**(412008 湖南株洲 湖南工业大学计算机与通信学院) 王志兵 李长云**

**(Hunan University of Technology, Zhuzhou 412008, China) WANG Zhi-bing LI Chang-yun**

**通讯地址:(412008 湖南株洲 湖南工业大学计算机与通信学院) 王志兵**

(收稿日期:2010.04.21)(修稿日期:2010.07.21)