ISSN 1000-9825, CODEN RUXUEW
Journal of Software, Vol.18, No.8, August 2007, pp.1968−1986
DOI: 10.1360/jos181968
© 2007 by *Journal of Software*. All rights reserved.
E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

自组织虚拟计算环境中的组信誉机制*

唐扬斌[1+], 王怀民[1,2], 常俊胜[1]

[1](国防科学技术大学 计算机学院,湖南 长沙 410073)

[2](国防科学技术大学并行与分布处理国家重点实验室,湖南 长沙 410073)

Clique-Based Reputation System in Self-Organizing Internet-Based Virtual Computing Environment

TANG Yang-Bin[1+], WANG Huai-Min[1,2], CHANG Jun-Sheng[1]

[1](School of Computer, National University of Defense Technology, Changsha 410073, China)

[2](National Laboratory for Parallel and Distributed Processing, National University of Defense Technology, Changsha 410073, China)

+ Corresponding author: Phn: +86-731-2586713, Fax: +86-731-4518311, E-mail: ybtang21c@gmail.com, http://www.nudt.edu.cn

Tang YB, Wang HM, Chang JS. Clique-Based reputation system in self-organizing Internet-based virtual computing environment. *Journal of Software*, 2007,18(8):1968−1986. http://www.jos.org.cn/1000-9825/18/1968.htm

Abstract: By borrowing the construction and evolution mechanisms of the social groups, a reputation system based on groups of autonomic elements (AE) is proposed, which is called clique-based reputation system (CRS). With the combination of the clique-based reputation evaluation and reputation inspired clique evolution, CRS makes the reputation model more realistic on evaluation of the characteristic of AE behaviors and better convergent properties. Meanwhile, the node cluster results from the clique evolution make a foundation for the differential service for AE based on reputation, incentivize them to take corporative behaviors, and provide reliable services.
Key words: self-organizing; Internet-based virtual computing environment; reputation; incentive

摘 要: 借鉴人类社会中的组织构造与演化机制,提出一种基于组结构的信誉机制——组信誉机制.通过组信誉评价和信誉驱动的组演化机制的结合,一方面使得信誉模型具有更好的自主元素行为刻画能力和收敛特征;另一方面,基于组演化形成的自主元素聚集能够为实现自主元素基于信誉的差异化服务奠定基础,达到激励其积极参与和可靠服务的目的.
关键词: 自组织;虚拟计算环境;信誉;激励
中图法分类号: TP393 **文献标识码**: A

近年来,随着Grid[1],P2P[2,3]等计算模式的兴起,基于节点自组织的方式实现计算资源的广泛聚合和综合利用已经成为构造互联网应用的一种重要方式.所谓节点的自组织(self-organizing)[4],在此特指在开放的网络环

* Supported by the National Natural Science Foundation for Distinguished Young Scholars of China under Grant No.60625203 (国家自然科学基金杰出青年基金); the National Basic Research Program of China under Grant Nos.2005CB321800, 2005CB321804 (国家重点基础研究发展计划(973))

Received 2007-03-01; Accepted 2007-04-26

© 中国科学院软件研究所 http://www.jos.org.cn

境下,用户节点作为资源的所有者,基于“自愿参与、自主协同”的原则自发地构造系统的行为.在此,我们将节点自组织所形成的软件系统(计算环境)称为自组织系统(计算环境).

自组织系统的一个重要特点是能够聚合大量处于“网络边缘”的计算资源,通过用户节点的自发参与形成新的或更强的计算能力.以SETI@home[5],eMule[6],KazAa[7]等为代表的自组织系统的蓬勃发展说明,在聚合“更多”资源方面,当前自组织系统的发展已经取得了很大的成功.

自组织系统的可用性(availability)反映了其资源聚合的实际效果,直接表现为系统内部可用的资源总量和服务的可靠程度,是其整体服务能力和服务质量的体现.从当前自组织系统发展的需求来看,有更多的用户参与系统,提供资源共享和可靠的服务,这是促进其良性发展的根本途径.然而大量的研究表明,由于缺乏鼓励节点积极参与和诚实服务的有效机制,许多自组织系统面临严重的“搭便车(free-riding)”[8,9]和虚假服务泛滥的问题[10],系统的可用性难以保障,稳定运行和良性发展面临威胁.

信誉机制是提高自组织系统可用性的重要手段之一[11].在信誉机制的研究中我们注意到,由于与人类社会的信任机制具有比较自然的对应,并且能够比较全面地综合与利用节点的历史信任信息,随着基于DHT的结构化P2P拓扑技术的日益成熟,使得在不需要某个集中信任权威的前提上也能有效地实现全局的信誉管理[12],基于推荐的全局信誉模型已经成为信誉构造中的首选.

然而,通过对当前一些具有代表性的全局信誉机制[10,12−14]的分析可知,虽然这些工作都通过不同角度取得了相应的进展,但其信誉评价通常只是作为识别不可靠服务的依据.这种方式能够在一定程度上降低不可靠服务对于系统可用性的影响,但从鼓励节点积极参与合作、贡献资源的角度却不能产生直接的效果.

本文工作的核心是基于信誉评价的激励机制设计,即在对节点进行信誉评价的基础上,根据信誉的高低向其提供差异性的服务,促使节点为了获得更好的服务自发地积极参与合作并提供可靠服务,以达到提高系统整体服务能力和服务质量的目的.

在具体的机制上,我们借鉴了社会系统中组织形成与演化的思想,在改进现有全局信誉机制的基础上提出了一种新的信誉机制——组信誉机制(clique-based reputation system,简称 CRS).

CRS 的一个重要特点是,通过组信誉评价与信誉驱动的组演化相结合,一方面利用组内节点间信誉信息的共享加速了节点间信任关系的建立,增强了信誉模型的节点行为刻画能力,并使其计算过程具有更加优良的收敛特性;另一方面,组演化形成的节点聚集为实现基于信誉的差异化服务奠定了基础,能够达到激励节点积极参与和可靠服务的目的,使得系统的运行在不断趋于稳定的同时,向着比较健康的方向演进和发展.

本文的研究工作是虚拟计算环境(Internet-based virtual computing environment,简称iVCE)可信体系与机制研究[15]的一部分,因此在以下的陈述中,我们对基于虚拟计算环境的体系结构模型[16]的相关概念进行表述.同时,在相关工作的介绍中,对于自主元素(autonomic element,简称AE)和节点两个概念将不加以区分.

本文第 1 节对相关研究工作进行介绍、分析和综述.第 2 节提出组信誉模型.第 3 节介绍组信誉机制的信誉信息管理及其分布求解协议.第 4 节引入信誉驱动的组演化机制.第 5 节对组信誉机制的实现激励机理进行分析和阐述.第 6 节通过仿真实验验证和分析组信誉机制的有效性.最后总结全文.

1 相关研究工作

与本文相关的研究工作主要包括全局信誉模型和激励机制设计两个方面.

1.1 全局信誉模型

信誉是对节点历史行为的评价,反映了节点的行为特征,由于其在一定程度上也可视为对节点间相互依赖关系可靠程度的度量,通常也称为可信度.所谓节点的全局信誉,是指系统内所有节点对于某节点的全局统一的信誉评价,也称为全局可信度.信誉模型主要关注如何利用节点既往的行为信息构造节点的信誉评价.

我们所提出的组信誉模型继承了EigenRep[17]提出并由窦文等人[14]进一步发展的基于推荐的全局信誉构造思想.我们在文献[14]中所提出的信誉模型和分布求解协议的基础上,针对其存在的问题和局限进行了较大的改进.

© 中国科学院软件研究所 http://www.jos.org.cn

EigenRep 和文献[14]所采用的全局信誉模型的核心思想是:当节点 i 需要了解任意节点 k 的全局信誉时,首先,k 的交易伙伴(与 k 发生过交易,或更一般地,使用过 k 所提供服务的节点)获知对 k 服务(例如文件下载)的评价信息,然后,基于这些交易伙伴本身的全局信誉或 i 对于它们(交易伙伴)的局部可信度(EigenRep),综合得到 k 的全局信誉.

在 EigenRep 中,节点的全局可信度定义为

$$T_k = \sum_j (C_{ij} \cdot C_{jk}) \tag{1}$$

其中,j为所有与k发生过交易的节点;C_{ij}为节点i对节点j的局部信任度(或局部信誉评价):

$$R_{ij} = \begin{cases} 0, & \sum_k (S_{ij} - F_{ij}) = 0 \text{ or } S_{ij} - F_{ij} < 0 \\ \dfrac{S_{ij} - F_{ij}}{\sum_k (S_{ij} - F_{ij})}, & \text{else} \end{cases} \tag{2}$$

其中,S_{ij}和F_{ij}分别为节点i对j在历史交易中所积累的满意评价数和不满意评价数.

窦文等人[14]通过分析认为,该模型的主要问题在于:(1) 迭代收敛的前提不合理.为了确保式(1)迭代求解的收敛性及其收敛速度,EigenRep引入了一定比例的“预信任”节点,这些节点“先验地”具有较高且不可更改的信誉.这样的假设一方面缺乏事实的合理性,另一方面也较难在实际操作中实现.(2) 缺乏对不良行为足够的惩罚,可能导致在节点i对j不满程度增加的情况下,对j信任度不降反升的情况发生.(3) 迭代带来巨大的网络开销.EigenRep的迭代求解协议决定了每次交易都会导致全网范围的信誉迭代,其巨大的消息开销($O(n^2)$,n为系统规模)在大型网络环境中缺乏工程上的可行性.

为此,提出了以下改进:

(1) 带有惩罚的节点推荐度(局部信任度),即

$$R_{ij} = \begin{cases} 0, & \sum_k S_{kj} = 0 \text{ or } S_{ij} - F_{ij} < 0 \\ \dfrac{S_{ij} - F_{ij}}{\sum_k S_{kj}}, & \text{else} \end{cases} \tag{3}$$

(2) 基于式(3)定义新的信任关系矩阵$R=|R_{ij}|_n$,将节点i的全局可信度定义为

$$T_i = \sum_k (R_{ki} \cdot T_k) \tag{4}$$

其中,k 为与 i 发生过交易的节点.在证明了式(4)对应信誉方程组的 Jacobi 和 Gauss-Sediel 迭代收敛后,给出了新的分布式求解协议.该协议的消息开销仅为 $O(n)$.

模型(3)、模型(4)较之模型(1)、模型(2)取得了一定的改进,特别是在工程可行性上提出了更为合理的方案.然而分析发现,模型(3)、模型(4)仍然存在一定的问题:

(1) 惩罚过重的问题.节点的信誉很大程度上决定于节点所提供服务的可靠性,因此,其服务成功率常常可以作为区分其特征的一个重要标准.对于行为特征稳定的节点,随着其提供服务次数的增加,服务成功和失败的比例将逐渐趋于稳定.按照式(3)所定义的推荐度,对于服务成功率 50%以下的节点,其所获得的推荐度会逐渐趋于 0,这意味着通过式(4)计算所得的信誉值也会趋于 0.而事实上,网络中可能大量存在这类服务质量不高或服务不稳定的节点,如果就此否定其贡献,显然并不利于达到鼓励节点参与的目的.

(2) 收敛问题.文献[14]虽然证明了其信誉迭代算法收敛,但是并未就如何选取合适的迭代初始值(也即新节点加入系统时所被赋予的默认信誉值)以及迭代的收敛结果进行讨论.事实上,如果在系统初始状态下默认的节点信誉都为 0,则不论系统如何演变,每次迭代的收敛结果仍然是所有节点的信誉为 0.为此,我们尝试考虑一种折衷方案,假设所有节点的初始信誉都为 0.5,按照文献[14]中第 3.3 节的分布求解算法进行迭代,得到的结果见表 1(其中,服务成功率依次为 A:0;B:0.5;C:1,节点比例为 3:3:4,仿真设置参见本文第 6 节).

© 中国科学院软件研究所 http://www.jos.org.cn

Table 1 Average reputation value for different type of AE

表 1 不同类型自主元素的平均信誉值

Cycle	Total transaction number	*A*	*B*	*C*
1 000	13 367	0.0	0.003 426 473 353 276 407	0.003 276 072 777 849 78
2 000	50 133	0.0	2.1882291164211924E−5	2.4236791689596394E−5
3 000	109 812	0.0	8.27538230523169E−8	9.387896699432177E−8
4 000	192 370	0.0	1.675727274320852E−10	2.1031191463403834E−10

虽然迭代的结果仍然能够提供一定的信誉等级差别,在一定程度上能够区分不同类型的节点,但是在以下两方面明显存在问题:(1) 信誉值收敛缓慢,且随着交易次数的增多呈现不断下降的趋势.这一方面使得利用该信誉值进行服务区分存在较大误差;另一方面,信誉值整体上不断降低的趋势意味着不同类型节点信誉评价的绝对差值将会不断减小,不利于节点信誉等级*的设置.(2) 在以上的实验设置中,*A*,*B*,*C* 这 3 类节点具有截然不同的行为特征,然而从迭代的结果可以明显看出,除了 *A* 类的信誉始终为 0 以外,*B* 类和 *C* 类的信誉值之差远小于它们与 *A* 类的信誉之差.这说明了该模型在信誉评价上与节点直观特征的不一致性.

分析造成以上现象的原因,一方面是对迭代初始值设定的忽视(事实上,对于节点的初始值设定通常是一个直觉过程,很难提出与实际情况完全吻合的均一初始值,必须从模型层面进行改进);另一方面,前面提到的惩罚过重问题也是造成节点信誉评价不断下降的原因.基于此,我们认为该模型及其分布式求解协议并不能很好地刻画节点参与系统时所表现的行为特征,需要从模型的角度提出新的更为合理的信誉评价模型.

综上,我们认为作为现有全局信誉模型的代表,以上两类模型分别在工程可行性和信誉刻画的合理性上存在明显的问题.为此,我们在提出新的信誉模型时将结合这两个方面提出改进.

1.2 激励机制设计

自组织系统中的激励机制研究发端于对系统自身特征的认识,即系统中的节点是具有自主特性的行为主体(Agent),其参与系统的目的更多地是为了获取自身利益(例如下载所需的文件)的满足,行为本质上是理性自私的(self-interested).因此,对节点行为的“控制”更多地要依赖于其主观上对于可能获得的利益的判断,而这在很大程度上取决于系统的规则(协议)设置.激励机制[18,19]的目的就是通过一定的交互规则设计调整节点对于自身收益的判断,使其在追求自身利益的同时能够确保系统整体利益的实现.

基于信誉的激励是激励研究的一个重要分支[20].不同于基于交易的激励(例如文献[21,22])通过与交易相关的等价物(例如等量资源或虚拟货币)交换来调节交易双方的直接收益,信誉机制主要基于信誉对节点提供不同服务回报,以此影响节点的行为决策.从这个意义上说,信誉机制可以视为一种间接的激励机制.

从对激励机制的大量研究中发现,Tit-for-Tat(TfT)[23]是一种典型的节点行为策略,即为了获取更多的收益,节点总是倾向于效仿收益较高的节点的行为.BitTorrent协议[24]正是利用这一原理,通过参与同一次文件交换会话节点间的相互回报获得了很高的系统吞吐率.2004 年,Hales[25]在社会驱动的合作(social inspired cooperation,简称SIC)的研究中利用节点分组的思想,通过仿真验证了TfT能够有效驱动一个P2P环境向着整体合作(即绝大多数节点采取系统所期望的合作行为)的方向演进.其基本思想如下:

P2P环境中的节点交互是一种典型的囚徒困境博弈[26],因此从自身利益的角度考虑,节点应该总是优先考虑不合作行为(例如不共享任何资源).然而,如果节点发现存在比自身平均收益更高的节点,则它会对其进行模仿——复制其行为(合作或不合作)和网络连接(拓扑邻居),以期获得更高的个人收益.在存在一定的随机扰动——节点行为和网络连接随机改变——的前提下,实验发现,即使是在最初全部由不合作节点组成的环境中,也只需较少的交互与演化,系统即可达到大多数节点(>99%)都采取合作行为的状态.在SIC中,具有相同行为和网络连接的节点通过在拓扑关系相互连接构成了拓扑上的组关系,共享同一个组标识(tag).组标识本身不表达任何行为特征.

* 设置不同信誉等级是信誉系统中为了简化信誉的使用所采取的一种方案,因为迭代的结果与节点的事实信誉仍然可能存在一定的误差,因此,为了避免影响信誉比较的结果,通常可以通过设置一定的信誉梯度(或可接受误差)来区分不同类型的节点.

© 中国科学院软件研究所 http://www.jos.org.cn

SIC的价值在于证明了TfT机制对于促进系统内部合作行为的有效性,然而,由于其中节点的组变迁决策完全依赖于交易双方平均收益的比较,而收益是节点的私有信息,在实际运行的系统中通常不可能直接获得,因此,其促进合作作用事实上是在放弃节点部分自主性的前提下实现的,使得其实际应用价值大为降低.

我们通过分析认为,在存在基于信誉的差异性服务机制的前提下,信誉事实上可以视为对自主元素所能获得的服务回报的预期(信誉越高,所能获得的可用服务越多,服务质量也越高).如果将组标识与一定的信誉评价相绑定,则通过信誉的比较能够有效驱动以上演化过程的进行,从而使该机制得以自发地产生作用(self-enforcing),而这也正是我们提出基于组信誉构造新的激励机制的基本出发点.

2 组信誉模型(clique reputation model,简称 CRep)

2.1 组拓扑结构

我们首先引入自主元素组和组拓扑的概念.

定义 1(自主元素组). 组(或自主元素组,clique)定义为共享同一“组标号”(tag)的自主元素的集合.Tag 是组的全局唯一标识.组内的自主元素称为组成员或成员元素.

从结构上看,自主元素组对应于一定的拓扑结构——组拓扑.组拓扑可以看作是在底层网络(例如基于DHT的结构化拓扑)上架构的一层新的层叠网络(overlay network)[2].

定义 2(组邻居关系). 组邻居关系是组拓扑中自主元素间的一种关联关系,自主元素 i 和 j 互为邻居是指在它们各自的组拓扑路由表中都保存着对方的逻辑地址(即自主元素对应节点在底层网络拓扑中的地址).

为了便于陈述,在以下未作特别说明的情况下,“邻居”一词均指组拓扑中的邻居关系.

定义 3(组拓扑). 组拓扑是自主元素组对应的拓扑结构,它满足如下基本条件:对任一自主元素 i,

i) i 与其所有同组的自主元素互为邻居;

ii) i 至少与一个非同组的自主元素保持邻居关系.

显然,由以上定义,在组拓扑中,处于同组的自主元素之间为全连接的关系,即每个自主元素都是同组其他自主元素的邻居.同时,每个组成员至少与某个组外的自主元素保持邻居关系(如图 1 所示).这样的结构决定了组拓扑具有一种“内紧外松”的特点,即组成员间是一种紧耦合的彼此紧密连接的关系,而组间则只是通过跨组的稀疏连接构成一定的邻居关系.显然,组的规模取决于自主元素所能拥有的组邻居上限,这是由具体的系统设计所决定的.如果设组邻居的上限为 d,则组的规模最大也为 d.

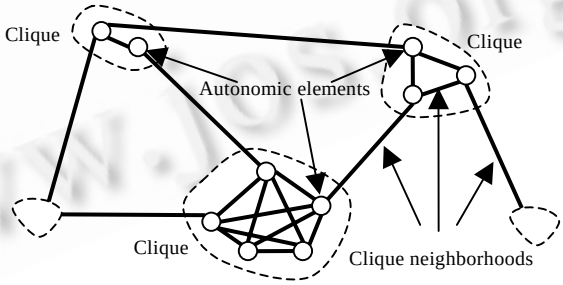


Fig.1 Clique topology

图 1 组拓扑结构

记号约定. 为了方便陈述,我们约定以下记号:组通常记为$C_A,C_B,\ldots$,记$CID_A,CID_B,\ldots$为组的标号(tag);自主元素用英文小写字母(例如$i,j,\ldots$)表示,其标识为$ID_i,ID_j,\ldots$;设自主元素i所在组为C_A,记$tag(i)=A$,对应地,C_A也可以表示为$C_{tag(i)}$;d为组拓扑中自主元素的最大邻居数;N_A为组C_A当前的成员数;T_A为组C_A的信誉.

2.2 组的全局信誉

定义 4(自主元素间服务评价). 自主元素i对自主元素j的服务评价记为E_{ij},取值范围是{(1,0),(0,1)},其中,

© 中国科学院软件研究所 http://www.jos.org.cn

(1,0)表示i对j的服务满意,(0,1)为不满意.

服务评价是在自主元素间发生了交易行为(即相互提供服务,例如文件下载、内存共享等)后,由使用服务的自主元素提交的.在现有的信誉模型中,服务评价都是以节点(自主元素)对节点(自主元素)的方式提交的.而在CRep 中,服务评价以组对自主元素(节点)的方式提交.即在评价信息中,服务评价者不再是某个自主元素,而是其当前所在的组**.

定义 5(组间组服务评价). 组C_A对C_B($A\neq B$)的服务评价P_{AB}是其对于在一段时间内***对C_B所提供服务的总评,

$$P_{AB}=\begin{cases}\dfrac{S_{AB}}{S_{AB}+F_{AB}}, & S_{AB}+F_{AB}\neq 0\\ 0, & \text{else}\end{cases} \tag{5}$$

其中,S_{AB}为C_A对C_B服务****的满意评价次数,F_{AB}为不满意评价的次数.$S_{AB}=\sum_{l\in C_B}SN_{Al}$,$F_{AB}=\sum_{l\in C_B}FN_{Al}$,$SN_{Al}$,$FN_{Bl}$分别为$C_A$对$C_B$中某自主元素$l$的服务满意和不满意评价次数.

定义 6(服务推荐). 组C_A对组C_B的服务推荐定义为

$$R_{AB}=P_{AB}\cdot\beta^{\frac{1}{S_{AB}+F_{AB}}} \tag{6}$$

其中,$0<\beta<1$ 为固定常数.R_{AB}也称为C_A对C_B的直接信任度.

类似地,可以定义组C_A对某自主元素i的推荐度(直接信任度)为

$$RN_{Al}=PN_{Al}\cdot\beta^{\frac{1}{SN_{Al}+FN_{Al}}},$$

其中,

$$PN_{Al}=\begin{cases}\dfrac{SN_{Al}}{SN_{Al}+FN_{Al}}, & SN_{Al}+FN_{Al}\neq 0\\ 0, & \text{else}\end{cases}.$$

在推荐度中,引入调节因子$\beta^{\frac{1}{S_{AB}+F_{AB}}}$的原因是为了尽可能保证推荐度的合理性,使之更好地反映自主元素实际行为的差异.例如,对于任意两个组C_U,C_V,设它们共交易 10 000 次,成功 5 000 次,而C_V与C_W交易仅 2 次,成功 1 次,则$R_{UV}\approx P_{UV}$,而$R_{WV}=P_{WV}\cdot\sqrt{\beta}$,这意味着$C_U$的推荐将更有价值,而$C_W$的推荐由于次数少而有所折扣.

定义 7(组全局信誉). 自主元素组C_B的全局信誉为

$$T_B=\begin{cases}\sum\limits_{A\in L(B)}\gamma^{T_A}\cdot R_{AB}\Big/\sum\limits_{A\in L(B)}\gamma^{T_A}, & L(B)\neq\varnothing\\ 0, & \text{else}\end{cases} \tag{7}$$

其中,$\gamma>2$ 为常数(例如$\gamma=e$),$L(B)$为所有使用过C_B中自主元素服务的组的集合.

以上定义既能保证一定的可信度等级,又能够避免奇异性,而且使得权值$\sum\limits_{A\in L(B)}\gamma^{T_A}\cdot R_{AB}\Big/\sum\limits_{A\in L(B)}\gamma^{T_A}$的使用显得“比较民主”.事实上,全局信誉$T_B$虽然仍旧依赖于相关交易组的全局信誉$T_A$及其推荐$R_{AB}$,但在$T_A$完全为 0 时,又不彻底抹杀其局部交易成功率(推荐度)R_{AB},而是给它一个最小的权$R_{AB}\Big/\sum\limits_{A\in L(B)}\gamma^{T_A}$.此外,$\gamma>2$ 保证了γ^T是关于 T 的增函数,因而该定义依然保留了合理的信誉等级.

式(7)事实上定义了一个关于组信誉的线性方程组,利用 Brouwer 不动点定理可以证明其解存在.

定理 1. 由式(7)定义的组全局信誉方程组的解存在.

限于篇幅,在此省略详细证明过程.

** 对于自主元素可能冒充其他组成员提交服务评价的情况,本文第 5 节将提出相应的保证机制,确保不会出现此类不良行为.

*** 通常可以采用一个固定的时间τ,τ的值视具体应用或其中所涉及的交易频度而定.

**** C_B为C_A提供的服务,即C_B中成员元素为C_A中成员元素所提供的服务.

© 中国科学院软件研究所 http://www.jos.org.cn

在 CRep 中,由于没有显式地对每个自主元素进行信誉评价,因此在需要利用信誉作为服务决策的依据时,组信誉即等同于自主元素的信誉.

在 CRep 中,基于自主元素间相互评价构造的自主元素信誉被推广为基于组对自主元素的服务评价构造的组信誉.这使同组自主元素得以共享所获得的服务评价信息,从而加速自主元素信誉积累的过程,一方面有利于系统较快地识别自主元素的行为特征,另一方面也有利于自主元素更好地获得与其表现相匹配的服务,使系统在演化过程中较快地趋向稳定.

为了简化陈述,如无特别说明,本文以下所述“组信誉”均指组全局信誉.此外,在本文中,为了与组信誉机制有所区别,我们将传统的基于节点(自主元素)对节点(自主元素)服务评价的信誉模型统称为个体信誉模型.

3 组信誉的分布求解

组信誉的分布求解协议包含信誉信息的分布存储和分布求解算法两部分.

3.1 信誉信息的分布存储

在自组织虚拟计算环境下,自主元素的行为本质上是不可控的,在没有集中信誉管理设施的前提下,不可能将自主元素的信誉值、服务评价等信息直接存储在其所在的节点上,否则将带来难以避免的安全问题[27].为此,我们基于文献[28]提出的Terrace拓扑设计了面向CRep的信誉信息分布存储机制.

Terrace拓扑是一种基于分布式Hash(DHT)技术[29,30]的结构化拓扑,在此,我们将其作为组拓扑的底层支撑.利用Terrace所具有的地址冗余容错能力,可以确保信誉信息不会因为个别Terrace拓扑节点的失效或退出而受到损失.

具体来说,对应于CRep的信誉模型,信誉信息的存放以组为单位,即对于任意的组C_A,我们通过一个均匀的Hash函数(例如SHA-1)HTD,将其标识CID_A投影到Terrace中的某个节点逻辑地址CD_A上,记为$CD_A=HTD(CID_A)$,CD_A所对应的Terrace节点称为组C_A的档案点.在组C_A的档案点上存储和维护的信誉相关信息包括组标识、当前组信誉值以及其他组对于该组内成员的服务评价等.一般来说,每个档案点至少应该包含一个如表 2 所示的数据结构.

Table 2 Data structure on CD_A

表 2 CD_A的数据结构

CID_A		$T_A^{(k+1)}$	Sec_A
ID_{i_1}	ID_{i_2}	…	CID_{im}
S_{K_1A}	F_{K_1A}	$T_{K_1}^{(k)}$	CID_{K_1}
$SN_{K_1i_1}$	$FN_{K_1i_1}$		
$SN_{K_1i_2}$	$FN_{K_1i_2}$		
…	…		
…	…	…	…
S_{K_nA}	F_{K_nA}	$T_{K_n}^{(k)}$	CID_{K_n}
$SN_{K_ni_1}$	$FN_{K_ni_1}$		
$SN_{K_ni_2}$	$FN_{K_ni_2}$		
…	…		

表 2 中,CID_A为组C_A的标识;$T_A^{(k+1)}$为由CD_A计算出的组C_A当前的全局信誉值;$(ID_{i_1},ID_{i_2},\ldots,CID_{im})$为$C_A$的成员元素的标识;$(CID_{K_1},\ldots,CID_{K_n})$为所有使用过$C_A$中成员元素服务的组的标识,$(T_{K_1}^{(k)},\ldots,T_{K_n}^{(k)})$为其各自对应的当前全局信誉值;$(SN_{K_1i_1},SN_{K_1i_2},\ldots,SN_{K_ni_1},SN_{K_ni_2},\ldots)$和$(FN_{K_1i_1},FN_{K_1i_2},\ldots,FN_{K_ni_1},FN_{K_ni_2},\ldots)$分别为这些组对与$C_A$中对应的成员的服务评价;$(S_{K_1A},\ldots,S_{K_nA})$和$(F_{K_1A},\ldots,F_{K_nA})$分别为这些组对$C_A$的总的服务评价信息.$Sec_A$为组$C_A$相关的安全机制信息,例如其公、私钥对.

© 中国科学院软件研究所 http://www.jos.org.cn

3.2 分布式求解协议

首先给出协议的几个原语及其语义:

Put(ID_i,CID_A,E_{Ai},f):将组C_A对自主元素i的评价E_{Ai}写入Terrace树中逻辑地址为$HTD(CID_{tag(i)})$的档案点(根据评价的结果同时更新SN_{Ai},FN_{Ai}和$S_{A,tag(i)}$,$F_{A,tag(i)}$),并触发该节点上的f过程.

Get(CID_B,T_B):从Terrace逻辑地址为$HTD(CID_B)$的档案点中读取组C_B的当前信誉并写入本地变量T_B.

Eval-Trans(ID_i):自主元素每次与其他自主元素交易(使用其服务)完毕,通过该原语提交服务评价,*Eval-Trans*(ID_i)返回true表示对服务满意,否则表示不满意.

在该求解协议中,任意自主元素 i 可能同时具有两个角色,既可能是服务使用者(评价提交者),也可能是某(几)个组的档案点.

作为服务使用者,自主元素i在使用了自主元素j的服务以后,将代表其所在组$C_{tag(i)}$对自主元素j的服务进行评价,并进而触发对于j所在组$C_{tag(j)}$的信誉更新,算法如下:

Procedure *Eval*(ID_j)
  if (*Eval-Trans*(ID_j)=true) then
    $SN_{tag(i),j} \leftarrow SN_{tag(i),j}+1$
    $S_{tag(i),tag(j)} \leftarrow S_{tag(i),tag(j)}+1$
  else
    $FN_{tag(i),j} \leftarrow FN_{tag(i),j}+1$
    $F_{tag(i),tag(j)} \leftarrow F_{tag(i),tag(j)}+1$
  endif
  Put(ID_j,$CID_{tag(i)}$,$E_{tag(i),j}$,*UpdateRep*);
End

其中,过程UpdateRep将在$C_{tag(j)}$的档案点上被触发执行,算法如下:

Procedure *UpdateRep*($CID_{tag(j)}$)
  $tmpSum \leftarrow 0$; $tmpT \leftarrow 0$;
  for (any B<>$tag(j)$)
    if ($S_{B,tag(j)}+F_{B,tag(j)}$<>0) and ($N_B$>0) then
      Get(CID_B,T_B);
      利用公式(5)、公式(6)计算$R_{B,tag(j)}$;
      $tmpSum \leftarrow tmpSum+\gamma^{T_B}$;
      $tmpT \leftarrow tmpT+\gamma^{T_B} \cdot R_{B,tag(j)}$;
    elseif (N_B=0)
    //成员数为 0 意味着该组事实上已经不存在
      将与C_B有关的服务评价删除;
    endif
  endfor
  $T_{tag(j)}=tmpT/tmpSum$;
End

显然,以上算法每次信誉计算的消息复杂度为$O(n)$,这与EigenRep所使用的全局迭代求解方法(消息复杂度为$O(n^2)$)相比,在效率上有了很大的提高.同时,考虑到随着系统的不断演化,组的平均规模会有所增长,造成系统内部组数量与自主元素数量的差距(例如在第 6 节的仿真中,当n/d=10 时,系统内部的组数量约为自主元素总数的 1/3).因此,在实际的计算代价上,以上的分布求解算法不但比EigenRep所使用的全局迭代方法有大幅度降低,也优于采用类似算法的个体信誉模型(例如,文献[14]中的分布求解算法和PeerTrust[12]的

© 中国科学院软件研究所 http://www.jos.org.cn

ComputeTrust_TVM/ATC算法).

综上,组信誉模型CRep及其分布求解协议在模型的收敛性和分布求解的效率上都比现有工作[10,12]有了明显的改进,同时修正了文献[14]的改进所存在的问题.如果在CRep中假设所有自主元素都是独立成组(不与其他自主元素同组),显然,其“退化”得到的个体信誉模型也具有以上特点.

4 信誉驱动的组演化

组的演化是与系统运行过程中自主元素间的交易行为同步发生的.随着系统内部交易的不断发生,组信誉会不断发生变化,自主元素为了寻求更好的信誉评价,会申请加入信誉更高的组;与此同时,组本身为了维持其内部信任的可靠性和较好的外部信誉评价(即组的全局信誉),也将进行优胜劣汰式的成员淘汰与更新,这都会导致组拓扑本身结构和外部连接关系的不断变化.

4.1 组成员的迁移

由于信誉较高组的成员通常能够获得较好的服务,因此,自主元素总是倾向于加入信誉较高的组.组成员的加入采用申请机制,任何一个组都只会接纳向其提供过成功服务的自主元素的加入.因此,通常在发生一次成功的交易后,如果提供服务的自主元素一方信誉较低,则其会向另一方提出加入组的申请.

例如,自主元素$i(i\in C_A$,组信誉$T_A)$申请加入自主元素$j(j\in C_B$,组信誉$T_B)$所在的组,如果以下条件满足,则可以接纳(如图 2 所示):

① $N_B<d$,即C_B中的成员未满.

② $T_B-T_A<\varepsilon$,其中,$0<\varepsilon<1$ 为常数,我们称其为信誉等级跨度.该条件表示i原属组的信誉处于一个与C_B的信誉差距相当的范围,从而避免了信誉过低的自主元素直接进入高信誉的组,导致组信誉的大幅波动.

③ $SN_{Bi}>\max(N_B/3,3)$且$RN_{Bi}>T_B$,第 1 部分的条件意味着申请者i与C_B之间具有一定的成功交易基础;第 2 部分表示C_B对i的直接信任应该超过其本身的全局信誉,这样做的好处是使得i的加入不但能够提高i自身的信誉评价,同时也可能对C_B的信誉提升有所帮助.

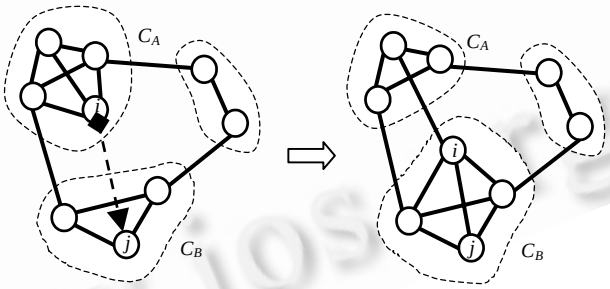


Fig.2 After provided successful service for j, i migrates from C_A to C_B

图 2 i向j提供成功服务后,由组C_A迁移到组C_B

申请及接纳的过程如下:

Step 1. i为j提供了成功的交易后,判断是否满足条件②,如果满足,则再向C_B的档案点CD_B申请加入该组;

Step 2. CD_B判断i是否满足接纳条件,如果满足,则继续;否则,结束;

Step 3. i除了保持与组C_A内某一随机成员的邻居关系以外,放弃与其中其他成员的邻居关系.CD_B通知C_B内的所有成员接纳i;

Step 4. i与组C_B内的所有自主元素相互建立邻居关系;

Step 5. C_A的档案点CD_A将与i有关的信誉评价信息(($SN_{K_1i},\ldots,SN_{K_ni}$)和($FN_{K_1i},\ldots,FN_{K_ni}$))发送到$C_B$的档案点$CD_B$,$CD_A$,$CD_B$更新各自的评价信息,并重新计算$C_A$,$C_B$的组信誉.

由于组的规模受到自主元素在组拓扑中的最大邻居数(d)的限制,故容易知道自主元素在组间迁移的消息

© 中国科学院软件研究所 http://www.jos.org.cn

代价为 $O(d)$.

4.2 开除组成员

当某个组成员多次造成内部交易失败,或其整体表现已经低于一定的可容忍范围时,可能导致其被组开除.

(i) 交易失败导致的开除

交易失败导致的开除是组织自身为了维护内部信任关系所采取的一种“自我清洁”行为,当发生了组内不良交易时,将会触发该过程.具体来说,若自主元素i在组C_A内造成了交易失败(例如提供虚假的服务),此时,将由请求服务的自主元素触发对i成员资格的审查,如果满足$FN_{Ai}>N_A/m$,即i在C_A内造成的失败交易次数超过了与组成员数相关的某个阈值(在仿真中我们取m=3),则i将被该组开除.

具体的开除过程如下:

Step 1. i 在与 j 的组内交易中造成了交易失败,j 在更新该组对 i 的评价的同时,检查是否满足对 i 的开除条件,如果不满足,则结束;否则,继续;

Step 2. 组内其他成员解除与 i 的邻居关系;

Step 3. i 独立创建一个新的组;

Step 4. C_A的档案点D_A将与i有关的评价信息迁移到i新的档案点$D_{tag(i)}=HTD(tag(i))$,并更新自身的评价信息,然后,D_A和$D_{tag(i)}$分别重新计算C_A,$C_{tag(i)}$的组信誉.

理论上,该机制存在自主元素间相互诋毁的可能,即自主元素通过伪造不成功的交易对其他自主元素进行诽谤,进而导致其被组所开除.虽然这种可能性确实存在,但是由于从自主元素本身利益出发,开除其他自主元素并不能使其自身获得任何现实的利益.相反地,如果开除的是一个对外来说行为良好的自主元素,其直接影响将会是组本身信誉提升速度的降低,进而影响该自主元素自身的利益.从这个意义上说,由于组本身可以视为由多个自主元素所组成的利益共同体(虚拟共同体),因此该现象很难存在发生的先决条件.

(ii) 整体表现不佳导致被开除

与交易失败导致的开除不同,自主元素因整体表现不佳被开除,主要是可以避免自主元素加入组后仅仅依赖他人贡献积累信誉而不贡献或很少贡献的情况发生.具体来说,由系统设定一个检查周期(在第 6 节的仿真中,我们设置为 100 个迭代周期),每当检查时刻到来,将由组档案点触发一次组内部的信任度检查,判断是否有达到最低容忍限度的成员元素,如果有,则通知组成员将其开除.具体来说,导致自主元素i被组C_A开除的条件如下:

$$\sum_{B\in L(i)} SN_{Bi} < \phi(N_A)\cdot \sum_{B\in L(A)} S_{BA} \text{ 或 } \sum_{B\in L(i)} FN_{Bi} > \psi(N_A)\cdot \sum_{B\in L(A)} F_{BA},$$

其中,$\phi(N_A)$是与组的规模相关的函数,通常情况下,可以取$\phi(N_A)=1/(N_A)^2$, $\psi(N_A)=1/(N_A)^{\frac{1}{2}}$.

该情况下具体的开除过程除了将情况(i)中的 Step 1 替换为档案点检查开除条件以外,其余都与(i)相同.

注意到可能存在档案点恶意开除自主元素的情况,我们可以通过第 3.1 节提出的基于 Terrace 拓扑的信誉信息存储进行相应的设置.具体来说,利用 Terrace 拓扑本身具有的地址冗余功能,即每个档案点都由多个相互独立的备份档案点组成,当某个档案点(通常情况下,只需某个主档案点进行开除条件判断)产生了开除决策时,必须要求其他备份档案点重新确认,如果结论一致则实施开除,否则撤销开除决定.显然,不可能总是发生多个档案点联合作弊的情况,因此,以上方法将可以有效地避免该问题的发生.

4.3 组的创建

在默认情况下,任何自主元素都可以创建组,但是对于新创建的组,通常分为如下两种情况:

(i) 对于新加入系统的自主元素,将默认创建一个新组,并设置其初始的信誉值为 0;

(ii) 对于被组开除的自主元素,在为其创建一个新的组后,将根据环境中各组对该自主元素的服务评价重新进行计算,得到新的组信誉值.

4.4 组的退出

对于任何一个组,当其中最后一个成员离开后,该组事实上就不存在了.在系统的实际运行过程中,我们并

© 中国科学院软件研究所 http://www.jos.org.cn

不提供显式的组注销操作,而是利用不断进行的信誉更新来逐步检查和“擦除”与该组有关的记录.事实上,注意到在第 3.2 节的UpdateRep过程中,我们通过加入(N_B>0)的判断对于该组是否存在进行了检验,如果该组事实上已经不存在,则会在执行更新过程的档案点中消除其所提供的所有服务评价,其效果等于在事实上注销了该组.

5 基于组拓扑演化的激励机制

在 CRS 中,通过组的创建、组自主元素的加入和退出以及组本身的退出机制,能够使自主元素根据自身的行为特征获得与之相匹配的网络地位.也即随着组的不断演化,一方面服务质量稳定和贡献较多的自主元素能够通过相互成功交易次数的增长而加入同一组织,形成相对稳定的自主元素聚集,从而通过相互邻居关系的建立确保各自能够更加方便地获得比较稳定和高质量的服务;另一方面,表现较差的自主元素则往往难以和其他行为较好的自主元素建立邻居关系,而是更多地以相对孤立和不稳定的组织形式存在,这意味着其所能获得的服务质量和稳定性较之行为良好的自主元素都存在一定的差距.因此,通过与组信誉评价的共同作用,组的演化机制事实上构成了对自主元素参与和贡献行为的激励,通过组演化机制所形成的良好行为,自主元素的聚集及其稳定发展将会对自组织虚拟计算本身的健康发展产生直接的促进.

我们将基于组拓扑演化的激励机制概括为表 3,其中,自主元素的行为对应于其在一次服务中所提供的服务,Good 表示其所提供的服务是可靠的,获得了满意的服务评价,Bad 则反之.

Table 3 Incentive mechanism based on clique topology evolution

表 3 基于组拓扑演化的激励机制

Feedback	Behaviors of AE	
	Good	Bad
In clique	Volunteer	Punish AE
Out clique	Award clique and AE	Punish clique

具体来说,对于自主元素行为的反馈,根据其发生在组内和跨组的不同可分为两类,根据其服务效果的“好”、“坏”又可进一步产生不同的影响:

(1) 组内部“好”的行为→义务

在人类社会的组织中,同一组织内部的成员间可以理解为是一种绝对信任的关系,即其信任关系的建立并不完全依赖于其过往的交易行为,而是基于其相同组织身份的判别,即一旦认定对方是和自己处于同一组织的成员,则信任对方.CRS 的设计思想也体现了这一原则,即认为组内的自主元素之间应该是相互绝对信任的,在此基础上,默认组内的节点有“义务”彼此提供可信的服务.

(2) 组内部“不好”的行为→惩罚个体

如果一个自主元素在组织内部多次有不好的行为,则意味着对其信任的错误,因此需要考虑将其及时清除出组.如果某个自主元素在组内多次获得不满意的评价,或该成员的整体表现已经低于一定的可容忍范围,则可能导致其被组开除(参见第 4.2 节).自主元素一旦被组开除将对其信誉进行重新计算.在我们的设置下,如果自主元素被组开除,则直接意味着其信誉评价的大幅度下降(参见第 6.5 节),显然,这是对其个体的一种有效的惩罚.

(3) 组织间“好”的行为→奖励整体与个体

在人类社会中,个人跨组织的交互行为往往可以被视为组织的行为,并引起对组织评价的改变.如果自主元素对非同组的自主元素提供了成功的服务,并收到满意的评价,则一方面,根据组信誉模型,可能直接导致组本身的信誉提高,并在一定程度上使该自主元素得以获得更好的服务反馈;另一方面,对信誉更高的组成员的成功服务将使其获得加入该组的可能,从而进一步获得更好的组拓扑地位.

(4) 组织间“不好”的行为→惩罚整体

如果自主元素在跨组织的交易中的服务不成功,其直接后果只是造成其所在组的信誉的降低.然而,由于这也会间接影响到该自主元素及其组内邻居所能获得的服务反馈,因此,大量此类不良行为将直接导致其他组成员为了追求更好的服务地位而离开,从长远来看,将造成组本身的不稳定,最终使得不良个体被孤立,因此也能达到间接惩罚的目的.此外,组内部定期的检查机制决定了,如果某个自主元素的行为与其邻居存在过大的差

© 中国科学院软件研究所 http://www.jos.org.cn

异,则很可能会被组自动清除.

综上,由于组可以视为自主元素所构成的利益共同体(虚拟共同体[16]),其中,自主元素的任何行为所产生的组间信誉评价的变化或内部信任关系的改变都会对该自主元素本身产生影响.在CRS中,组内的良好交易行为虽然并不意味着更好的个体信誉评价,但是,其不良行为却可能导致惩罚的产生.而自主元素一旦被开除,根据组的演化机制,其内部的不良行为将会被“暴露”,进而导致其信誉大幅度降低,造成其难以被其他组所接纳,而在事实上成为孤立的自主元素,这无疑将对其信誉的积累产生直接的影响.另一方面,如果不良自主元素采取更为“狡猾”的策略,总是对外采取不良行为,而对内却只是提供可靠的服务,则可能在一定程度上造成组信誉度的整体下降,其直接后果是其他行为良好的组成员不断离开该组,也会使得该不良个体最终被孤立,第 6.5 节对于此类节点恶意行为的仿真验证了这一结论.

6 验证与分析

6.1 仿真设计

我们采用离散事件仿真的方法,基于一个 P2P 交易环境对 CRS 的效果进行评价和验证.在仿真环境中,所有的自主元素按照一定的概率随机发生交易,交易成功与否取决于该自主元素所提供的服务的可靠性.为了更好地反映信誉模型刻画自主元素行为特征的能力,我们对影响自主元素信誉的一些客观因素进行了简化,假设所有自主元素都可以相互提供服务,而服务的可靠性完全取决于用户的主观参与意愿.为此,我们根据自主元素的服务可靠性,引入 3 类不同的自主元素:*A* 类为不合作元素,其所提供的服务总是不可靠的,服务成功的概率为 0;*B* 类为不稳定元素,服务成功的概率为 50%;*C* 类为可靠元素,其对外总是提供成功的服务,服务成功概率为 100%.

在比较机制的选择上,鉴于此前所述[14]改进在信誉收敛上存在的问题,我们选取EigenRep模型作为参照.为了验证引入组机制对于信誉模型收敛性的改善,我们也将此前提到的CRep退化得到的个体誉模型作为一个比较对象.我们认为,引入组机制前后,模型的表现差异对于其他全局信誉模型应该也是同样适用的.

我们从以下几个方面考察 CRS 的作用:

(1) 信誉模型的比较.与个体信誉模型对照,考察组信誉模型对自主元素行为特征的刻画能力及其算法扩展性方面的差异.包括:

- 对不同行为特征自主元素的识别能力.通过与 EigenRep 模型以及 CRep 退化得到的个体信誉模型比较,反映 CRep 模型在刻画自主元素行为特征能力上的改进,具体包括信誉收敛值的合理性、信誉收敛的速度.
- 消息代价的比较.由于EigenRep的求解算法复杂度为$O(N^2)$,因此,主要与CRep退化得到的个体信誉模型比较,考察在不同的系统规模和组规模下,信誉分布求解的消息代价差异.

(2) 组拓扑演化.主要演示和考察组演化机制的合理性.按照 CRS 的设计,随着系统的运行,行为表现良好的自主元素将逐渐形成比较稳定的组邻居关系,而对于行为表现较差的自主元素,其在组拓扑中总是相对孤立的.

(3) 基于组拓扑演化的激励.主要是通过不同类型的自主元素随着系统运行而形成的拓扑地位的不同,反映不同行为特征自主元素所能获得的服务预期的差异.

(4) 针对 CRS 的恶意攻击行为.主要考察自主元素通过欺诈手段进入信誉较高的组后,改变行为提供不可靠服务对于 CRS 的影响.

运行环境为 P4 2.4GHz,1GB 内存的 Windows2000 工作站,仿真程序利用 Java 语言编写.对于可重复的实验,基于不同的随机因子重复运行 5~10 遍,取其平均值作为评价的依据.基本参数设置见表 4.

© 中国科学院软件研究所 http://www.jos.org.cn

Table 4 Simulation environment configuration

表 4 仿真环境设置

Parameters type	Parameters name	Value/Range
Topology	System scale (N)	100~500
	Max neighborhoods (d)	2~20
	Initial topology	Random, no clique
Event	Max cycle	1 000~6 000
	Simulation type	Event driven
	Max events within each cycle	N/10
Evolution	Reputation degree span (ε)	0.05
	Self-Inspection period	100 cycles
Reputation model	Recommendation factor (β)	1/e
	Reputation iterate factor (γ)	e

6.2 信誉模型的比较

6.2.1 对不同行为特征自主元素的识别能力

设置 3 类自主元素在环境中所占的比例分别为 A:B:C=3:3:4.比较的对象为 CRep 模型、EigenRep 模型(为了保证其算法的收敛性,设置 10%的信誉始终为 1 的自主元素为预信任节点)和 CRep-AE 模型(如图 3 所示).

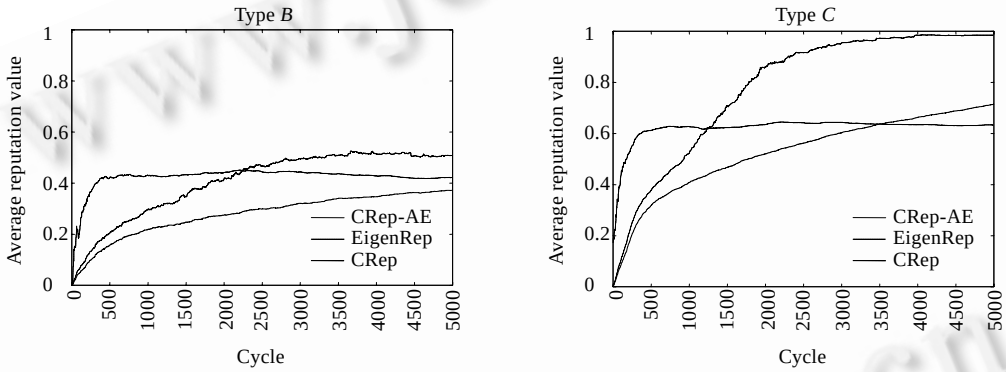


Fig.3 Reputation curve for different type of AEs

图 3 不同特征自主元素的信誉曲线

由图 3 可以看出:

(1) 两类自主元素最终的 CRep 和 CRep-AE 信誉值基本上与其服务成功概率相吻合,这说明基于组信誉能够有效地区分自主元素在服务可靠性上的差异.相对而言,EigenRep 模型在两类自主元素差异的刻画上较为模糊.此外,考虑到 A 类元素信誉始终为 0 这一情况,可以发现 3 类自主元素间的 EigenRep 信誉差异相对不够一致(见表 5).

Table 5 Average reputation value for different type of AEs under three reputation models

表 5 不同类型自主元素在 3 种信誉模型下的平均信誉值

Average reputation value	A	B	C
Node-Based CRep	0.0	0.371 628 873 423 850 5	0.713 899 884 653 094 4
EigenRep	0.0	0.421 476 537 792 971 34	0.633 040 425 121 938 4
CRep	0.0	0.508 042 073 199 528 5	0.985 211 134 027 276 7

(2) 总体来看,自主元素的组信誉值总是高于其个体信誉值,事实上,通过 10 次采用不同随机因子的重复迭代发现,所有自主元素的 CRep-AE 信誉平均为 0.397 0486 159,CRep 信誉平均为 0.546 497 075 6.3 类自主元素的信誉平均值对比见表 5.

(3) 从信誉值随时间变化的情况来看,CRep 信誉趋向稳定的过程总体上比 CRep-AE 更加迅速,事实上,就 C 类自主元素而言,其直观的信誉值应该为 1,CRep 信誉在第 3 000 个迭代周期(此时,每个自主元素平均的服务次

© 中国科学院软件研究所 http://www.jos.org.cn

数约为 124)就已经基本稳定,但 CRep-AE 信誉直至迭代结束仍与直观信誉有一定的差距.这验证了组演化机制在加速信誉收敛速度上的作用.

通过比较我们认为,组信誉模型不仅能够有效地识别自主元素服务能力的差异,提供比较合理的信誉评价,同时也有利于自主元素更快地积累信誉.对于自主元素来说,由于加入组能够获得更快速的信誉积累,因此在 CRS 的作用下,自主元素总是会自发地形成组.这也可以说明我们所提出的组演化机制与 SIC 相比,其改进(参见第 1.2 节)的合理性.

6.2.2 消息代价的比较

对不同的最大组规模(d)信誉迭代所涉及的消息代价进行比较.在此,消息代价是指在信誉维护过程中,所有与信誉信息相关的查询和写入操作所引起的消息传输总数.其中,对于个体信誉模型而言,主要包含服务评价的写入和信誉更新相关信誉查询(例如第 3.2 节的Get(ID_B,T_B)原语)的消息代价;对于组信誉模型,除了以上两者以外,还包含与组演化有关的所有服务评价迁移和信誉更新的消息代价.由于EigenRep模型采用了全局迭代的信誉求解算法,其消息代价为$O(N^2)$,故在此只与CRep退化得到的个体信誉模型进行比较,其具体算法与文献[14]的分布求解算法相同.

由图 4(a)可以看出,随着自主元素邻居数的增加,在相同的系统规模(100)和迭代周期(2 000)下,个体信誉模型的平均消息代价基本维持不变,而组信誉模型的平均消息传输代价虽然在 d 较小时高于个体信誉模型*****,但是随着 d 的增大,其总体上呈现不断下降的趋势.因为随着组平均规模的不断增长,大量发生在组内部的交易评价不再被用于进行组间的信誉迭代,相当于减小了信誉模型所涉及的迭代规模.

另一方面(如图 4(b)所示),对于确定的最大组规模(d=10),随着系统(N=200)的不断演化以及组的不断成长,组信誉模型与个体信誉模型的自主元素信誉维护的平均消息代价之差呈现不断加大的趋势.同时可以看到,组信誉模型总的平均消息代价基本上是随着迭代次数的增长而近似线性增长的,这反映出组信誉模型在稳定状态下具有更好的可扩展性,在一定程度上减轻了全局信誉模型信誉求解固有的消息代价较高的问题.

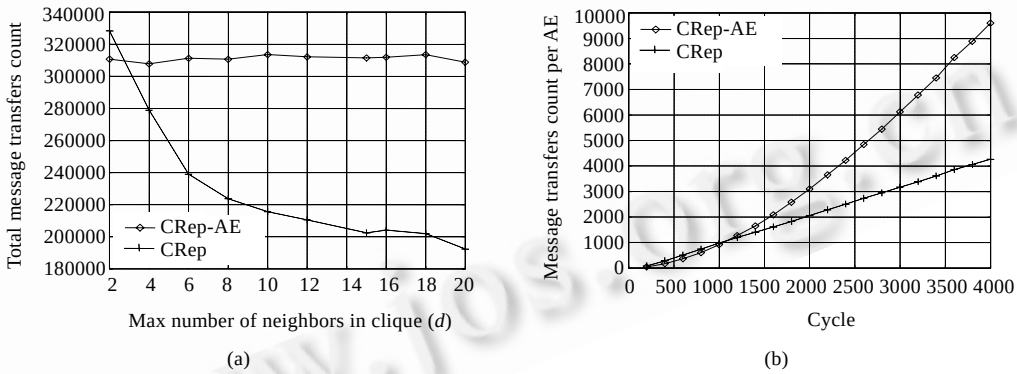


Fig.4 Number of messages for CRep solving under clique-based model and AE-based model

图 4 基于组和基于个体的 CRep 模型求解的消息数

6.3 组拓扑的演化

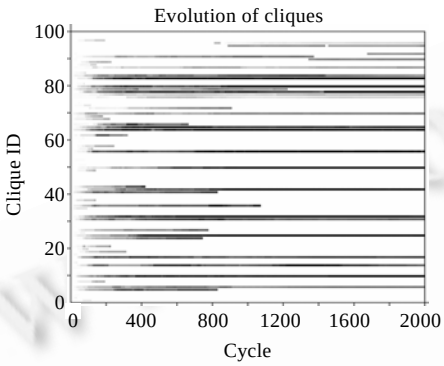
在 CRS 中,组拓扑演化是实现对自主元素行为激励的途径.随着自主元素间交易的不断发生,组信誉也在发生变化,以此为基础,通过自主元素在不同组间的迁移,形成了组拓扑结构的不断演化.

在此,为了突出演化的效果(N/d 越小,效果越明显),我们选择了一个相对较小的系统规模(N=100,d=9)进行实验.

***** 这是由于当自主元素在组间迁移时会涉及到信誉值的更新,而组信誉模型所带来的消息减少由于组规模很小,并不足以抵消这部分的额外消息代价.

© 中国科学院软件研究所 http://www.jos.org.cn

在图5中,我们采用了与第6.2节相同的自主元素类型设置,考察在2 000个迭代周期内组的演化轨迹(其中,nodeID:1~10 为*A*类;11~20 为*B*类;其余为*C*类).其中,每条水平直线代表一个独立的组,线条的颜色深浅代表组的信誉高低(信誉越高的颜色越深).可以看到,通常情况下,信誉越高的组所具有的生命周期越长.在迭代结束时,除了少量信誉较低的非单个成员的组存在以外,大多数留存下来的非单个成员的组都具有相对稳定和较高的信誉值.从SIC[25]的角度来解释,这意味着组内部采取合作行为(例如提供较高质量的服务)的自主元素(节点)越多,则组织越稳定.



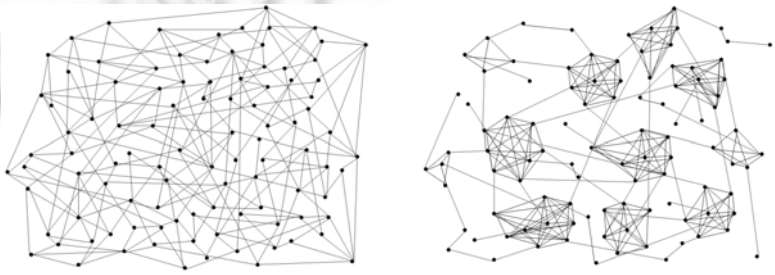
Tag ID	Node ID
94	11,12,25,28
90	36,44,53,58,69,72,73,80
86	19
83	29,94
75	31,38,47,48,68,70,74,98
69	10,66
68	37,43,45,46,64,83,85
58	32,49,52,55,57,67,76,92
44	35,60,65,78,79,88,93
36	30,41,61,84
35	14,26,89,90,95
30	15,27
24	0,24
23	23
22	33,40,50,54,77,81,86,96,
20	20,34,63,82,87,97,99
17	39,42,51,56,59,62,71,75,91
16	16
15	1,6
13	13
11	9
10	17
8	21
6	5
5	18
4	4,22
3	3
2	2,8
0	7

Fig.5 Evolution trace of cliques, the composing of cliques after evolution is listed on right side

图 5 组的演化轨迹,右侧为演化后的组成员构成

通过对固定系统规模(N=100,d=9)的网络进行重复仿真(针对不同的随机设置重复 10 遍,每遍 2 000 个迭代周期),我们发现迭代结束时,系统内的组数量基本稳定在 25~32 个范围内,而此时成员数较多(≥7)的组的平均信誉为 0.998 7(省略其余小数部分).

图 6 为与图 5 对应的迭代前后,组拓扑结构的示意图.



(a) Before evolution (random topology) (b) After evolution (clustered topology)

(a) 演化前(随机拓扑) (b) 演化后(聚合拓扑)

Fig.6 Clique topology evolution (corresponding with Fig.5)

图 6 组拓扑演化(与图 5 对应)

© 中国科学院软件研究所 http://www.jos.org.cn

6.4 基于组拓扑演化的激励

拓扑演化的目的是通过自主元素不同的拓扑位置的改变,为向其提供差异性的服务反馈奠定基础,从而激励自主元素的良好行为(积极贡献和提供可靠的服务),孤立和抑制不良行为.为了对拓扑演化的结果进行评价,我们引入组拓扑地位的概念:

定义 8(自主元素的组拓扑地位). 对于拓扑中的任意自主元素i,称$W_i=\sum_{j\in H(i)} T_j$ ($H(i)$为i在组拓扑中的邻居自主元素,T_j为自主元素j的组信誉值)为i在组拓扑中的地位.

定义 9(自主元素的平均组拓扑地位). 对于拓扑中的i类(集合记为S_i)自主元素,称$Q_i=\frac{\sum_{m\in S_i} W_m}{|S_i|}$为$i$类自主元素在组拓扑中的平均地位.

在此,我们依据自主元素提供服务的成功率将其划分为 10 类($S1$~$S10$),对应的服务成功率$P_i=i/10$, $i=0,\ldots,10$,如图 7 所示.

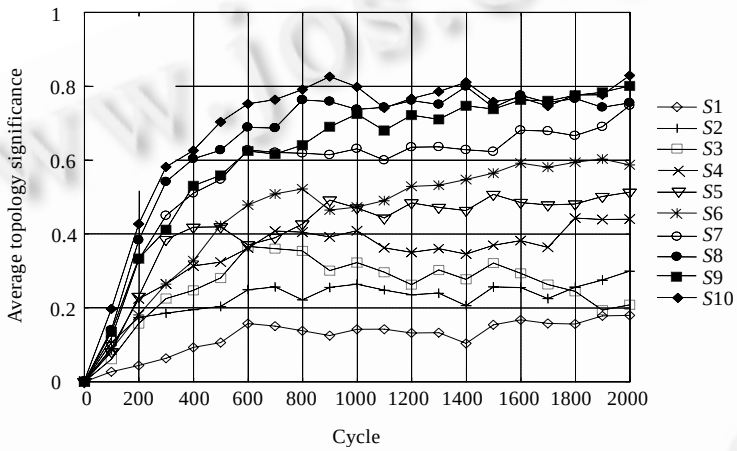


Fig.7 Topology significance for 10 types of AE

图 7 10 类自主元素的拓扑地位

显然,对于以上不同类型的自主元素,随着系统的演化,自主元素的服务成功率越高,其平均的组拓扑地位也越高.造成这一结果的根本原因是,在大量交互后,表现较好的自主元素会通过组织演化形成一定的聚集,确保其周围的邻居都是具有较好服务质量的自主元素;而行为不好的自主元素则由于无法获得稳定的组织地位,不但其信誉难以提升,同时也很难与高信誉的自主元素建立稳定的邻居关系.

以上结果意味着,随着系统的演化,服务质量好、对系统贡献较大的自主元素也将能够获得更好的服务质量;反之,服务质量较差的自主元素则必须通过不断提供可靠服务,提升自身的信誉以改善所处的网络地位,这也就达到了基于信誉的激励所期望的“赏善罚恶”的目的(参见第 5 节).

6.5 针对CRS的恶意攻击行为

由于在 CRS 中自主元素信誉的积累事实上是与其他同组自主元素共同实现的,这使得恶意自主元素有可能通过伪装,利用与较高信誉自主元素之间的信任关系获得好的信誉评价,并进而采取不良的行为获利.

在此,我们主要考虑一类针对组信誉机制的恶意攻击行为,即自主元素通过伪装成好的自主元素(服务质量高),加入由好的自主元素构成的组,在获取较高信誉以后,改变行为进行非法获利的情况.为了有所比较,我们仍然考虑采用组信誉和个体信誉模型情况下自主元素信誉的变化,以此来反映组信誉模型对于此类恶意行为的抵御能力.

我们考虑一种最为极端的攻击情况,即不良自主元素首先通过采取合作行为,提供可靠服务,进入了一个最

© 中国科学院软件研究所 http://www.jos.org.cn

高信誉等级的组(T>0.95),然后改变行为为不合作,开始提供不可靠的服务.其不合作行为可以进一步分为两种情况:

(a) 完全的不合作行为,即其不论对同组的自主元素还是不同组的自主元素都采取不合作的行为,提供不可靠的服务.

(b) 策略性的不合作行为,即其对同组的自主元素采取合作的行为总是提供可靠的服务,但是对非同组的自主元素则采取不合作的行为,提供不可靠的服务.显然,策略性的不合作行为将具有更大的危害性.

6.5.1 完全的不合作行为

采用第6.2.1节相同的环境设置.在前2 000个周期,伪装自主元素一直采取好的行为,其信誉等级达到最高.接下来,其改变行为模式为完全不合作.

由图8可以看出:(1) 2 000个迭代周期后,随着系统内部交易的增加,由于不合作自主元素的行为改变,导致组信誉整体出现波动,进而随着其内部交易失败次数的增加,不合作自主元素最终被组所开除.以此为分界,组的信誉很快回复到原有水平,而不合作自主元素的信誉则不断下滑,并最终降低到其应有的信誉等级附近.(2) 相对于个体信誉模型平缓的变化曲线,组信誉模型对于自主元素的这种行为改变具有更加“敏捷”的反应,因此也能够更加准确地反映自主元素个体即时的行为特征.究其原因,我们认为,由于组信誉模型在信誉构造的过程中,通过组邻居间历史交互信息的共享获得了更多的“直接信任”信息,因此,对于自主元素的行为改变能够作出更快的响应.

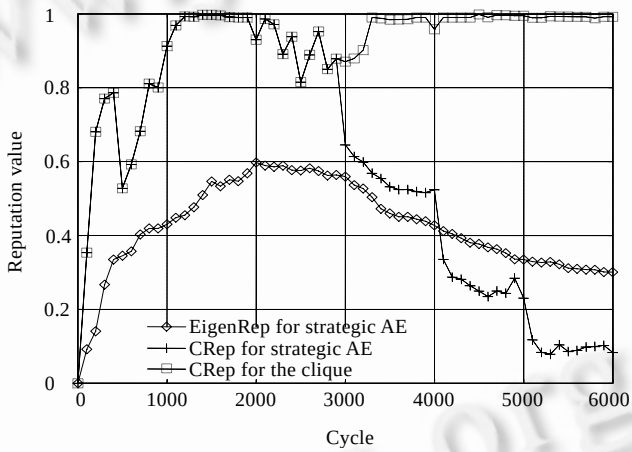


Fig.8 Reputation curves with a complete uncoorperate AE

图8 包含完全不合作自主元素时的信誉曲线

6.5.2 策略性的不合作行为

作为一种更加“合理”的攻击模式,策略性的不合作行为由于保持了与组邻居的合作关系,可能导致对于组邻居元素长期的信誉伤害,因此,显然具有更大的危害性.

图9为就此问题进行的仿真.从图中可以看出,虽然策略性不合作自主元素并不会被组所开除(因为其内部评价都是好的),但是,随着其不合作行为的增多,组信誉呈现出下降的趋势,因此导致其他合作自主元素为了寻求更好的服务地位而离开,最终结果仍然是将不合作自主元素孤立出来,并使其信誉下降到应有的水平.就图中来看,在不合作自主元素的7个邻居元素中,只有两个因为离开组较晚而信誉出现了大幅度的下滑(信誉下降服务ΔT>0.5).通过分析具体的演化过程,我们发现其原因是这两个自主元素在不良自主元素改变策略的后续演化过程中仅为组外部自主元素进行了较少的几次成功服务,因此没有得到足够的申请机会.这事实上也反映出系统对于积极参与合作的行为具有较好的及时响应能力,相反,行为不积极的自主元素的服务地位难以得到保障.

综上,通过以上仿真与分析,我们认为组信誉机制不但在模型的刻画能力和算法收敛性上具有现有信誉模型所不具备的特点,而且,通过组演化所形成的自主元素聚集有利于根据自主元素的贡献度和行为特征提供差

© 中国科学院软件研究所 http://www.jos.org.cn

异性的服务,达到激励其参与和积极贡献的目的.此外,针对策略性攻击行为的仿真也说明,组信誉机制是具有一定鲁棒性的信誉机制.

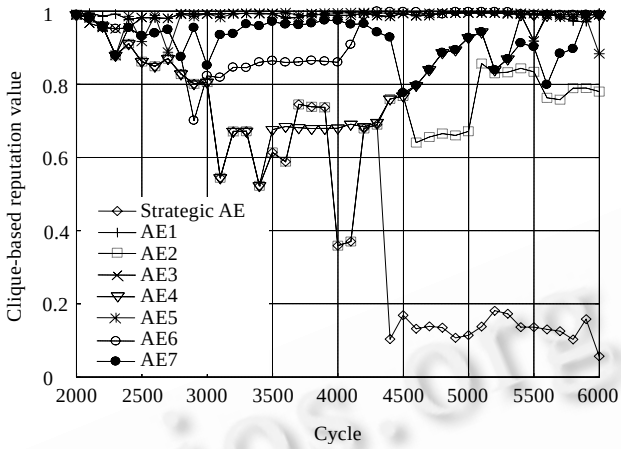


Fig.9 Reputation curves with a strategic uncoorperate AE

图 9 包含策略性不合作自主元素时的信誉曲线

7 结 论

对于自组织虚拟计算环境而言,由于无法直接控制自主元素的参与行为,使得系统常常面临严重的可用性问题,这使得提出有效的机制、激励节点积极贡献和可靠服务、抑制不良行为的影响成为其发展的必须.本文所提出的组信誉机制(CRS),其核心是通过信誉驱动的组拓扑演化赋予自主元素不同的拓扑地位,一方面使得行为良好的节点能够通过聚集相互更加有效地提供良好的服务,有利于系统整体运行的稳定;另一方面,行为不良的自主元素总是会被孤立于拓扑的边缘,其所能获得的服务范围和服务质量都是相对较差的,从而促使其为了获得更好的服务而改善行为,进而形成对其积极参与贡献和提供可靠服务的有效激励.

CRS 的一个重要特点是,通过基于组的信誉评价和信誉驱动的组演化的相互作用,在获得良好激励效果的同时,也改进了信誉模型在节点特征能力和算法收敛性等方面的不足.通过 CRS 的研究我们认为,在信誉机制的研究中,信誉模型与信誉应用机制的关联设计不仅能使信誉机制在促进自组织环境健康运行和良性发展方面起到有效的作用,而且对于获得更为准确、高效的信誉评价也是非常有价值的.

References:

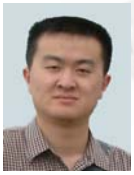
[1] Foster I, Kesselman C. The Grid 2: Blueprint for a New Computing Infrastructure. San Francisco: Morgan Kaufmann Publishers, 2004.
[2] Oram A. Peer-to-Peer: Harnessing the Power of Disruptive Technologies. Sebastopol: O'Reilly and Associates, 2001.
[3] Milojicic DS, Kalogeraki V, Lukose R, Nagaraja K, Pruyne J, Richard B, Rollins S, Xu ZC. Peer-to-Peer computing. Technology Report, HPL-2002-57, Palo Alto: HP Labs, 2002.
[4] Self-Organizing systems (SOS) FAQ version 2.99 for USENET newsgroup comp.theory.self-org-sys. 2006. http://www.calresco.org/sos/sosfaq.htm
[5] SETI@home. Current total statistics. 2006. http://seticlassic.ssl.berkeley.edu/totals.html
[6] eMule. 2006. http://www.emule-project.net
[7] KaZaA file sharing network. 2002. http://www.kazaa.com
[8] Hughes D, Coulson G, Walkerdine J. Free riding on gnutella revisited: The bell tolls? IEEE Distributed System Online, 2005,6(6). http://csdl.computer.org/comp/mags/ds/2005/06/o6001.pdf
[9] Bernardo EA, Huberman A. Free riding on Gnutella. Technical Report, SSL-00-63, Xerox PARC, 2000.
[10] Kamvar SD, Schlosser MT. EigenRep: Reputation management in P2P networks. In: Proc. of the 12th Int'l World Wide Web Conf. New York: ACM Press, 2003. 123−134.
[11] Chang E, Dillon T, Hussain FK. Trust and Reputation for Service-Oriented Environments: Technologies for Building Business Intelligence and Consumer Confidence. West Sussex: John Wiley & Sons, 2005.
[12] Xiong L, Liu L. PeerTrust: Supporting reputation-based trust in peer-to-peer communities. IEEE Trans. on Data and Knowledge Engineering, Special Issue on Peer-to-Peer Based Data Management, 2004,16(7):843−857.

© 中国科学院软件研究所 http://www.jos.org.cn

[13] Xiong L, Liu L. A reputation-based trust model for peer-to-peer ecommerce communities. In: Proc. of the IEEE Conf. on ECommerce (CEC 2003). San Diego: ACM Press, 2003. 228−229.
[14] Dou W, Wang HM, Jia Y, Zou P. A recommendation-based peer-to-peer trust model. Journal of Software, 2004,15(4):571−583 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/15/571.htm
[15] Wang HM, Tang YB, Yin G, Li L. Trustworthiness of Internet-based software. Science in China (Series E), 2006,36(10): 1156−1169 (in Chinese with English abstract).
[16] Lu XC, Wang HM, Wang J. Conceptions and architecture of Internet-based virtual computing environment. Science in China (Series E), 2006,36(10):1081−1099 (in Chinese with English abstract).
[17] Kamvar SD, Schlosser MT, Garcia-Molina H. The EigenTrust algorithm for reputation management in P2P networks. In: Proc. of the 12th Int'l World Wide Web Conf. New York: ACM Press, 2003. 640−651.
[18] Nisan N, Ronen A. Algorithmic mechanism design. Games and Economic Behavior, 2001,35(1):166−196.
[19] Sami R. Distributed algorithmic mechanism design [Ph.D. Thesis]. New Haven: Yale University, 2003.
[20] Obreiter P, Nimis J. A taxonomy of incentive patterns—The design space of incentives for cooperation. In: Moro G, Sartori C, Singh MP, eds. Proc. of the 2nd Int'l Workshop on Agents and P2P Computing. LNCS 2872, Melbourne: Springer-Verlag, 2003. 678−685.
[21] Golle P, Leyton-Brown K, Mironov I. Incentives for sharing in peer-to-peer networks. In: Proc. of the 3rd ACM Conf. on Electronic Commerce. LNCS 2232, Tampa: Springer-Verlag, 2001. 75−82.
[22] Garcia-Martinez A, Feldman M. Gnushare: Enforcing sharing in Gnutella-style peer-to-peer networks. University of California at Berkeley, 2002. http://www.sims.berkeley.edu/~mfeldman/research/gnushare.pdf
[23] Axelrod R. The Evolution of Cooperation. New York: Basic Books, 1984.
[24] Cohen B. Incentives build robustness in BitTorrent. In: Proc. of the 1st Workshop on the Economics of Peer-to-Peer Systems (P2P-Econ 2003). Berkeley: ACM Press, 2003.
[25] Hales D. Sociologically inspired approaches for self-*: Examples and prospects. In: Babaoglu O, Jelasity M, Montresor A, Fetzer C, Leonardi S, van Moorsel A, van Steen M, eds. Proc. of the Workshop on Self-Star Properties in Complex Information Systems. LNCS 3460, Berlin: Springer-Verlag, 2005. 433−445.
[26] Zhang WY. Game Theory and Information Economics. Shanghai: Shanghai Shenghuo-Dushu-Xinzhi Joint Publishing Company, Shanghai People's Publishing House, 2004 (in Chinese).
[27] Singh A, Liu L. TrustMe: Anonymous management of trust relationships in decentralized P2P systems. In: Proc. of the 3rd IEEE Int'l Conf. on P2P Computing. IEEE Computer Society, 2003. 142−149.
[28] Dou W. The research on trust-aware P2P topologies and constructing technologies [Ph.D. Thesis]. Changsha: National University of Defense Technology, 2003 (in Chinese with English abstract).
[29] Ratnasamy S, Handley M, Karp R, Shenker S. Application-Level multicast using content-addressable networks. In: Crowcroft J, Hofmann M, eds. Proc. of the 3rd Int'l Workshop on Networked Group Communication. LNCS 2233, Berlin: Springer-Verlag, 2001. 14−21.
[30] Karger D, Lehman E, Leighton T, Levine M, Lewin D, Painigraphy R. Consistent hashing and random trees: Distributed caching protocols for relieving hot spots on the world wide Web. In: Proc. of the ACM Symp. on Theory of Computing. ACM Press, 1997. 654−663.

附中文参考文献:

[14] 窦文,王怀民,贾焰,邹鹏.构造基于推荐的 Peer-to-Peer 环境下的 Trust 模型.软件学报,2004,15(4):571−583. http://www.jos.org.cn/1000-9825/15/571.htm
[15] 王怀民,唐扬斌,尹刚,李磊.互联网软件的可信机理.中国科学(E 辑),2006,36(10):1156−1169.
[16] 卢锡城,王怀民,王戟.虚拟计算环境 iVCE:概念与体系结构.中国科学(E 辑), 2006,36(10):1081−1099.
[26] 张维迎.博奕论与信息经济学.上海:上海三联书店,上海人民出版社,2004.
[28] 窦文.信任敏感的 P2P 拓扑构造及其相关技术研究[博士学位论文].长沙:国防科学技术大学,2003.



唐扬斌(1975－),男,四川开江人,博士生,讲师,主要研究领域为可信计算,分布式算法设计.

王怀民(1962－),男,博士,教授,博士生导师,CCF 高级会员,主要研究领域为分布计算中间件,软件 Agent,网络与信息安全.

常俊胜(1978－),男,博士生,主要研究领域为分布式信任管理.

© 中国科学院软件研究所 http://www.jos.org.cn