

文章编号:1671-8860(2013)05-0626-05

文献标志码:A

# 云环境下可信服务的时效策略

顾 鑫<sup>1</sup> 徐正全<sup>1</sup> 涂洪涛<sup>2</sup> 刘小俊<sup>1,3</sup>

(1 武汉大学测绘遥感信息工程国家重点实验室,武汉市珞喻路129号,430079)

(2 武汉软件工程职业学院计算机与软件学院,武汉市东湖新技术开发区光谷大道117号,430205)

(3 黄冈师范学院物理与电子信息学院,湖北黄冈市开发区新港二路146号,438000)

**摘要:**针对宿主式云计算在可信服务中缺乏高可用性的特点,从可信服务评价的角度,提出了基于宿主云时效策略,并建立了时效模型。通过采用阈值分隔、诱导有序加权平均(IOWA)算子、透明信任评价和动态优先级调度思想,对宿主云服务节点的可信度做出动态计算。实验结果表明,宿主云时效策略能优先选择可信度高的服务资源,在一定程度上提高了宿主云服务安全应用的稳定性。

**关键词:**宿主云;时效策略;可信评估

**中图法分类号:**TP393.08

随着世界经济全球化和信息技术的不断发展,云技术已成为当前信息领域的又一个热点。目前,英特尔、微软、亚马逊、谷歌、IBM等公司都提出了“云计划”,比较典型是由微软提供的宿主服务器架构模式的云服务平台。它允许将网络启动服务器安放至互联网的任何地点,甚至连智能移动终端也可以建立云服务。尽管宿主云系统功能强大,可以无限虚拟服务资源,但在信任和安全问题等方面极具争议<sup>[1]</sup>。宿主云服务架构设计提供极大自由度,同时也带来不稳定的因素。众多自建的云服务资源加入,使宿主云无法提供高可用性和管理上的灵活性。较高质量要求可信服务存在单点失败的可能,造成云服务交付延时或失效。因此,云服务的可信性最终体现在它是否能通过预期、评估等手段,选择合适的可信服务资源节点来提供服务。

## 1 时效策略

宿主云节点提供稳定的可信服务,对可信度高、服务稳定的资源进行选择,而时效则是体现服务资源稳定可信的重要参数。Beth、Josang等人分别提出了基于经验和概率解释的信任评估模型和 subjective logic 模型<sup>[2-3]</sup>,但模型并没有充分考虑到信任时效问题。王守信等利用云模型理论<sup>[4]</sup>

对信任管理问题进行了建模,但模型中仅用等比衰减数列简单的描述信任时效性。李小勇等在提出了多维信任评估指标模型<sup>[5]</sup>的同时,涉及到信任时效的简单评价。本文从可信服务评价的角度,针对宿主云服务中不稳定因素,提出了一种云服务资源节点选择时效策略。

可信服务的宿主云时效策略归纳为以下几步:  
① 判断是否有空闲的宿主云服务节点,若无,则返回继续搜索;若有,则对所有空闲节点进行时效性评价;  
② 建立时效模型计算得到空闲资源节点信任值;  
③ 利用节点可信度与“动态级”的乘积作为节点选择参考量,判断当前已搜索云节点是否能优先提供可信服务,若不符合,则舍去重新搜索;若符合,要求则分配该节点任务提供全部或部分服务;  
④ 返回步骤①,继续搜索节点分配余下服务任务。

## 2 时效模型

### 2.1 模型定义

自然信任随时间衰减变化<sup>[2-5]</sup>的属性存在于计算机网络。借用金融学时间价值的含义,该资源节点在等待服务过程中随着时间推移而产生的信任贬值。宿主云服务节点的当前可信度需要参考历史评估,因此,在建立时效模型时应有几点考虑:  
① 仅反映信任评价值本身与时间的关系,与

评价的角度无关。② 历史评价中距当前评价时刻太久的信任评价记为评价过期或评价失效。③ 历史评价中距当前评价时刻越近的信任评价,参考价值越高。④ 时效模型可以根据客观要求,改变历史评价数据间关联的重要性。

综上所述,假设时效模型由可衡量效用历史评价合集与随机效用评价误差构成,且两个效用指标是线性关系,则建立如下公式表示时效模型:

$$H = f(x_L - \Delta x \otimes F(t - L)) \quad (1)$$

式中,  $f(\cdot)$  为信任时效计算函数,  $f(x) = x$ ;  $x_L$  为节点经过  $L$  时刻的评价信任度,反映节点提供服务固定观测属性;  $\Delta x$  为信任度误差项,代表了信任随时间变化的随机效用;  $F(t - L)$  是时间函数;  $\Delta x \otimes F(t - L)$  代表距离评价时刻时间  $L$  对当前评价信任度的影响作用。

## 2.2 当前评价信任度 $x_L$

设  $K = \{x(1), x(2), \dots, x(k)\}$  为包含  $k$  个历史评价的历史评价合集,  $x_L$  是对历史评价合集中每个历史评价数据的加权计算, 定义  $x_L$  为:

$$x_L = f(K) = \sum_{i=1}^k x(i) \omega_i \quad (2)$$

且所有权值  $\omega_i$  的总和为 1。

由模型的定义可知,需将历史评价合集分为两个部分,并仅对时刻较近的历史数据进行加权计算。令有时间阈值  $T_y$  将时间坐标轴分为两部分,坐标轴起点为评价开始时刻,坐标轴终点为当前评价时刻  $T_N$ ,即  $x_L$  的下标时刻  $L$ ,将历史评价合集  $K$  分解为两个子集合,分别代表历史有效评价集合  $K_a$  与历史失效评价集合  $K_{una}$ ,且满足  $K_{una} \cup K_a = K$ ,且  $K_{una} \cap K_a = \emptyset$ 。由于历史评价数据发生时刻并不是等间隔,求解每个历史评价值权重时,不能直接按照时间比例进行权值分配。为了区分不同历史评价对时间的重要性,将历史有效评价集合  $K_a$  其所在的时间段分割为  $m$  个子区间,标记此子区间为时间窗口  $W$ 。假设落在同一时间窗口中的历史评价值都具有相同的时效权重,即  $\omega_W/n$ ,如图 1 所示。

由于时间窗口间隔相等,并且窗口时间严格顺序,本文引入文献[6]提出的有序加权平均算子和诱导有序加权平均算子算法来计算历史时效权重  $\omega_i$ 。根据文献[7]所提出的最大熵诱导有序加权平均值计算方法,代入式(2)可以求出:

$$x_L = \sum_{i=1}^k x(i) \times \omega_i = \sum_{i=1}^k x(i) \times \frac{\sqrt{\omega_{W_1} (m-j) \omega_{W_m} (j-1)}}{n}$$

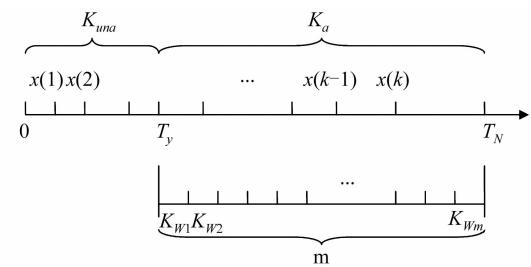


图 1 时效模型

Fig. 1 Aging Model

式中,  $x(i)$  为集合  $K_a$  的第  $i$  个历史评价数据;  $j$  表示  $x(i)$  所在的第  $j$  个时间窗口;  $\omega_W$  为第  $j$  时间窗口的权重值;  $n$  为第  $j$  个时间窗口内所包含的历史评价数据个数。

1) 时效性判定。阈值  $T_y$  限定当前节点对可用历史记录  $x(i)$  的取值范围,发生在阈值  $T_y$  时刻前的所有历史评价不参与信任度计算,反之则参与。当  $T_y = 0$  时,则表示整个历史评价合集参与当前时刻评估计算。

2) 窗口数  $m$  反映的是宿主云服务节点对历史时间价值重要程度的分级。若将时间价值重要性分为 5 个等级{不太重要,重要,比较重要,很重要,非常重要},则  $m=5$ 。实际评价中应按照时间价值的重要性选取合适的时间窗口数。

## 2.3 误差项 $\Delta x$

采用经验预测法,通过对大量云资源节点的数据观测分析<sup>[8-9]</sup>,得到  $\Delta x$  的经验分布  $F(x_i)$ 。通过经验分布的两次对数发现,  $\ln(-\ln(F(x_i)))$  与  $x_i$  呈近似线性关系,符合 Gumbel 分布函数的性质。因此,可假设  $\Delta x$  服从位置参数为  $u$ 、刻度参数为  $\delta$  的 Gumbel 极值分布。

由于  $\Delta x$  对  $i$  是相互独立的,  $\Delta x$  的渐近分布估计  $\hat{G}(x_i) = \exp\{-\exp(-(x_i - \hat{u})/\hat{\delta})\}$ ,其中  $\hat{u}$  和  $\hat{\delta}$  是采用 Gumbel 分布对经验数据拟合出的极大似然估值。

## 2.4 时间影响量 $\Delta x \otimes F(t - L)$

$\Delta x \otimes F(t - L)$  反映的是信任误差量  $\Delta x$  随着时间  $L$  变化的量。根据经验假设发生评价时刻服从刻度参数为  $\gamma$  的指数分布,则有:

$$F(t - L) = \int_L^{+\infty} (t - L) f(t) dt = \exp(-\gamma L)/\gamma$$

结合文献[10]的研究结果,评价时刻  $L$  和  $\Delta x$  的关系式为  $L = 1/(1 - \hat{G}(x_i))$ 。

## 2.5 模型简化

通过以上计算,并顾及式(1),时效模型为:

$$H = \sum_{i=1}^k x(i) \times \sqrt[m-1]{\omega_{W1} (m-j) \omega_{Wm} (j-1)} / n - \{ \hat{a} - \hat{\delta} \ln[-\ln(1 - \gamma / \exp(-\gamma L))] \}$$

上式中隐含的条件为:

$$\alpha = \frac{1}{m-1} \sum_{j=1}^m (m-j) \omega_{Wj}$$

由于刻度参数  $\hat{\delta} > 0$ , 当满足  $0 \leq L + 1/\gamma$  时(在实际情况中可以实现), 可以求得  $\partial H / \partial L < 0$ , 即表示时效模型评价值  $H$  随着时间  $L$  单调递减。当时间足够长时, 信任评价值减小到零, 符合信任衰减的特点。规定  $H$  的取值范围在  $[0, 1]$  之间。

时效模型中隐含的时效影响因子  $\alpha$  代表  $m$  个时间窗口之间的重要程度。当  $\alpha$  变大时, 即时效模型中  $x_L$  对离评价时刻越近的评价数据的权重就越大。特别地, 当  $\alpha = 0.5$  时, 每个时间窗口权重相同, 得到的  $x_L$  为历史评价合集的平均值。当  $\alpha = 1$  时, 有时间窗口权重矢量  $\omega = (0, 0, \dots, 1)$ , 此时信任评价值等于离发生评价时刻最近的一次历史评价值。所以,  $\alpha$  取值范围为  $[0.5, 1]$ 。

若已经存在  $n$  次历史评价, 则令当前时刻通过时效模型产生的评价值为  $x(n+1)$ 。当  $\alpha = 1$  时, 时效模型中的  $x_L$  等于最后一次历史评价  $x(n)$ , 时效模型可以简化为如下递推式:

$$x(n+1) = x(n) - g(L), n \geq 0 \quad (3)$$

式中,  $g(L) = \hat{a} - \hat{\delta} \ln[-\ln(1 - \gamma / \exp(-\gamma L))]$ ; 当  $n=0$  时,  $x(0)$  代表的是资源节点的初始经验值, 可以参照云服务中其他相似的资源节点给出, 也可以通过经验值给出。

简化的时效模型可以解决以下两个问题: ① 递推式可以避免对所有云资源节点的历史评价进行时效权重计算, 简化计算开销, 更方便地获得当前时刻评价预测值。② 若存在新加入的云资源节点或者节点交互较少, 历史评价次数不足以套用时效模型计算的情况,  $x_L$  不能直接利用式(2)进行计算。若系统能根据经验给出信任度初值  $x(0)$ , 可以利用简化递推式(3)求出评价时刻的信任度。

### 3 时效策略算法 CTDLS

传统的 DLS 算法在调度决策时, 能够适应资源异构性特征, 但没有考虑资源的可信程度的影响<sup>[11]</sup>。基于以上时效模型的设计, 本文提出了基于宿主云的时效策略算法——CTDLS。该算法在保留了 DLS 调度优点的同时, 综合考虑云资源节点信任度, 其算法  $CTDLS(v_i, m_j)$  定义如下:

$$CTDLS(v_i, m_j) = CT(v_i, m_j) \times DL(v_i, m_j) = H \times (SL(v_i) - \max\{t_{i,j}^A, t_j^M\} + \Delta(v_i, m_j))$$

式中,  $CT(v_i, m_j)$  表示云服务端把子任务  $v_i$  调度到云节点  $m_j$  上的可信度  $H$ , 由时效模型的定义可知,  $CT(v_i, m_j)$  取值介于  $[0, 1]$  之间;  $SL(v_i) = \max\{t_{i,j}^A, t_j^M\} + \Delta(v_i, m_j)$  是 DLS 算法,  $SL(v_i)$  为任务静态级, 在一个调度期内, 它是一个常数;  $\max\{t_{i,j}^A, t_j^M\}$  表示任务  $v_i$  到节点  $m_j$  上开始执行的时间,  $t_{i,j}^A$  表示任务  $v_i$  到节点  $m_j$  上时所需的数据输入时间,  $t_j^M$  表示节点  $m_j$  空闲时可用于执行任务  $v_i$  的时间;  $\Delta(v_i, m_j) = t_i^E - t_{i,j}^A$  反映计算机性能,  $t_i^E$  是任务  $v_i$  在所有空闲机器上执行所需时间的期望,  $t_{i,j}^E$  表示任务  $v_i$  在节点  $m_j$  上执行的时间。

## 4 实验对比分析

### 4.1 仿真环境与参数设定

仿真硬件环境采用实验室服务器作为云平台。软件环境采用云计算仿真软件 CloudSim, 模拟宿主云服务资源节点提供云服务。在仿真环境中, 机器在交互的信任度在  $[0, 1]$  中随机产生, 每个任务在资源上的交互执行最大时间与链路数量恒定, 每个子任务的输入、输出数据量是从  $[10, 1000]$  中循环选取。设定资源节点的时效模型窗口典型值为  $m=5$ , 时间阈值为 0(即所有的历史评价都参与计算), 时效影响因子  $\alpha$  分别为 0.8 和 1 两组, 并对两组进行 100 次实验, 实验结果用 100 次均值表示。

### 4.2 仿真及结果分析

图 2 和图 3 为普通任务情况下, 采用时效影响因子  $\alpha$  分别为 0.8 和 1 仿真, 平均成功率和平均调用长度与传统 DLS 对比结果如图 2。

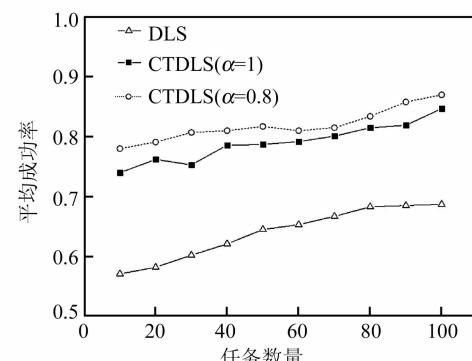


图 2 平均执行成功率

Fig. 2 Average Ratio of Successful Execution

图2中,随着任务数的增长,三者的结果都呈上升趋势。DLS算法由于没有考虑资源节点失效导致任务调度失败的风险,任务提交的成功率在0.65左右浮动。而本文采用的时效策略算法在任务调度的成功率方面明显高于DLS算法,且效影响因子 $\alpha$ 取值为0.8时较取值为1时高。这是因为效影响因子 $\alpha$ 在取值为0.8时综合考虑了所有节点的历史变化,取值为1时仅考虑节点最后一次历史评价,更能反映资源节点在整个任务过程中资源云节点的信任程度,所以前者比后者的曲线更加平滑,提高了任务调度成功率。

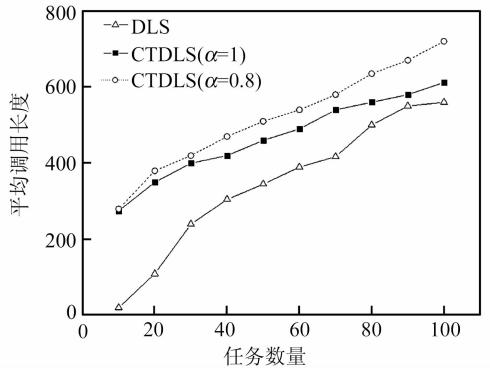


图3 调用时间长度

Fig. 3 Average Schedule Length

对比图3中反映的平均调用长度结果可知,DLS算法的平均调度时长均小于两种时效因子取值下的平均时长,随着任务数量的增加,时效策略调用长度曲线略呈增大趋势。结合图2、图3,虽然时效策略任务平均调度时长要大于DLS算法,但云服务资源执行成功率较DLS算法有很大的提升,这说明时效策略以牺牲了调度时间为代价来换取提交任务的成功率。平均调用时长在时效因子 $\alpha$ 取值为0.8时较取值为1时长,这是因为后者利用了时效策略模型中的简化递推式(3),避免了对全部历史评价计算的过程。随着任务数量的增加,递推公式对调用时长的优势越来越明显,从而后者平均调用长度较前者要短。

通过以上实验分析对比可以得出:①时效策略是以牺牲调度时长来获得可信的执行环境,且以较小的时间代价换取较大的任务执行成功率;②时效因子的变化使得时效策略具有高灵活性。针对不同类型的云服务,可以过设置时效因子来制衡调度任务时间与执行环境可信性之间的矛盾。总的来说,采用时效策略选取云服务资源节点更为可靠,在一定程度上降低了节点失效带来的影响。该策略具有良好的借鉴作用,不仅适用于宿主式云服务,同时也适用于大规模网络任务调度。

## 参 考 文 献

- [1] 李虹,李昊. 可信云安全的关键技术与实现[M]. 人民邮电出版社, 2010, 414-420
- [2] Beth T, Boreherding M, Klein B. Valuation of Trust in Open Networks[C]. Proceedings of the European Symposium on Research in Computer Security, Brighton, U K, 1994
- [3] Jøsang A, Knapskog S J. A Metric for Trusted Systems[C]. Proceedings of the 21st National Security Conference, Crystal City, Virginia, 1998
- [4] 王守信,张莉,等.一种基于云模型的主观信任评价方法[J].软件学报, 2010, 21(6): 1 341-1 352
- [5] Li Xiaoyong, Zhou Feng, Yang Xudong. A Multi-dimensional Trust Evaluation Model for Large-scale P2P Computing[J]. Journal of Parallel and Distributed Computing, 2011, 71:837-847
- [6] Yager R R. On Ordered Weighted Averaging Aggregation Operators in Multicriteria Decision Making[J]. IEEE Trans on Systems, Man, and Cybernetics, 1988, 18(1):183-190
- [7] Fullér R, Majlender P. An analytic Approach for Obtaining Maximal Entropy OWA Operator Weights[J]. Fuzzy Sets and Systems, 2001, 124 : 53-57
- [8] 顾鑫,徐正全,刘进. 基于云理论的可信研究及展望[J]. 通信学报, 2011, 132(7):176-181
- [9] 顾鑫,徐正全,方艺霖,等. Mobile WiMAX 可信服务系统的研究[J]. 武汉大学学报·信息科学版, 2012, 37(4):500-503
- [10] 罗纯,王筑娟. Gumbel 分布参数估计及在水位资料分析中应用[J]. 应用概率统计, 2005, 21(2): 169-175
- [11] Wang Wei, Zeng Guosun, Zhang Junqi, et al. Dynamic Trust Evaluation and Scheduling Framework for Cloud Computing[J]. Security and Communication Networks, 2012, 5:311-318
- [12] 黄书强,付仲良. 一种干扰避免的无线 Mesh 网络信道分配算法[J]. 武汉大学学报·信息科学版, 2013, 38(2):248-251
- [13] 肖汉,周清雪,张祖勋. 基于多 GPU 的 Harris 角点检测燕行算法[J]. 武汉大学学报·信息科学版, 2012, 37(7):876-881
- [14] 周平,唐新明,张过. 多时相统计数据空间动态可视化模型研究[J]. 武汉大学学报·信息科学版, 2012, 37(9):1 130-1 133

第一作者简介:顾鑫,博士生,主要研究方向为多媒体通信、可信网络、网络安全、云可信等。

E-mail:xuzq@whu.edu.cn

## Time Effectiveness in Trust Services Under Cloud Environment

GU Xin<sup>1</sup> XU Zhengquan<sup>1</sup> TU Hongtao<sup>2</sup> LIU Xiaojun<sup>1,3</sup>

(1 State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing,  
Wuhan University, 129 Luoyu Road, Wuhan 430079, China)

(2 Compute and Software College, Wuhan Vocational College of Software and Engineering,  
117 Optical Valley Street, Wuhan 430205, China)

(3 School of Physical Science and Technology, Huanggang Normal University, 146 Xingang 2nd Road, Huanggang 438000, China)

**Abstract:** Dealing with the problem of lacking high availability in trust services under hosting cloud computing environment, an time effectiveness strategy and an time effectiveness model for hosting cloud computing were proposed from the perspective of service evaluation in trust. It dynamically calculate trustworthiness degree of hosting cloud service node by utilizing threshold methods, IOWA (induce ordered weighted averaging) operators, transparent trust evaluation and dynamic level scheduling algorithm. Our experimental results showed that this hosting cloud time effectiveness strategy can choose high credibility service resources in priority, and enhanced the stability and availability of cloud services application to some extent.

**Key words:** cloud hosting; time effectiveness strategy; trust evaluation

**About the first author:** GU Xin, Ph. D candidate, majors in the media communication, trusted network, network security and trust cloud.  
E-mail: xuzq@whu.edu.cn

(上接第 607 页)

## Grid Pattern Recognition in Road Network Using Principal Component Analysis

TIAN Jing<sup>1, 2</sup> HE Qiu<sup>1</sup> ZHOU Mengjie<sup>1</sup>

(1 School of Resource and Environment Science, Wuhan University, 129 Luoyu Road, Wuhan 430079, China)  
(2 Key Laboratory of Geographic Information System, Ministry of Education,  
Wuhan University, 129 Luoyu Road, Wuhan 430079, China)

**Abstract:** The spatial pattern recognition and implicit information discovery have generated interest in the field of spatial data mining. Grid pattern is one of the most typical patterns in road networks. This paper presents a method for grid pattern recognition using principal component analysis. The method first defines shape measures and relation measures to formalize meshes in road networks. The measures are: rectangularity, convexity, consistent arrangement degree, rectangularity of mesh having largest consistent arrangement degree, convexity of mesh having largest consistent arrangement degree. Second, the principal components are created and analyzed. Finally, the method identifies grid pattern via the meshes' membership degree of the grid derived from the first principal component. Experiments on Shenzhen road network and Wuhan road network have been conducted. The results show that it is valid in grid pattern recognition and it is a versatile method to some extent.

**Key words:** road network; grid pattern; pattern recognition; principal component analysis

**About the first author:** TIAN Jing, Ph. D, lecturer, majors in automated map generalization and spatial data mining.

E-mail: yutaka-2010@163.com