

需求演化排队模型研究

严玉清^{1,2} 李师贤¹ 孙为军^{1,3} 黄昌勤⁴

(中山大学计算机科学系 广州 510275)¹ (广东外语外贸大学思科信息学院 广州 510006)²
(广东工业大学计算机学院 广州 510006)³ (华南师范大学教育信息技术学院 广州 510631)⁴

摘要 用定量手段研究需求演化需要相关方法加以指导。利用排队论来分析需求变化请求从提出到实现的整个过程所具有的排队模型特征。在 $M/M/1/m/m$ 假设之下,改进了现有需求成熟度的计算方法,并通过实例计算了该排队模型的特征指标,分析了这些数量对评价需求分析的效率及需求工程师的能力所具有的借鉴意义,并指出排队论在指导用定量手段研究需求变化管理上的作用。

关键词 需求演化,需求规约,排队论

中图分类号 TP311.5 **文献标识码** A

Study of Requirements Evolution Based on Queuing Theory

YAN Yu-qing^{1,2} LI Shi-xian¹ SUN Wei-jun^{1,3} HUANG Chang-qin⁴

(Department of Computer Science, Sun Yat-sen University, Guangzhou 510275, China)¹

(Cisco School of Informatics, Guangdong University of Foreign Studies, Guangzhou 510006, China)²

(Faculty of Computer, Guangdong University of Technology, Guangzhou 510006, China)³

(College of Educational Information Technology, South China Normal University, Guangzhou 510631, China)⁴

Abstract It needs methodology to study requirements evolution by quantitative methods. Based on queuing theory, this paper analyzed the process of requirements change requests and pointed out the characteristics of queuing models. The current method of calculating requirements maturity index was improved by $M/M/1/m/m$. An example shows that the quantities got from this queuing model have their meanings to evaluate the effectiveness of requirements analysis and the capability of requirements engineers. The function of queuing theory in guiding the management of requirements change was pointed out.

Keywords Requirements evolution, Requirements specification, Queuing theory

1 引言

软件通常存在多种不同的版本。利用敏捷开发方法开发的软件以及由市场驱动开发的软件,产品机构可以有同一个软件产品在不同时间演化阶段上的多个版本,也可以针对不同用户和市场开发的不同版本。例如,美国 SAS 公司开发的统计软件有针对统计教学用的版本,也有针对不同规模企业用的企业级版本,还有针对使用不同自然语言的用户版本,以便在不同的国家销售。这些不同的版本有的正在被使用,有的正在被开发,有的仍然在定义。这些情形都需要对需求规约进行版本计划^[1],以确定需求的优先级和开发的优先顺序,以便以最快的速度、最新颖的功能抢占市场份额,并根据市场和用户需求的变化,继续对版本进行更新。这个过程是一个迭代式的增量开发过程^[2,3],如图 1 所示。这种开发是避免需求风险的一种好的做法^[4]。在图 1 中,需求管理包括多项活动:需求变化管理、需求缺陷管理、需求跟踪管理、需求一致性管理等。需求变化管理是需求管理中最重要内容。需求变化包括给原来版本增加新的需求,以及修改原来版本的

需求。需求变化过程遵循着相同的需求工程原则和方法,但涉众可能会发生变化。这时,新的涉众可以查阅已有的需求文档和需求规约去获得对项目的理解,并对项目产生新的构思。需求版本的不断变化形成了一个需求版本演化流。有两种类型的演化流:一个是有限流,即软件最终终止在某一个需求规约上不再发生变化(软件终结版);另一个是无限流,软件不会终止在某个需求规约上,它总会根据市场的需求而适时地升级。这个过程反映了需求和软件变化的历史。记录和跟踪这个过程,可以研究项目需求及软件变化的特点、需求及软件变化的趋势,也可以与其它相关软件进行比较,以获得一些有益的结论,从而充实软件文化在理论和实践中的内容,是软件文化及产业的一个历史记录和缩影。

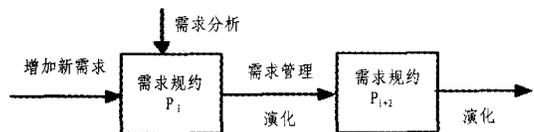


图 1 需求规约演化图

到稿日期:2011-06-09 返修日期:2011-09-06 本文受国家自然科学基金(60940033),广东省自然科学基金(10151063101000046)资助。

严玉清(1963—),女,博士生,副教授,主要研究方向为软件工程、软件需求管理、运筹学, E-mail: yyqelsa@yahoo.com.cn;李师贤(1944—),男,教授,博士生导师,主要研究方向为软件工程、形式语义学;孙为军(1975—),男,博士生,讲师,主要研究方向为软件工程等;黄昌勤(1972—),男,博士后,教授,主要研究方向为服务计算、基于语义 Web 的智能技术。

对需求演化过程的研究可分为定性研究与定量研究。在定量研究方面,又可从多个角度进行。例如:1)从经济学角度,可分为数量经济学和社会经济学^[5];从数量经济学角度,不研究人对过程的影响,只从纯数学角度去研究需求变化的数量、需求变化数量与需求成熟度的关系、需求变化数量所服从的分布及特点^[2,6]、需求变化的传播范围^[7-9]及对成本^[10]和进度的影响等;从社会经济学角度,要把人——开发涉众作为影响过程的因素加以考虑,同时要考虑过程对人、对社会的影响;2)从时间演化角度,可以研究需求及软件如何随着版本和时间的推移在数量上和特征上发生变化的特点和规律;3)从管理与决策角度,利用经验研究方法定量地研究需求变化和演化的过程及实施需求变化的步骤以及所产生的风险^[4,11]等。还可以把这些不同的角度结合起来研究,以便更加全面地了解和理解需求演化的特点和规律,正确评估需求变化带来的正面和负面的影响。

下面利用排队论对需求演化过程进行分析和建模。结果表明, $M/M/1/m/m$ 模型一方面改进了文献^[2,6]的计算需求成熟度的方法,另一方面它所提供的数量能够用来评估需求分析的效率、需求管理的效率以及工程师的能力。

本文第2节回顾当前用定量方法研究需求变化和演化的一些观点、方法;第3节概述排队论,包括排队模型的组成、与排队模型有关的主要数量指标及排队论的研究内容;第4节,先分析需求版本演化过程所具有的排队模型特征,然后利用文献^[2,6]所提供的22个不同软件版本的总需求和变化需求的数量,基于排队论方法建立需求演化排队模型,定义需求变化概率并指出其所揭示的事实;最后总结本文研究,并指出进一步的研究方向。

2 相关研究

需求度量属于软件度量范畴。文献^[11]认为软件度量方法应该尽可能简单易用,并利用了贝叶斯网络的因果关系模型、经验软件工程和多准则决策支持等方法来探讨如何对软件开发各项活动进行有效度量、管理决策支持和软件风险评估。文献^[12]为了实现CMM的需求管理关键过程域的目标,基于GQM范式(Goal Question Metrics paradigm)提出了度量需求的稳定性和易变性的38项软件测量指标,为软件开发公司实施CMM提供了实际的指导。文献^[13]对需求的质量提出了其它的度量方法。在对需求的质量进行度量时,需求的稳定性和易变性是最重要的度量内容^[2,6,14],也是用定量方法研究需求演化特点的主要对象。文献^[2,6]基于经验方法,利用定量的手段,在一个实际案例中收集了22个不同软件版本的需求和变化需求的总数量,并对某些具体的功能需求在各个不同版本中的变化情况进行实际的观察和数据收集,从反映需求的稳定性和易变性角度提出了许多概念和计算公式,如需求成熟度(RMI)、历史需求成熟度(HRMI)、需求稳定性(RSD)、需求变化平均数(ARC)等;利用数据直方图和趋势图来研究8种不同功能需求的演化特点及趋势,并发现功能需求之间的依赖关系会影响需求变化,会对变化的范围产生影响。作者把需求成熟度(RMI)定义为刻画需求就绪(readiness of requirements)程度的量: $RMI = \frac{R_T - R_C}{R_T}$,其中 R_T 为某一版本中需求的总数量, R_C 为这一版本中变化需求

的数量。这个数值反映了每个需求的一种平均变化情况。在实际开发中,由于数据收集存在不完整性,因此要准确统计每个版本的总需求数量和总的变化需求数量,从而准确计算RMI并估计RMI的分布特点,不是一件容易的事。但如果事先有意识地收集需求变化的数据,并利用一些统计手段对所有版本的需求成熟度研究其分布特点,然后把这种分布特征推广到其它类似的项目中去指导其它系统的开发决策,是有现实意义的。

3 排队论概述

3.1 排队系统的组成及特征

排队论(Queueing Theory)是研究拥挤现象或大量服务过程的一门数学分支^[15-18],它主要通过研究各种服务系统在排队等待中的概率特点,来解决系统的最优设计和最优控制问题。常见的排队现象如人们去商场购物付款、故障车床的维修、计算机中CPU处理各种任务等。这些现象的共同特征是存在着“顾客”(例如人们、车床、任务)与“服务台”(例如收款机、银行柜台、维修点、CPU)之间的一种服务关系,如图2所示。

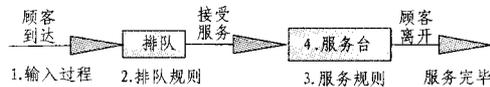


图2 排队模型流程图

图2说明了排队系统由4个基本部分组成:(1)输入过程。描述顾客来源及顾客到达系统的方式,例如顾客总体是有限还是无限、顾客到达系统是单个还是成批等。(2)排队规则。是指顾客在等候服务的过程中对排队的态度或应遵循的规则,例如损失制、等待制和混合制。(3)服务规则。是指顾客等待接受服务所拥有的权限,例如先到先服务、随机服务、有优先权服务等。(4)服务台的特征和队列类型。包括服务台数量、队列类型、服务时间的随机性。

描述排队系统特征的主要数量指标有:顾客相继到达的时间间隔的分布和服务时间的分布(这两种分布需要实测的数据来确定)、等候队长与系统队长、等候时间与逗留时间、到达率、服务率、服务强度、忙期等。要研究与计算这些指标,首先要计算系统状态概率(系统中顾客数的平均值)。

3.2 排队论的研究内容

排队论的研究内容主要有3部分^[18]:

(1)研究各种排队系统的概率规律性,主要是研究等待时间分布、队长分布等,包括瞬态和稳态两种。

(2)研究排队系统的最优化——静态最优(设计最优)和动态最优(运营最优)。

(3)研究排队系统的统计推断,即判断一个给定的排队系统符合于哪种模型,以便根据排队理论进行分析和研究。

4 需求演化排队模型的特征

把图1的需求规约演化的过程更加具体一些,如图3所示。下面用排队论知识结合图2来分析图3具有的排队模型特征。在图3中,对于第*i*个过程 $P_i, i=1,2,3,\dots$,需求工程师收集到需求变化请求集 pR_i ,可以把它看成为进入服务系统的顾客; cR_i 为变化请求累加集,是正在排队等候服务的顾客,即正排队等候变化实现的需求;实施变化的工程师为服务

台;正式接受变化的需求集 fR_i 为正在接受服务的顾客,当服务完毕之后,它们离开服务队列,加入到需求规约 RS_i 中。这个过程具有图 2 所示的服务系统的特点。在形成最终的需求规约之前,需求工程师接收到多个需求变化请求集,然后对其进行进行处理。这个排队模型尽管对顾客源(变化需求的数量)及系统容量(能够容纳的最大变化需求数量)没有限制,但对某一个软件的开发来说,系统容量和顾客源本质上是有限的。这个排队过程看成为:(1) $M^k/M/1/m/m$:变化请求集的到达和服务时间都服从指数分布;(2) $G^k/G/1/m/m$:变化请求集的到达和服务时间都服从一般分布。

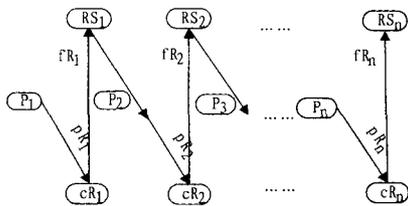


图 3 需求变化的过程性

还可以根据实际情况把这个排队模型看成其它类型的模型,如 $M^k/G/1/m/m, M^k/GI/1/m/m$ 等。此处,对于需求变化的实现,通常是按优先权服务的。在每次服务的数量上,由于存在需求依赖关系^[19],使得每次提出的变化请求通常不会只有一个需求,而是多个需求。而且由于对需求变化的工作规范要求,使得需求变化的实现是成批的,而不是单个实现的。此处, $M^k/M/1/m/m$ 和 $G^k/G/1/m/m$ 与一般排队论中机器因故障停机待修的问题有相似的地方,也有不同的地方。需求变化请求集可以来自总体(原需求规约)中,也可是新增的需求而与原来的总体无关。但不论如何,变化实现之后,所有需求都要回到总体中去,并且还有可能重新成为变化请求集的对象。

另外,还可以根据实际情况把服务台看成为 3 个串联服务台——提出需求变化请求的工程师、对变化请求进行检查和发布实施命令的变化控制委员及实施变化的需求工程师,它们共同合作来完成需求变化的过程。

以上的模型都假定服务台不会发生故障,但实际情况是排队模型的服务台有可能会发生故障。有故障的需求演化排队模型是指实现需求变化和修正需求缺陷的工程师中途离开这个开发团队而导致服务的终止。对于此种情形,可进一步研究可修的需求演化排队模型^[18]或有差错的需求演化排队模型^[15]的特点。

下面在 $M/M/1/m/m$ 的假设之下来研究需求演化服务模型的特征。

4.1 需求演化排队模型的特征研究

$M/M/1/m/m$,即每次只有一个顾客到达,顾客相继到达系统的时间间隔和服务时间皆服从负指数分布;系统内有 1 个服务台,顾客源为 m ,系统容量为 m 。为了计算此排队模型所反映的特征指标量,需要收集一些数据^[15,18]:单位时间需求变化请求的平均数,用 λ 表示;单位时间实现变化的平均数,用 μ 表示;需求总数 m 。利用这些已知量,可进行如下计算^[15]:

服务强度

$$\rho = \frac{\lambda}{\mu} \quad (1)$$

服务设施闲置率

$$p_0 = \left[\sum_{k=0}^m \frac{m!}{(m-k)!} \rho^k \right]^{-1} \quad (2)$$

利用率

$$\xi = 1 - p_0 \quad (3)$$

系统内顾客数的均值

$$L_s = m - \frac{1 - p_0}{\rho} \quad (4)$$

等候线上的顾客数均值

$$L_q = L_s - 1 + p_0 \quad (5)$$

系统内顾客花费的时间均值

$$W_s = \frac{L_s}{\mu(1 - p_0)} \quad (6)$$

顾客在等候线上的时间均值

$$W_q = \frac{L_q}{\mu(1 - p_0)} \quad (7)$$

每条需求的成熟度

$$\eta = \frac{m - L_s}{m} \quad (8)$$

需求工程师的工作效率

$$\xi = 1 - p_0 \quad (9)$$

如果知道等候的总时间 T_w ,则可估算变化请求的总数为 $cR = \lambda T_w$ 。如果知道实现变化的服务总时间 T_s ,则可计算正式接受变化的需求总数为 $fR = \mu T_s$ 。还没有接受变化实现的需求数量为 $cR - fR = \lambda T_w - \mu T_s$,这时需求成熟度的计算公式为(Anderson Stuart 和 Massimo Felici 的计算方法^[2,6]):

$$\eta = \frac{m - fR - cR}{m}, 0 \leq \eta \leq 1 \quad (10)$$

显然,在实际运用中,利用式(8)比运用式(10)更加容易度量需求的成熟度。这是因为在实际的开发过程中,实际的数据 fR, cR 不容易收集到,而通过之前相似项目的数据收集进行相关的统计计算,然后把这些统计出的结论类比过来,是一种好的实践方法。也就是说,在度量某个项目的需求成熟度时,不需要实际测量每次对项目的变化请求数量,也不需要记录每次对实现的需求变化数量,可事先有意识地收集相关变化需求的数据,利用排队论提供的方法来分析实际项目开发的特点,获得平均值 λ, μ ,并估算项目的规模获得需求总数 m ,从而达到方便估算需求成熟度的目的。

这里,需要强调量化需求成熟度的好处:它是量化需求的可接受性的重要指标。需求的可接受性是指需求需要满足这些特征^[20,21]:完整性、正确性、可行性、必要性、可测试性、无冗余性、无二义性、可跟踪性等。需求的成熟度越高(η 的值接近 1),说明该需求过程中变化需求的数量越小;反之,需求的成熟度越低(η 的值接近 0),说明该需求过程中变化需求的数量越大。因此,需求成熟度起到一个预警作用,可以作为需求工程师的需求开发能力高低的一个评判指标,也可作为需求工程师开发需求的效率的一个评价指标(需求工程师的工作效率也可用式(9)来度量。实际中,可综合这两个量来评价需求工程师的实际需求开发效率),或作为该项目是否容易开发程度的度量。因此,需求成熟度是量化开发组织需求分析效率的一个指标,是需求工程师工作效率的反映,也是项目复

杂度或规模的一个间接度量。

4.2 实例分析

从文献[2,6]中分别提取 22 个软件版本的每一版本的需求变化总数目和需求总数目,得表 1 和表 2 中的数据。

表 1 22 个软件版本需求变化单位数

版本号	1	2	3	4	5	6	7	8	9	10	11
需求变化单位数	0.65	0.2	0.6	0.5	0.475	1.2	0.4	0.19	1.1	0.88	0.1
版本号	12	13	14	15	16	17	18	19	20	21	22
需求变化单位数	3.65	0.75	1.4	0.18	0.39	0.02	6.5	0.85	2.15	2.45	1.15

表 2 22 个软件版本需求总数单位

版本号	1	2	3	4	5	6	7	8	9	10	11
需求总数单位	3.55	3.6	3.65	3.65	3.8	4	4.1	4.2	4.4	4.6	4.7
版本号	12	13	14	15	16	17	18	19	20	21	22
需求总数单位	5.6	5.7	6.1	6.15	6.2	6.25	7.9	8.1	8.4	8.6	8.7

收集与需求变化有关的数据,是一件困难的事情,特别是收集每一个软件版本的每个需求基线及变化情况的数据、需要方法的指导。本文为了说明排队论在需求变化的定量研究中的作用和意义,利用文献[2,6]所提供的数据作为本例中的部分数据,在适当的假设之下计算需求演化服务模型的特征量。由表 1,可计算出平均需求变化单位数 $c \approx 1$,由表 2 可得平均需求总数量单位 $m \approx 5$,则变化请求的到达率近似为 $\lambda =$

$\frac{c}{m} \approx 0.2$ 。假设服务率为 $\mu = 0.4$,则可计算:

$$\text{服务强度: } \rho = \frac{\lambda}{\mu} = \frac{0.2}{0.4} = 0.5$$

需求工程师的闲置率为:

$$p_0 = \left[\sum_{k=0}^5 \frac{5!}{(5-k)!} \rho^k \right]^{-1}$$

表 3 需求变化的概率

版本号	1	2	3	4	5	6	7	8	9	10	11
需求变化的概率	0.183	0.056	0.164	0.137	0.125	0.3	0.098	0.045	0.25	0.191	0.021
版本号	12	13	14	15	16	17	18	19	20	21	22
需求变化的概率	0.652	0.132	0.23	0.029	0.063	0.003	0.823	0.105	0.256	0.285	0.132

结束语 本文利用排队论首先分析了某一软件版本规约中需求变化请求的提出及实现过程具有的排队模型特征。对于 $M/M/1/m/m$,结合例子计算了这种需求演化排队模型的一些主要性能指标,并分析这些数量指标在改进现有计算需求成熟度的方法、评估需求分析的效率和工程师的能力等方面的作用。

我们利用排队论研究需求变化管理的目的是:(1)一方面,把变化的请求和实现过程看成为一个服务系统,这有助于了解需求变化请求出现的情况以及变化实现的效率;(2)另一方面,根据进度和成本的要求,可对需求变化管理工作的流程依据排队模型的特征提出一些设计要求,以配备能够达到此效率的工程师和相应的工具。

利用排队论来研究需求变化过程的好处:(1)指导数据收集。通常,在需求开发过程中,工程师没有特意地去收集与需求变化相关的数据以供将来研究,也不知道怎样去收集数据。利用排队论,在数据收集方面可获得有益的指导。(2)提供数量化计算工具。排队论提供了思考和解决问题的方法和公式,可以用量化手段有效地研究需求变化过程。

$$= (1 + 5\rho + 20\rho^2 + 60\rho^3 + 120\rho^4 + 120\rho^5)^{-1}$$

$$= \frac{1}{25} = 0.04$$

服务设施的利用率,即需求工程师的工作效率为:

$$1 - p_0 = 0.96$$

系统内等待实现的变化需求数的均值为:

$$L_s = m - \frac{1 - p_0}{\rho} = 3.08$$

等候线上的变化需求数的均值为:

$$L_q = L_s - 1 + p_0 = 2.12$$

系统内顾客花费的时间均值为:

$$W_s = \frac{L_s}{\mu(1 - p_0)} = 0.8$$

顾客在等候线上的时间均值为:

$$W_q = \frac{L_q}{\mu(1 - p_0)} = 0.66$$

每条需求的成熟度为:

$$\eta = \frac{m - L_s}{m} = 0.39$$

由以上分析与计算可见,利用排队论作为方法与手段在定量研究需求演化的特点方面是有效的,也是值得继续探讨和研究的。

4.3 需求变化的概率

假如所有需求变化请求百分百被实现,这样由表 1 和表 2,可以计算出需求演化流在各个版本中出现的概率(需求变化概率),如表 3 所列。需求变化概率可作为需求发生变化的可能性,这种可能性的大小同样也体现了需求分析的效率。概率低,效率就高,反之则效率低。需求变化概率与需求演化流及规约版本流的终止条件是否有关,值得进一步探讨。

我们进一步要研究其它情形的需求演化排队模型,如 $M^k/M/1/m/m$, $M^k/M/1/\infty/\infty$, $M^k/M/c/\infty/\infty$ 等。要根据实际的需求变化管理过程,在排队论的指导下进行适当的推理。

参考文献

- [1] Carlshamre P, Sandahl K, Lindvall M, et al. An Industrial Survey of Requirements Interdependencies in Software Product Release Planning[C]//Fifth International Symposium on Requirements Engineering. Toronto, Canada, 2001:84-91
- [2] Ricerche C N D, Felici M. Requirements evolution understanding formally software engineering processes within industrial contexts[R]. Bando: Italian National Research Council(CNR), 2002
- [3] D'Avila Garcez A S, Russo A, Nuseibeh B, et al. An analysis-revision cycle to evolve requirements specifications[C]//The 16th Annual International Conference on Automated Software Engineering. Washington, DC, USA: IEEE Computer Society, 2001: 354-358

(下转第 123 页)

合构件提供服务目标所需服务。根据 3.2 节中服务目标对 AccountManager 和 Wage 构件建立的复合构件 AccountWage 行为视图如图 3 所示。

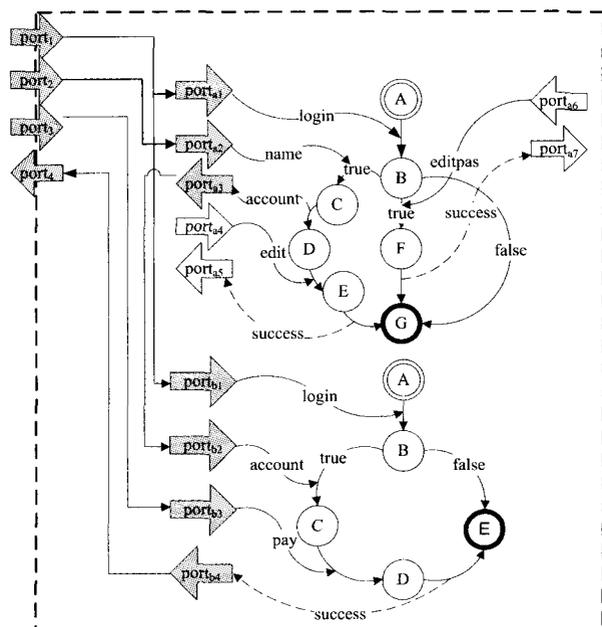


图 3 复合构件 AccountWage 行为视图

结束语 本文通过研究构件行为,介绍了行为的 π 演算进程描述,并以此建立行为片段的 π 关系推导,实现对行为的片段分解。提供了一种基于逻辑推理的构件行为片段提取与重组算法,根据服务目标所提供的输入、输出作为待求解,从候选构件中挖掘能满足目标需求的行为片段,通过行为片段重组,满足服务目标需求,最后通过实例证明了本方法的有效性。

参考文献

(上接第 109 页)

[4] Chuk Y. A Quantitative Methodology for Software Risk Control [C]// IEEE International Conference on Systems Management and Cybernetics. San Antonio, TX, 1994: 2015-2020

[5] [美] Boehm B W. 软件工程经济学[M]. 李师贤,等译. 北京:机械工业出版社,2004

[6] Stuart A, Massimo F. Quantitative Aspects of Requirements Evolution[C]// The 26th Annual International Computer Software and Application Conference. Oxford, England, 2002: 27-32

[7] Yan Yu-qing, Li Shi-xian, Liu Xian-ming. Quantitative Analysis for Requirements Evolution's Ripple-Effect [C]// Bangkok, Thailand: IEEE Computer Society, Los Alamitos, CA, US, 2009: 423-427

[8] Li Yin, Li Juan, Yang Ye, et al. Requirement-Centric Traceability for Change Impact Analysis: A Case Study [C]// Making Globally Distributed Software Development a Success Story. Berlin / Heidelberg: Springer, 2008: 100-111

[9] Yan Yu-qing, Li Shi-xian, Sun Wei-jun. Dependency based technique for identifying the ripple effect of requirements evolution [J]. Being accepted by Journal of Software

[10] Lavazza L, Valletto G. Requirements-based Estimations of Change Costs [J]. An International Journal of Empirical Software Engineering, 2000, 5(3): 229-243

[11] Fenton N E, Neil M. Software Metrics: Roadmap [C]// ICSE'00 Proceedings of the Conference on The Future of Software Engi-

[1] 任洪敏,钱乐秋. 构件组装及形式化推导研究[J]. 软件学报, 2003, 14(6): 1066-1074

[2] Sadaoui S. Composition of structured process specifications [J]. Electronic Notes in Theoretical Computer Science, 2003, 82(5): 132-143

[3] Allen R, Garland D. A formal basis for architectural connection [J]. ACM Transactions on Software Engineering and Methodology, 1997, 6(3): 213-249

[4] 梅宏,陈锋,冯耀东,等. ABC: 基于体系结构、面向构件的软件开发方法[J]. 软件学报, 2003, 14(4): 721-732

[5] 胡军,于笑丰,张岩,等. 基于场景规约的构件式系统设计分析与验证[J]. 计算机学报, 2006, 29(4): 513-525

[6] 李良明,王志坚,唐龙业. 构件功能行为测试的研究[J]. 小型微型计算机系统, 2010, 31(4): 686-690

[7] Hameurlain N. A Formal Framework for Component Protocols Behavioral Compatibility [C]// Proceedings of the 13th Asia Pacific Software Engineering Conference. Bangalore: IEEE Computer Society, 2006: 87-94

[8] 罗毅,李兴宇,关连伟. 构件演化中的系统行为一致性的研究 [J]. 计算机科学, 2008, 35(1): 266-270

[9] Milner R, Parrow J, Walker D. A calculus of mobile processes [J]. Information and Computation, 1992, 100(1): 1-40

[10] Sangiorgi D. Expressing mobility in process algebras: First-order and higher-order paradigms [D]. Edinburgh: University of Edinburgh, 1992

[11] 李长云,李贇生,何频捷. 一种形式化的动态体系结构描述语言 [J]. 软件学报, 2006, 15(6): 1349-1359

[12] 张弛. 基于行为描述的软件构件组合兼容性检查 [J]. 计算机工程, 2010, 36(12): 46-51

[13] 肖刚,王培君,陆佳炜,等. 基于逻辑推理的构件组装策略及其算法 [J]. 信息与控制, 2009, 38(6): 759-764

neering, 2000: 357-370

[12] Loconsole A. Measuring the Requirements Management Key Process Area [C]// Proceedings of ESCOM-European Software Control and Metrics Conference. London, UK, April 2001

[13] Hammer T F, Huffman L L, Rosenberg L H. Doing requirements right the first time [J]. The Journal of Defence Software Engineering, 1998: 20-25

[14] Loconsole A. ICSE'04 Proceedings of the 26th International Conference on Software Engineering [C]// 2004: 42-44

[15] 陆传贵. 排队论(第 2 版) [M]. 北京: 北京邮电大学出版社, 2009

[16] 陆凤山. 排队论及其应用 [M]. 长沙: 湖南科学技术出版社, 1984

[17] [美] 华兴. 排队论与随机服务系统 [M]. 上海: 上海翻译出版社, 1987

[18] 唐应辉, 唐小我. 排队论——基础与分析技术 [M]. 北京: 科学出版社, 2006

[19] Dahlstedt Å G, Persson A. Requirements Interdependencies-Moulding the State of Research into a Research Agenda [C]// The Ninth International Workshop on Requirements Engineering: Foundation for Software Quality (REFSQ 2003), held in conjunction with CAiSE, 2003: 71-80

[20] [美] Wieggers K E. 软件需求(第 2 版) [M]. 刘伟琴, 刘洪涛, 译. 北京: 清华大学出版社, 2004

[21] [美] Kovitz B L. 实用软件需求(第 1 版) [M]. 胡辉良, 张翌, 译. 北京: 机械工业出版社, 2005