

软件可靠性

评估系统

软件工程
Agent

⑥

基于 Agent 的软件可靠性评估系统

Agent-based System of Software Reliability Evaluation

董 威 毛新军 陈 磊 王 戟 齐治昌

(国防科技大学计算机学院 长沙 410073)

TP311.5

Abstract Software reliability evaluation is a challenging research in the area of software engineering. To the features of software reliability and localization of current system for software reliability evaluation, we suggest an Agent-based system of software reliability evaluation in this paper. The system constructs an open, flexible software structure based on the method of Multi-agent system. It fulfills software reliability evaluation and reliability models selection with contract net protocol in DAI. We also give the Web-based implementing method of the system, and discuss the features of system in the end.

Keywords Agent, Software reliability model, Software reliability evaluation system, Contract net

1 引言

现代软件产品的重要特征之一是软件质量,而软件可靠性被认为是软件质量的关键因素。软件可靠性指软件在特定环境和给定时间内无故障运行的概率^[1]。它面向用户而不是开发者,是软件质量诸多因素中最容易度量的属性。软件可靠性模型是随机过程的一种表示,可以将软件可靠性或与其直接有关的量(如平均无故障时间或故障率)表示成时间以及软件产品特性或开发过程的函数。软件可靠性评估以软件可靠性模型为基础对软件进行可靠性评价和预测,以在软件开发和维护过程中对软件可靠性进行管理和控制。

软件可靠性很大程度上反映了软件开发组织和人员的个性,因而其规律性难以刻画,不同的软件系统可能表现出不同的软件可靠性特征。这一特点与目前出现的许多软件可靠性模型是相一致的。现阶段没有一个普遍适用的模型能对所有的产品都作出最好的可靠性评估,不同的软件项目可能需要选择不同的、对其适用的软件可靠性模型。因此在软件可靠性评估中,模型选择是关键。目前国内外的软件可靠性评估系统有 AT&T 软件可靠性工程工具箱、SMERFS、SRMP、SoRel、CASRE 和软件可靠性专家系统 SRES 等多种。

2 现有软件可靠性评估系统的局限性

现有软件可靠性评估系统都以软件可靠性模型库为核心,根据系统测试阶段得到的故障数据选择合适的软件可靠性模型并对软件可靠性进行评价和预测。软件可靠性模型至今已有近百个,它们基于的前提条件各不相同,假设的可靠性随机过程也变化多样。例如可靠性模型根据假设的故障总数可分为有限故障类和无限故障类;根据时间域可分为时钟时间和执行时间两类;假设故障发生时间的分布有指数分布、Weibull 分布等多种;模型使用的参数估计方法有最大似然法、最小二乘法法和贝叶斯估计等。

面对如此繁多而复杂的可靠性模型,一个可靠性评估系统很难把它们都包括进去。现有的可靠性评估系统中,其模型库最多也就包含十几个有代表性的模型。我们不能说对一个软件项目最适用的模型就一定包含在这些有代表性的模型中。随着可靠性建模的发展,不断有许多新的建模方法和模型提出,如用回归技术建立线形模型、用模糊方法进行建模以及神经网络模型等。如何把这些新模型(包括已有但未在模型库中提供的模型)及时有效地提供给用户使用,以扩大用户选择适用模型的范围是目前许多研究人员关心的问题。

现有的软件可靠性评估系统在模型选择的科学和

董 威 博士生,主要研究领域为软件可靠性、软件测试;毛新军 博士,主要研究领域为 Agent 理论、技术和应用,软件重用;陈 磊 硕士生,主要研究领域为软件工程;王 戟 博士,主要研究领域为软件形式化方法,软件可靠性,软件测试;齐治昌 教授,博士生导师,主要研究领域为软件工程。

合理性以及系统结构的灵活性等方面存在诸多局限性,具体表现在:

(1)软件结构不灵活。在现有可靠性评估系统中加入新的软件可靠性模型并不是一件简单的事情。由于模型选择和可靠性评估由系统集中控制,所以系统在扩充时需要较大的改动,这限制了系统的灵活性和可扩充性。实际情况是,用户根据自身的需求希望能方便地将各种新的所需模型加入到模型库中,使系统得到扩充以适应可靠性工程发展的需要。

(2)不能有效地支持软件可靠性模型的共享和重用。软件可靠性模型之间相对独立,因此软件可靠性评估系统适合在分布式环境中运行,每个模型可以充分利用自身的资源并且能并发自主地执行。在一个用户开发出新模型并加入到系统中后,应能通过网络环境供其他用户共享和重用。现有软件可靠性评估系统还不能做到这点。

(3)选择合适的可靠性模型对于用户是一件困难的事情。软件可靠性模型在应用时具有不一致性,不同模型评估同一个软件的可靠性时,所得结果会出现差异。尽管有的系统能通过分析模型预测的准确性来指导用户选择模型,但这些方法目前还有局限性,人们仍在寻找改进方法的途径。随着系统中模型的增多,即使在系统中用现有方法对所有模型的预测结果进行准确性分析和比较,也会使系统的控制越来越复杂,这些情况都对用户选择模型造成极大的困难。

软件可靠性的上述特点和现有软件可靠性评估系统的局限性要求开发出开放、灵活、软件可靠性评估系统。在适应软件可靠性模型不断增长的过程中,系统应能有效地重用和共享现有软件资源,并具有科学、合理的可靠性模型选择策略和算法以提高软件可靠性评估的准确性。

3 基于 Agent 的软件可靠性评估系统结构框架

Agent 是指驻留在某一环境下能持续、自主地发挥作用,满足驻留性、自主性、反应性、社会性等特征的计算实体^[2]。Agent 的这些特征非常适合于动态、开放系统的开发。基于 Agent 的软件可靠性评估系统的结构框架如图1所示。系统中的实体分为两种:软件可靠性评估 Agent (SREA) 和软件可靠性模型 Agent (SRMA)。系统采用分布式人工智能 (DAI) 中合同网的问题求解方式进行软件可靠性评估。合同网问题求解方式假设发散、分布的环境中有两种角色的 Agent: 管理者和承包者。在管理者发布一项任务后,承包者根据自己的能力和责任对任务进行评价,并向管理者投标;管理者对所有投标进行评估并选择一个承包者签定合

同,最后综合各个承包方执行该合同的结果得到问题的最终解^[2]。

在基于 Agent 的软件可靠性评估系统中, SREA 相当于管理者, SRMA 相当于承包者。SREA 把接收到的软件故障数据送给多个 SRMA 进行软件可靠性评估,并对返回结果进行分析,判断哪个模型最适用于这组故障数据以及当前软件可靠性是否满足可靠性需求指标。SRMA 对应于某个具体的软件可靠性模型,它根据接收到的故障数据进行参数估计,然后对软件的可靠性进行评估,并判断该模型对这组数据的预测是否客观有效。当一个新的 SRMA 要加入系统时,它向 SREA 发出注册请求,使得自身在系统中能被 SREA 认可。整个系统是开放的, SREA 和 SRMA 可能位于不同结点上,所在平台可以是异构的,所有的 SREA 和 SRMA 以及它们之间的交互和合作关系构成整个软件可靠性评估系统。

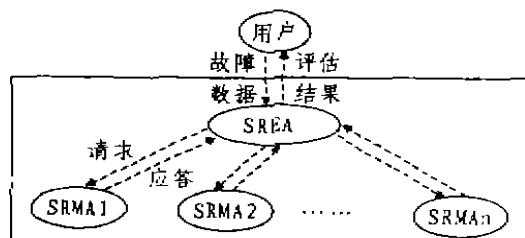


图1 基于 Agent 的软件可靠性评估系统结构框架

4 可靠性模型选择和评估过程

系统对软件进行可靠性评估的过程如下。

步1: 用户向环境中的 SREA 发送一组故障数据以及用户需求中确定的软件可靠性指标。

步2: SREA 根据自己所拥有的环境中 SRMA 的信息,向所有注册的 SRMA 发出软件可靠性评估服务请求,并把故障数据发送给这些 SRMA。

步3: 每个 SRMA 在接收到服务请求和故障数据后,先判断自己是否接受该服务请求,判断的依据包括:故障数据类型是否符合自己的要求(故障计数数据还是故障时间间隔数据),故障数据的时间域是否合适(基于时钟时间还是执行时间)以及故障数据本身的一些特性是否适合于该模型。若该模型不能或不适合于处理这组故障数据,就向 SREA 发一个应答,拒绝进行服务;否则就接受服务请求。

步4: SRMA 使用故障数据来估计模型参数,它能根据效果自动选择参数估计方法。在 SRMA 对软件可靠性进行评估之后,还应评价自己的预测是否客观准

确,这就需要检测预测和观测的失效行为之间系统而客观的差别。这里可以使用 u-结构图和 y-结构图等方法,它们用来判定预测是否平均地接近实际分布 $F_i(t)$, y-结构图还能检测到 u-结构图不能核查到的一些偏差。这样 SRMA 能在一定程度上辅助检测该模型是否合适,根据检测结果还可对预测进行重新校准,SRMA 最后把得到的可靠性评估结果和对自身的客观性评价一起返回给 SREA。

步5: SREA 在接收到从各个 SRMA 返回的结果后要要进行模型选择。通过对各模型预测的序列似然度比率(PLR)进行比较,能(至少渐进地)表明一般意义下一对预测中哪一个更精确, SREA 最终综合模型之间的 PLR 比较和模型对自身预测准确性的评价,从所有模型中选择一个对该组故障数据最合适的模型。

步6: SREA 把所选的模型对软件可靠性评估的结果和用户需求中确定的可靠性指标进行比较,判断软件可靠性是否已满足需求,以确定软件是应该继续进行系统测试还是可以提交使用, SREA 最终把所有结果返回给提交该故障数据的用户。 SREA 还要把最适用的模型名称和地址提供给用户,使用户能够积累相应的知识,在以后可以决定直接使用该模型。

可以看到该软件可靠性评估系统是一个 Multi-agent 系统,它自然地刻画了软件可靠性评估系统中数据、信息、资源和控制等的分布性以及问题求解的合作、协同、竞争等方式。从软件开发的角度看,它体现了一种“分而治之”的开发策略,把一个复杂、难以把握的软件可靠性评估问题的求解策略分解为一组易于处理的子策略。这些子策略交给一组自主的 SRMA 和 SREA 加以实施,它们之间通过交互来达到对整个问题的求解。

5 基于 Web 的系统实现

在当前 Internet 迅速发展且广泛使用的情况下,如果能把上述基于 Agent 的软件可靠性评估系统在 Internet 环境中实现,使用户可以通过 Web 浏览器来完成软件可靠性评估,将会大大拓宽软件可靠性评估系统的使用范围,从而推动软件可靠性工程的应用和发展。

基于 Agent 的软件可靠性评估系统基于 Web 的实现结构如图2所示。它可以分为三层:客户层、中间层和模型层。中间层和模型层的服务器运行使用 CGI 编写的 Agent 应用程序, SREA 驻留在中间层的可靠性服务器上,而每个 SRMA 驻留在模型层中的模型服务器上,各层的 Agent 应用程序之间以及和其它应用程序之间要进行交互和通信。

(1) 客户层和中间层的交互 客户层和中间层通

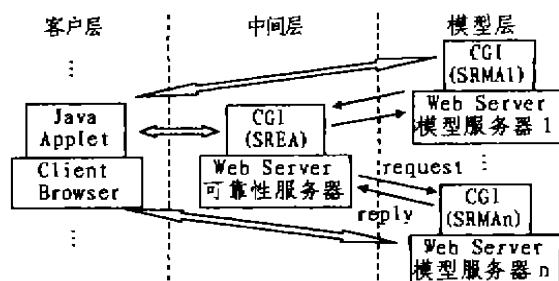


图2 系统在 Internet 中基于 Web 的实现结构

过 Client/Server 模式工作。用户在客户机上通过 Web 浏览器访问中间层的可靠性服务器。当使用可靠性服务器提供的软件可靠性评估服务时,调用服务器的标准 CGI 接口,把客户机中提供的故障数据发送给可靠性服务器中的 SREA。在可靠性服务器把评估结果返回到客户机后,要把原始数据和结果曲线绘制在客户机上。这里利用 Java Applet 的平台无关性以及客户端运行的高效性来实现在客户机中的图形绘制。

(2) 中间层和模型层的通信 中间层和模型层之间属于 Internet 上 Agent 应用程序之间的通信,可以采用 KQML(知识查询与操纵语言)作为通信语言。KQML 在 Agent 应用程序之间不仅能传递要表达的信息,还能表示信息的性质,如是一种请求或一种服务功能的公告等,这些都是通过通信原语来完成的。KQML 提供的通信操作原语中有很多用于 Agent 间进行信息交换的协议,能够较容易地实现基于合同网的通信协议, SREA 应用程序把请求和数据发送给每个模型服务器, SRMA 在计算完成后将结果数据一次返回给 SREA,也包括可能拒绝服务的应答。

模型服务器还可以向可靠性服务器发送两种请求:注册请求和注销请求。注册请求把 SRMA 的名称和服务器地址发送给可靠性服务器,放在注册表里。只有在注册后,可靠性服务器才能与该模型服务器进行通信。在 SREA 对所有 SRMA 返回的结果作出选择和评价后,把所选择的 SRMA 的名称和服务器地址随同结果一起返回给用户。

(3) 客户层与模型层的直接交互 这种情况只有当用户拥有可靠性评估的经验信息之后才可能进行。用户根据以前使用系统的结果,知道哪种模型可能对该类故障数据最有效,于是使用以前由中间层返回的该 SRMA 的服务器地址,直接与其进行交互。它们之间的工作模式和协议类似于客户层与中间层之间所使用的,不同之处在于客户机直接把数据文件送给模型服务器,而 SRMA 在计算完后把结果直接返回到客户

机并绘制出来。

结论 与传统系统相比,基于 Agent 的软件可靠性评估系统具有以下特点:

(1)可靠性模型选择和决策的科学性和合理性。系统为故障数据选择合适的可靠性模型是“二阶段决策”过程:第一阶段是 SRMA 在接收到故障数据时根据数据外部特性来判断模型是否可处理该类数据,在模型运行后根据预测和观测失效行为之间的偏差来客观评价自身的预测准确性;第二阶段是 SREA 在收到所有模型返回的结果后对各模型预测的精确性进行比较,判断哪个模型更适用。SREA 综合以上两个阶段的结果选择出最适用的模型。该决策过程是由请求方和服务方合作共同完成的,比起单方面决策更科学和合理。

(2)系统结构具有较高的灵活性。系统中的 SREA 和 SRMA 能够对其内部状态和行为实施完全自主的控制,具有高度自治性和独立性。从软件开发的角上看,这种软件系统的模块化、可维护性、可重用性也就更好。在要增加新模型时,只需开发一个新的 SRMA 加入到系统环境中并进行注册,无须对原有系统进行改动,而且能充分重用已有模型。

(3)系统适合在分布式环境中应用,实现了并发和共享。由于 SRMA 具有驻留性、反应性以及独立性等特征,所以它们的行为可以并发地运行。从软件结构上看,它是一个多线程的系统。在网络和分布式环境中,各 SRMA 可以驻留在不同的主机上,能够充分地利用

自身资源,多个用户还可以对它们共享使用。

(4)系统通过问题分解简化了求解过程。系统利用了分布式人工智能中合同网的求解方式,把为故障数据选择合适模型并进行可靠性评估的问题分解到 SREA 和 SRMA 中来实现,降低了求解的复杂性。整个系统通过实体之间交互来实现合作、协同和竞争,从而完成整个问题的求解。

总体说来,基于 Agent 的软件可靠性评估系统与传统软件可靠性评估系统相比具有很多优越性,与当前迅速发展的网络与分布式环境结合起来,可以对软件可靠性工程的应用和发展起到很大的促进作用。

参考文献

- 1 Lyu M R Handbook of Software Reliability Engineering. IEEE Computer Society Presss, 1996
- 2 Wooldridge M, Jennings N R. Intelligent Agents: Theory and Practice. Knowledge Engineering Review, 1995, 10 (2): 115~132
- 3 Nwana H, Lee L. Coordination in Software Agent Systems. BT Technology Journal, 1996, 14(4)
- 4 Musa J D et al. Software Reliability Measurement, Prediction, Application. McGraw-Hill Book Company, 1987
- 5 Prieto-Diaz Status Report: Software Reusability. IEEE Software, 1993, 10(3): 61~66
- 6 徐仁佐,郑仁杰,等.软件可靠性模型及应用.清华大学出版社,广西科技出版社, 1994
- 7 Jacobs R A, et al. Task decomposition through competition in a modular connectionist Architecture, the what and where vision tasks. Cognitive Science, 1991(15): 219~250
- 8 Mandelbrot B B. The fractal geometry of nature. Freeman, San Francisco, 1982
- 9 Murre J M J. Categorization and learning in neural networks. Modelling and implementation in a modular framework. Dissertation, Leiden University, 1992
- 10 Posner M I, et al. Localization of cognitive operations in the human brain. Science, 1988(240): 1627~1631
- 11 Prasinkiewicz P, Hanan J. Lindenmayer systems, fractals and plants. Springer-Verlag, New York, 1989
- 12 Whitley D, Starkweather T. Genitor II. A distributed genetic algorithm. Journal of Experimental and Theoretical Artificial Intelligence, 1990
- 13 Solis S A. Learning and generalization in layered neural networks: the contiguity problem. In: Personnas L, Dreyfus G, eds. Neural networks, from models to applications. I. D. S. E. T, Paris, 1989: 168~177
- 12 王洪燕,杨敬安.并行遗传算法研究进展.计算机科学, 1999(6)
- 13 王洪燕,杨敬安.并行进化 BP 神经网络.合肥工业大学学报, 1993(3)
- 14 王洪燕,杨敬安.关于主从式分层并行遗传算法的研究.见:第五届全国计算机应用联合学术会议论文集, 1999
- 1 Rumelhart D E, et al. Learning internal representations by error propagation. Parallel Distributed Processing: Explorations in Microstructures of Cognition, 1986, 1(1): 318~362
- 2 Tanese R. Distributed genetic algorithms. In: Schaffer J D, ed. Proc. of the Third Interl. Conf. on Genetic Algorithms. San Mateo, CA: Morgan Kaufmann, 1989: 434~439
- 3 Dodd N. Optimization of network structure using genetic algorithms. In: Widrow B, Angeniol B, eds. Proc. of the Interl. Neural Network Conf., INNC-90-Pairs. Kluwer, Dordrecht, 1990: 693~696
- 4 Hank G, et al. The gambler's ruin problem, genetic algorithms, and the sizing of populations. In: Back T, ed. Proc. of the Fourth Interl. Conf. on Evolutionary Computation. New York: IEEE Press, 1997: 7~12

(上接第45页)

搜索空间的维数;2)解的模块化复制策略,以及对有效的规则的重用,降低了计算的复杂度;3)采用由简单到复杂的设计方法明显优于直接对复杂问题进行优化;4)采用并行处理机制和问题分解处理方法来直接提高算法收敛的速度。

参考文献