

面向全局约束的 QoS 分解与服务状态监控机制研究

刘志忠, 彭辉, 曹雷

(解放军理工大学指挥信息系统学院, 江苏南京 210007)

摘要: 组合服务 QoS 需求的全局性和服务 QoS 状态监控的局部性之间的矛盾使得组合服务的可用性难以满足用户的需求. 针对此问题, 提出了组合服务的自适应框架, 重点研究了全局 QoS 分配与服务的 QoS 状态监控机制; 通过分析服务的 QoS 历史信息来预测服务的 QoS 阈值, 从而为全局 QoS 分解提供启发信息. 并引入松弛系数以降低服务自适应的开销. 实验表明: 通过 QoS 阈值预测和松弛系数的使用能够实现以较小的自适应开销适应环境变化满足用户端到端的 QoS 需求, 从而提高组合服务的可用性.

关键词: QoS; 全局约束; 约束分配; 松弛系数

中图分类号: TP317

文献标识码: A

文章编号: 0372-2112 (2016)04-0886-07

电子学报 URL: <http://www.ejournal.org.cn>

DOI: 10.3969/j.issn.0372-2112.2016.04.019

Research on QoS Decomposing and Service Status Monitor for Composite Service with Global QoS Constraint

LIU Zhi-zhong, PENG Hui, CAO Lei

(Department of CAISR Information System, University of PLA Science and Technology, Nanjing, Jiangsu 210007, China)

Abstract: Contradiction between the global QoS requirements and local service QoS status monitor decreases the availability of composite service, and makes it hard to meet users' requirements. Aiming at this problem, a framework for composite service adaption is proposed. And the mechanism for composite services QoS decomposing and individual service status monitor in this framework is demonstrated. To get the threshold of individual service, an approach to predict the service QoS by analyzing the historical QoS is studied. Further, to decrease the overhead of composite service adaption, the relax index is introduced into the mechanism. The experimental results show that predicting the QoS threshold and relax index can make the composite service QoS decomposing more reasonable, and can satisfy the user's end-to-end requirement through little adjustment to the greatest extent when changes happen.

Key words: QoS; global constraint; QoS decomposing; relax index

1 引言

服务组合通过组合多个已有服务以形成新的服务, 从而满足单个服务所无法满足的用户需求. 然而环境的开放性与动态性导致服务状态发生异常, 从而使组合服务不可用. 所谓的服务状态异常^[1], 包括服务的功能性异常(如: 服务的接口发生变化等)和服务的性能异常(如: 服务的响应时间超时等). 对于服务的功能性异常, 目前已有相应的研究工作展开: 包括服务的版本化、服务演化等. 在性能异常方面, 为了保证服务的可用性, 通常所采用的方法是服务状态监控, 然后根据监控结果对组合服务实施相应调整以满足用户的需求.

对于服务状态的监控, 研究人员也提出了各种不同的方法: 如基于 AOP (Aspect Oriented Programming) 的服务状态监控^[2]; 基于主动服务的 QoS 提交等. 但这些研究工作都有一个共同的假设: 用户针对组合服务中各个服务均给出了确切的 QoS 阈值, 为服务的 QoS 监控提供参考. 在此假设下, 对组合服务的监控将转变成对单个服务的 QoS 状态监控, 当单个服务的 QoS 超出规定的阈值时, 服务组合启动自适应机制以保证组合服务满足阈值.

在实际的服务组合需求中, 特别是在“以用户为中心”的服务组合中, 用户仅仅对组合服务提出全局的 QoS 约束, 而不关注个体服务的 QoS 阈值. 因此, 如何将

用户的全局 QoS 约束分解为各个服务的阈值,从而为服务的 QoS 监控提供参考值是“以用户为中心”的服务自适应机制的核心。

全局 QoS 约束分解可以看成是服务组合中 QoS 聚合^[3]的逆向问题. 单个服务的 QoS 以及服务的组合结构对 QoS 分解都有重要影响. QoS 分解的原则是分解结果须尽量反应 QoS 聚合中各组件服务的 QoS 在组合服务 QoS 中的重要性. 因此个体服务的 QoS 和聚合后的组合服务 QoS 是影响 QoS 分配的重要因素. 但是在服务监控系统中, 个体服务的精确 QoS 值是在监控以后才能获取, 缺乏预先性. 提前获取 QoS 值, 通常只能根据历史信息来预测在当前时刻可能的 QoS 值.

本文围绕“以用户为中心”的服务组合中服务状态监控展开研究, 重点研究单个服务的 QoS 预测机制与全局 QoS 分解机制.

2 相关研究工作

基于全局 QoS 约束的服务组合是服务组合领域研究的主要问题之一. 其基本思想是用户给定全局的 QoS 需求, 服务组合引擎以此需求为约束寻找一组服务以满足用户的功能需求. 此过程主要涉及多个服务的数据流和控制流如何组合以及服务的 QoS 聚合等相关问题. 该问题的研究通常将基于全局 QoS 约束的服务选择问题映射为多维背包问题, 使用混合整数规划的方法来完成服务组合. 但是, 该类方法的可扩展性较差, 当服务规模较大时, 无法满足实时性要求. 基于此, 北京邮电大学的王尚广等针对全局约束的自动服务组合提出了一种基于全局 QoS 约束分解的动态服务选择方法 (Web Service Dynamic Selection Approach, WSDSA)^[4]. 该方法通过基于模糊逻辑的自适应调整方法 (Adaptive Adjustment Method, AAM) 和自适应粒子群优化算法 (Adaptive Particle Swarm Optimization, APSO) 将全局 QoS 约束自适应地分解为满足用户偏好的局部约束, 然后利用局部最优获得最合适的组合服务. WSDSA 方法具有较好的有效性, 仅用较低的时间花费就得到最优解或接近全局最优解, 能够满足服务选择的实时性和动态性要求. 该方法通过约束分解机制可以降低问题搜索的空间, 提高服务组合的效率, 该方法为本文的全局 QoS 约束分解提供了参考.

Eduardo^[5]等提出了“以用户为中心”的服务组合框架, 他将自动服务组合的整个生命周期分为多个不同阶段, 并在所提出的 DynamiCos 中构建这些不同的组合组件. 此外, 为了实现“以用户为中心”, 为不同知识背景, 不同认知能力的最终用户提供个性化的服务, 在 DynamiCos 的基础上, 研究了自适应的 A-DynamiCos 系统, 支持异构的用户可以根据自身需要由领域专家制

定支持策略 (Support Strategy), 从而实现在服务组合过程中的指导作用. 但是该研究工作主要针对“以用户为中心”的服务组合中的功能性方面, 对于非功能性要求, 特别是由于环境动态变化所导致的服务非功能性要求变化并未考虑. 该研究重点突出了“以用户为中心”的特性, 强调了服务组合中人的参与.

KJ Lin^[6]等针对服务组合中的端到端约束及其自适应问题展开了研究. 他们通过研究责任模型来探测和诊断组合服务中的组件服务失效问题. 在研究中, 他们使用贝叶斯网络来确定服务组合中最可能出现问题的服务, 然后通过一个证据通道的选择算法确定服务组合中某个服务是否需要被检测, 并采用强化学习的方法来管理服务的动态性. 在此基础上, 他们还针对端到端约束的服务重配置问题展开了研究^[7,8]. 通过迭代来确定失效的替换区域, 从而扩展服务替换的选择区域以提高服务重配置的可能性. 此研究尚未涉及到如何将总体的 QoS 约束分配成个体服务的 QoS, 仅仅是基于服务的等级协议 (SLA) 来对服务实施检测.

在服务状态监控与组合服务自适应研究中, 服务 QoS 的预测是一个重要的研究内容. 研究人员针对此问题提出了大量的方法, 特别是针对数值型数据预测其预测模型有时间序列平滑预测法、回归模型、马尔科夫预测法、灰色系统预测等. 范小芹等提出了 Web 服务各随机 QoS 指标的度量方法和自适应 QoS 管理体系结构^[9]. 该结构根据 QoS 的历史信息, 建立服务 QoS 模型的分布模型, 包括均值、偏差等, 从而根据该模型来预测下一时刻的 QoS 值. 在获取到 QoS 预测值的基础上, 结合利用随机型离散事件系统的动态控制方法——马尔可夫决策过程 (MDP), 设计出随机 QoS 感知的可靠 Web 服务组合算法. 该算法通过考虑随机性 QoS 度量方法和 QoS 管理体系结构, 采用平衡了“风险”与“报酬”的 MDP 方法有效地提高服务组合的成功率.

文献[10]提出了一种 QoS 感知的自适应服务过程配置方法, 该方法通过重配置高效解决服务组合自适应问题, 且代价较小. 该方法在服务组合过程中选择备选的服务, 并将该服务作为备份资源以实现高效的重配置. 该文提出了基于区域的重配置算法, 通过识别一个有限的重配置区域以最小化重配置算法的代价.

3 服务状态监控与自适应框架

由于环境的动态性导致环境的非功能性属性发生变化, 从而使得原本满足用户需求的服务组合无法满足用户需求, 需要重新实施组合以适应环境的变化. 而这对于时效性要求较高的应用 (如: 军事侦察、信用卡交易) 等显然是不合适的. 因此, 服务组合机制通常需要为用户屏蔽这种环境变化所导致的服务组合失效.

组合服务自适应机制将自动基于服务状态的监控结果实施重新组合或调整组合以满足此需求.

基于 AOP 的思想,本文提出了如图 1 所示的服务监控与自适应框架.该体系结构包括以下几部分内容:

(1) BPEL 流程定义模块:主要依赖 BPEL 流程定义工具设计相应的 BPEL 流程,或者根据领域专家所设计的流程模板,以用户需求为导向,通过实例绑定来生成新的 BPEL 流程.

(2) BPEL 执行引擎:主要完成 BPEL 流程实例的创建,流程的实例的管理,包括流程状态的管理,以及 BPEL 执行过程中,各活动的调用等等.

(3) AOP 执行引擎: AOP 执行引擎主要为了支持 AO4BPEL^[11] 流程语言的支持,支持方面流程的部署,以及方面流程的执行.该执行将对 BPEL 流程引擎中的行为调用和变量访问进行截获,并根据方面流程的信息,在调用流程的活动之前(before)/之后(after)/前后(around)来调用相应方面的信息. AOP 执行引擎主要是在流程执行过程中,将方面流程和业务流程实施编织来实现对业务流程的监控.

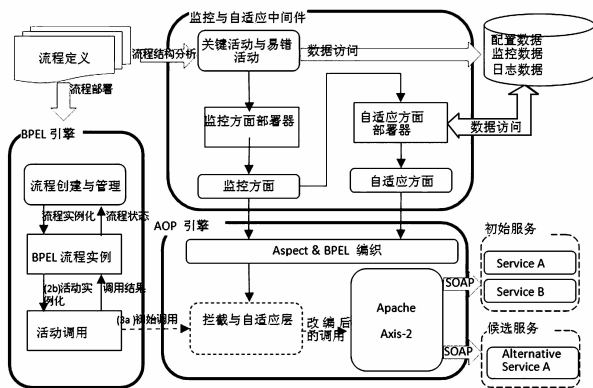


图1 基于AOP的服务监控与自适应框架

整个框架的核心是服务的状态监控和自适应机制,而且状态监控是组合服务自适应的触发条件.因此,准确地监控服务的状态,并根据监控所获得的状态信息精准地判定服务状态是否异常,是组合自适应机制的关键所在.基于 AOP 的服务状态监控方法将基于服务的 QoS 阈值和服务的监测值来判断服务的状态,其机制本身不是本文的重点,本文主要研究服务的 QoS 模型及如何获取监控所需要的阈值和 QoS 预测值.

4 面向全局约束的服务 QoS 分配与监控机制

在给定全局约束的情况下,将全局 QoS 分解为组件服务的 QoS 阈值,并对服务 QoS 状态实施监控是本文研究的核心.其基础则是服务的 QoS 模型.

4.1 服务的 QoS 模型

为了规范服务的 QoS 表示,通常需要对服务的 QoS 值进行归一化处理.经过归一化处理后的服务 QoS 值具有统一特性:(1) QoS 的取值范围统一为(0,1];(2) QoS 性质统一,即取值越大,表示服务的质量越差,所以 QoS 的值是越小越好.

对于服务的 QoS 状态通常由三个不同值来刻画: QoS 的真实值,用字母 q 表示;服务提供者给定的 QoS 声称值,用 \hat{q} 表示;由用户来设定的 QoS 约束值(或阈值),用 \hat{q} 表示.这三个值通常并不完全相同,但通常希望 QoS 声称值和 QoS 真实值一致.为了即促使服务提供者尽量提供真实的 QoS,研究人员研究了各种信誉激励算法.但在动态环境下,服务的 QoS 真实值难以实时获取,通常用用户的实时监控值来代替.在监控服务状态时,服务的 QoS 阈值是判定服务是否失效的一个参考值,也是实施组合服务自适应的触发条件.这些不同的 QoS 表示值之间的关系如图 2 所示.

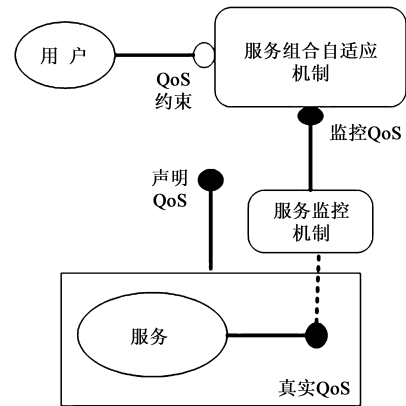


图2 服务各QoS值之间的关系图

在此模型下,组合服务触发自适应的时机是个体服务的 QoS 监控值超过其 QoS 阈值.但是,服务的 QoS 监控值只有在服务执行后才能获取,这对于组合服务的自适应缺乏必要的预先性.通常解决的方法是通过 QoS 预测机制来预测服务可能的 QoS 值,得到服务的 QoS 预测值 \hat{q} ,以预测值来替代真实值,从而实现对服务状态的事先判断.这样,服务自适应的核心问题就是得到服务的 QoS 阈值以及服务 QoS 的预测值.

4.2 全局约束下的组合服务 QoS 分解机制

通常在服务组合的用户需求中,用户只给定组合服务端到端的 QoS 约束,而不会给定单个服务的 QoS 阈值.为了监控个体服务,需要通过约束分解来获取服务的 QoS 阈值.和基于全局约束的服务 QoS 分解相对应的一个问题是服务组合中 QoS 聚合问题.服务 QoS 聚合指的是在给定各组件服务 QoS 值的情况下,根据组合服务的结构以及 QoS 属性得到组合服务的全局

QoS. 同样,组合服务的全局 QoS 约束分解机制也需要考虑服务 QoS 属性以及服务组合的结构. 本文的研究主要以服务 QoS 中的时间开销作为研究对象. 另外,考虑到服务组合的基本结构之间的可转化性和可组合性,本文仅研究最基本结构,即串行结构和并行结构.

研究表明^[12]:在串行结构中,组合的 QoS 是各组件服务 QoS 的累加求和,即给定串行结构 $Seq(S_1, S_2, \dots, S_n)$, 其中各组件服务 S_i 的 QoS 为 QoS_{S_i} , 得到全局 QoS 为 $QoS_{seq} = \sum QoS_{S_i}$. 而针对并行结构,组合的 QoS 值是 $QoS_{seq} = \text{Max } QoS_{S_i}$.

针对全局 QoS 的分解问题,可以得到一些相应的启示. 给定串行结构 $Seq(S_1, S_2, \dots, S_n)$ 和全局的 QoS 约束 QoS_c , 将全局 QoS 分解以得到各组件服务的 QoS 阈值最简单的方法是将全局 QoS 平均地分配到各组件服务中,即各组件 QoS 的阈值为:

$$\widehat{QoS}_{S_i} = QoS_c / n$$

给定并行结构 $Par(S_1, S_2, \dots, S_n)$ 和全局的 QoS 约束 QoS_c , 那么各组件服务的 QoS 阈值可以直接分解为:

$$\widehat{QoS}_{S_i} = QoS_c$$

该模型针对并行结构的约束分解是可行的,但是对于串行结构显然没有考虑到各个组件服务之间的差异,不同的服务其 QoS 阈值约束显然不同. 因此,在考虑服务的阈值分解时,需要结合服务目前的监控值来实施. 给定各服务的 QoS 监控值 QoS_{S_i} , 可以利用如下的 QoS 约束分解模型来获取各组件服务的 QoS 阈值:

$$\widehat{QoS}_{S_i} = QoS_{S_i} * QoS_c / \sum QoS_{S_i}$$

根据服务的监控值来分解服务的约束值对于服务组合的自适应具有滞后效应. 将上面的分解机制进行修正,利用服务 QoS 的预测值 \widehat{QoS}_{S_i} 来指导服务的全局 QoS 约束分解.

$$\widehat{QoS}_{S_i} = \widehat{QoS}_{S_i} * QoS_c / \sum \widehat{QoS}_{S_i}$$

为了正确合理地对全局的 QoS 约束进行分解,需要合适的服务 QoS 预测机制来预测服务的 QoS 值. 本文预测的目标并不是为了精确定量服务的 QoS 值,而是需要根据服务的 QoS 值来确定服务 QoS 的阈值,因此对于服务 QoS 的预测可以在现有的预测机制上做一定的改进.

4.3 基于极值的服务 QoS 预测机制

预测服务的 QoS 值主要有基于时间序列的方法、基于马尔科夫随机过程的方法以及基于灰色系统的方法等. 这些方法所预测的均是服务的 QoS 真实值. 本文预测的目标是服务 QoS 的阈值. 主要通过分析服务 QoS 真实值的历史信息,获得服务 QoS 的最大可能值,即 QoS 的极值,并基于此推断出服务的 QoS 阈值. 服务

QoS 的极值通常和 QoS 的变化规律相关. 有研究指出,以 QoS 值的自相关特性为依据,服务的 QoS 变化规律大致可以分为周期性 QoS 值,趋势性 QoS 值以及随机性的 QoS 值等,如下图 3 所示.

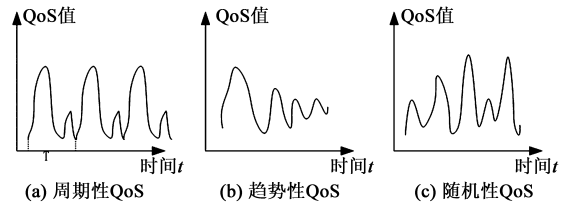


图3 服务QoS值的自相关特性

假定已知此前 k 个时刻的 QoS 值为序列 $t_{n-1}, t_{n-2}, \dots, t_{n-k}$, 需要预测当前 t_n 时刻的 QoS 值. 针对不同自相关特性的,基于极值的 QoS 预测方法将分不同情况来预测服务的 QoS 阈值.

1) 周期性 QoS

给定服务的 QoS 随时间变化的关系如图 3(a) 所示,服务的 QoS 阈值预测将以一个时间周期 T 作为 QoS 的预测周期,分析 t_n 所在的时间周期区域 $(t, t + T]$, 其中 $t \leq t_n \leq t + T$, 并利用区域对此前的 K 个数据进行截取,以保证 $t \leq t_{n-i} \leq t + T$, 得到时间序列 $t_{n-1}, t_{n-2}, \dots, t_{n-i}$, 其中 $i \leq k$. 基于此序列进行拟合,得到函数 $f: t \rightarrow (0, 1)$, 以获取 t_n 时刻的 $f(t_n)$. 此外,根据 QoS 监控的历史信息,可以获取该 QoS 周期中的最大值 QoS_{max} . 基于此预测 t_n 时刻的 QoS 阈值如下:

$$\widehat{QoS}_{t_n} = \text{Max} \{ QoS_{max}, f(t_n) \}$$

2) 趋势性 QoS

给定服务的 QoS 随时间变化的关系如图 3(b) 所示,即 QoS 的值呈稳定化的趋势发展. 在此情况下,给定此前 k 个时间点的 QoS, 据此进行拟合,得到函数 $f: t \rightarrow (0, 1)$, 以获取 t_n 时刻的 $f(t_n)$. 对于趋势性的 QoS, 在 t_n 时刻的服务 QoS 阈值为:

$$\widehat{QoS}_{t_n} = f(t_n)$$

3) 随机性 QoS

给定服务的 QoS 随时间变化的关系如图 3(c) 所示,即 QoS 值随机变化,没有规律性. 在此情况下服务的 QoS 拟合没有任何意义,因此,服务在 t_n 时刻的 QoS 阈值可以定义到在给定时间段的 QoS 的最大值,即:

$$\widehat{QoS}_{t_n} = \text{Max} \{ QoS_{t_n} \}$$

5 QoS 分解与预测机制评价

5.1 评价指标

为了验证全局 QoS 约束的分解与预测机制对组合服务状态监控的有效性,需要建立相应的性能评价体系来评价其效果. 由于用户的需求是全局 QoS 需求,因

此监控机制的目标是保证当某个服务的 QoS 值发生变化且导致组合服务全局 QoS 无法满足的情况下,需要触发服务组合自适应机制.同时,由于服务组合的自适应需要一定的时间开销,因此我们希望 QoS 分解与预测机制只有当单个服务的 QoS 变化且导致服务全局 QoS 无法满足时,服务自适应才被触发.评价体系需要反应该需求.为给出相应的评价体系,先做一些基本假设.

假设实验过程中对某复杂组合服务监控其 N 次执行;在整个监控期间,真正发生单个服务 QoS 状态异常且导致组合服务的全局 QoS 无法满足的次数为 N_T ,在这所有次数中,真正被监控到的次数为 N_{TT} ,其中未监测到次数为 N_{TF} ;在监控期间,监控机制共监控到由于服务 QoS 状态异常且导致全局 QoS 异常的次数为 N_F ,其中正确报警次数为 $N_{FT} = N_{TT}$,错误的报警次数为 N_{FF} .基于这些假设,可以定义如下的评价指标:

(1)漏报率

漏报率为真正发生了由于局部 QoS 异常而导致的全局约束违背,但是监控机制并未监控到的次数占总体发生次数的比例,可以表示如下:

$$1 - N_{TT}/N_T$$

和漏报率对应的是准确率,即真正发生局部 QoS 异常而导致的全局约束违背,且监控机制监控到该情况发生的次数占总监控到的次数的比例,即 N_{TT}/N_T .

(2)误报率

误报率为没有发生局部 QoS 异常而导致的全局约束违背,但是监控机制却认为该情况已发生的次数占总监控到的次数的比例,可以表示如下:

$$1 - N_{FT}/N_F$$

和误报率对应的为精确率,即真正发生局部 QoS 异常而导致的全局约束违背,且监控机制监控到该情况发生的次数占总监控到的次数的比例,即 N_{FT}/N_F .

误报率和漏报率存在一定的矛盾,通常误报率越低时漏报率可能会有所提高.监控机制所追求的目标是以尽可能低的代价来尽可能满足用户的需求.漏报将导致无法触发自适应机制最终无法满足用户需求,而误报则导致不必要的用户自适应,提高系统开销.因此,监控机制的目标将是“杜绝漏报,减少误报”.

5.2 松弛系数

在组合服务中,通常较少遇到多个服务同时发生异常的情况,大部分情况下是个别服务发生异常.而用户的需求则是全局 QoS 约束.因此,在计算服务的阈值时,我们在预测机制中引入了松弛系数(Relax Index) R 来放松服务的 QoS 阈值约束.引入松弛系数后, QoS 预测阈值可以修正如下:

(1)周期性 QoS

$$\widehat{QoS}_{t_n} = R * \text{Max} \{ QoS_{\text{max}}, f(t_n) \}$$

(2)趋势性 QoS

$$\widehat{QoS}_{t_n} = R * f(t_n)$$

(3)随机性 QoS

$$\widehat{QoS}_{t_n} = R * \text{Max} \{ QoS_{t_i} \}$$

其中 $R > 0$.

通过松弛系数的引入,一定程度上可以在误报率和漏报率之间进行取舍.当 $R = 1$ 时,直接将预测所得到的阈值作为其 QoS 阈值;当 $R > 1$ 时,将在预测所得到的阈值基础上做一定放松,从而得到真正的阈值.服务同时状态异常的概率小,因此实际阈值会比 QoS 的预测值稍大,因此松弛系统通常取值要大于 1;但是随着 R 增大,监控机制监控到服务失效的可能性会降低,也就是说监控机制的漏报率会比较低;因此,在实施 QoS 的预测与分配时,需要对松弛系数 R 进行相应的分析,以实现在杜绝漏报的情况下降低误报率.

5.3 实验分析

(1)实验设置

为了分析本文所提出的服务 QoS 分解与预测机制的有效性以及效率,本文以图 4 所示的复杂结构的组合服务为实验对象进行系列的实验.该结构包含了 17 个服务节点,节点之间的关系包括串行和并行两种.

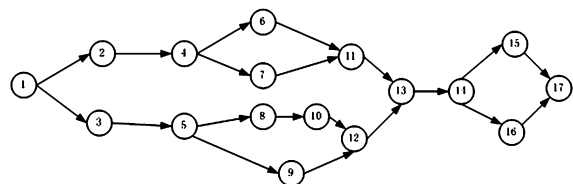


图 4 示例组合结构

本文研究的 QoS 以服务的执行时间为例.对于服务组合中的每个节点,假定该服务的执行时间为一个单值.此外,为了验证 QoS 属性的自相关特性对 QoS 预测和分解机制的影响,实验设定组合结构中各节点 QoS 的时间自相关特性如表 1 所示.

表 1 不同时间特性的节点

QoS 模型	节点序号
周期性 QoS	1, 5, 6, 8, 12, 15
趋势性 QoS	2, 4, 9, 10, 11, 16
随机性 QoS	3, 7, 13, 14, 17

模拟实验将随机地在这些服务中选择一个服务,人为改变其 QoS 值,让其发生状态异常.然后根据本文所提出机制来对个体服务的 QoS 进行预测并据此对全局 QoS 约束进行分解,然后通过监控单个服务的 QoS 值异常来推断组合服务是否违反全局 QoS 约束.此外,

实验过程中还考虑了松弛系数对整个预测与分解机制的影响. 为此, 本文共开展了两组实验.

第一组实验主要是比较本文所提出的预测模型与传统的预测模型对最终端到端 QoS 监控效果的影响. 传统的预测模型针对不同时间特性的 QoS 采用基于最小二乘法的曲线拟合方法(对于随机性 QoS 则采用前 M 个值取均值作为预测值). 本文所提方法在拟合的基础上再采用极值和松弛系数来对约束进行放松, 松弛系数采用固定值为 $R = 1.2$. 分别在不同条件下共进行 100 次实验. 在这 100 次实验中, 随机地设定某个服务节点 QoS 值异常变化. 在传统的预测模型中, 当服务的实际值超过预测值的 20% 时, 触发组合服务自适应; 而

在本文所提出的预测与分解机制中根据全局目前的约束情况来计算服务的 QoS 阈值, 并在此基础上根据预测的 QoS 值和预测的阈值进行比较, 当 QoS 超出阈值时, 系统认为组合服务违反了端到端 QoS 约束.

第二组实验的目标是比较在本文所提出方法中, 松弛系数对不同时间特性 QoS 属性的影响. 因此, 分别针对 QoS 具有不同时间特性的服务节点进行实验, 不同节点在取不同 R 值的情况下分别进行 50 次实验, 最终得到漏报率和误报率的变化. 然后分析得到 R 对具有不同时间特性的 QoS 属性的服务节点的影响.

(二) 实验结果及其分析

第一组实验取得了如表 2 所示的实验结果.

表 2 第一组实验的结果

QoS 类型	实验次数	实际异常次数	局部监控异常次数			基于极值的预测分解监控异常次数		
			监控到次数	未监控次数	监控错误次数	监控到次数	未监控次数	监控错误次数
周期性 QoS	35	10	11	2	3	10	1	1
趋势性 QoS	28	15	15	1	1	16	0	1
随机性 QoS	37	21	18	5	2	20	2	3

表 3 不同松弛系数下的实验结果

QoS 类型	$R = 0.8$					$R = 1.0$				
	监控到次数	未监控到次数	监控错误次数	漏报率	误报率	监控到次数	未监控到次数	监控错误次数	漏报率	误报率
周期性 QoS	60	5	15	0.1	0.25	58	3	11	0.06	0.19
趋势性 QoS	55	3	8	0.06	0.15	54	1	5	0.1	0.093
随机性 QoS	65	8	23	0.16	0.35	55	3	8	0.06	0.145
QoS 类型	$R = 1.2$					$R = 1.3$				
	监控到次数	未监控到次数	监控错误次数	漏报率	误报率	监控到次数	未监控到次数	监控错误次数	漏报率	误报率
周期性 QoS	53	2	5	0.04	0.09	53	3	0	0.02	0.057
趋势性 QoS	52	1	3	0.02	0.058	52	2	4	0.04	0.077
随机性 QoS	53	1	4	0.02	0.075	53	5	8	0.1	0.15

上面实验所得到的数据可以用图 5 表示.

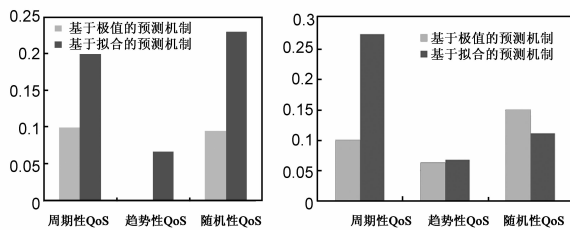


图 5 基于极值的预测与分解机制和拟合预测机制的比较

上面的图表分析显示, 当 $R = 1.2$ 时, 基于极值的预测与分解机制可以在不降低监控系统的误报率的情况下, 降低监控系统的漏报率. 实验结果说明不管针对何种时间特性的 QoS, 基于极值的预测与分解机制方法都能够更好地实现通过监控局部 QoS 值来判定组合服务是否满足全局的 QoS 约束.

第二组实验得到如表 3 所示的实验结果.

基于上面的实验结果, 针对不同时间特性的 QoS 属性, 分别比较不同的松弛系数对服务状态监控的漏

报率和误报率的影响, 如图 6 所示.

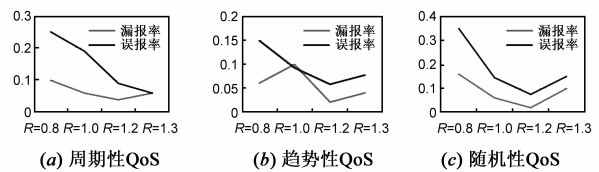


图 6 松弛系数与漏报率、误报率的关系

上面的比较表明: 不管 QoS 属性具有何种时间相关性, 为了达到监控目标(通过监控个别服务来预测全局 QoS 约束违背), 在一定的范围内(约 $R < 1.2$), 松弛系数 R 须越大越好, R 越大 QoS 监控的误报率和漏报率均有所降低; 针对不同时间特性的 QoS 属性, 为了获得最佳的监控效果, R 的取值有所不同.

6 结论和未来的工作

用户全局 QoS 约束的全局性与服务 QoS 状态监控的局部性之间的矛盾使得研究全局 QoS 约束下的约束分解成为必要. 通过借鉴基于全局约束的 QoS 感知的服务

组合中 QoS 聚合的相关技术,本文研究了全局 QoS 约束分解机制以及单个服务的 QoS 预测机制.为了在尽量杜绝漏报率的情况下,降低服务状态监控的漏报率,本文针对不同特性的 QoS,在基于服务 QoS 的历史信息的基础上,采用了基于极值的 QoS 预测方法来启发全局 QoS 的分解,并且在服务状态监控中引入了松弛系数.通过设置不同的松弛系数,可以在一定的程度上调节服务状态监控的误报率和漏报率,并使之达到最佳.

下一步的研究重点将是优化松弛系数 R .如果服务组合自适应框架能够根据历史的松弛系数和漏报率与误报率之间的关系,自动调整或选择最佳的松弛系数,这对于实现“以用户为中心”的服务组合自适应机制将是非常有用的.此外,本文的研究是在已有的 QoS 预测机制上的改进,目前 QoS 预测机制多种多样,本文主要针对基于曲线拟合的预测机制,基于其他预测机制的改进将是本文下一步的研究工作.

除了单一服务状态异常外,同时多个服务状态发生异常也是组合服务中会遇到的问题.如何在多个服务同时发生异常时进行监控检测,并判定组合服务是否违反全局约束也是我们下一步需要解决的问题.

参考文献

- [1] Yi Wang, Ying Wang. A survey of change management in service-based environments [J]. *Service Oriented Computing and Applications*, 2013, 7(4): 259 – 273.
- [2] Anis Charfi, Mira Mezini. AO4BPEL: an aspect-oriented extension to BPEL [J]. *World Wide Web*, 2007, 10(3): 309 – 344.
- [3] Liu Zhizhong, He Yihui, Wang Yong. QoS aggregation for end-to-end qos requirements in service composition [A]. *Proceedings of International Conference on Computer and Management* [C]. Wuhan, Hubei; IEEE, 2012. 1390 – 1393.
- [4] 王尚广, 孙其博, 杨放春. 基于全局 QoS 约束分解的 Web 服务动态选择 [J]. *软件学报*, 2011, 22(7): 1426 – 1439. WANG Shang-Guang, SUN Qi-Bo, YANG Fang-Chun. Web service dynamic selection by the decomposition of global QoS constraints [J]. *Journal of Software*, 2011, 22(7): 1426 – 1439. (in Chinese)
- [5] Eduardo Manuel Gonçalves da Silva. User-centric service composition-towards personalised service composition and delivery [D]. Enschede, Holland; University of TWENTE, 2011.
- [6] Yue Zhang, Kwei-Jay Lin, Jane Y J Hsu. Accountability monitoring and reasoning in service-oriented architectures [J]. *Service Oriented Computing and Applications*, 2007, 1(1): 35 – 50.
- [7] Kwei-Jay Lin, Jing Zhang, Yan long Zhai. An efficient approach for service process reconfiguration in SOA with end-to-end QoS constraints [A]. *Proceedings of IEEE Conference on Commerce and Enterprise Computing* [C]. Vienna, Austria; IEEE, 2009. 146 – 153.
- [8] 翟岩龙. 开放网络环境中动态自适应服务组合技术研究 [D]. 北京: 北京理工大学, 2009. Zhai Yan-Long. *Research on dynamic adaptive service composition in open network environment* [D]. Beijing: Beijing Institute of Technology, 2009. (in Chinese)
- [9] 范小芹, 蒋昌俊, 王俊丽, 庞善臣. 随机 QoS 感知的可靠 Web 服务组合 [J]. *软件学报*, 2009, 20(3): 546 – 556. FAN Xiao-Qin, JIANG Chang-Jun, WANG Jun-Li, PANG Shan-Chen. Random-QoS-aware reliable web service composition [J]. *Journal of Software*, 2009, 20(3): 546 – 556. (in Chinese)
- [10] Jing Li, Dianfu Ma, Xiupei Mei. Adaptive QoS-aware service process reconfiguration [A]. *Proceedings of International Conference on Service Computing* [C]. Washington, DC; IEEE, 2011. 282 – 289.
- [11] Anis Charfi, Mira Mezini. AO4BPEL: an aspect-oriented extension to BPEL [J]. *World Wide Web*, 2007, 10(3): 309 – 344.
- [12] Zeng LZ, Benatallah B, Dumas M. Quality driven Web services composition [A]. *Proceedings of the 12th Int' l Conf on World Wide Web* [C]. New York; ACM, 2003. 411 – 421.

作者简介



刘志忠 男, 1980 年 2 月生于江西吉安, 博士, 讲师, 主要研究领域为面向服务计算、指挥信息系统工程.

E-mail: lzz_liu@126.com



彭辉 男, 1980 年 3 月生于湖北武汉, 博士, 讲师, 主要研究方向为多 UAV 协同控制与优化、服务优化.

E-mail: penghui_ph@163.com