

一种多 agent 系统的组织适应模型研究

曹凤雪¹, 黄 成²

(1 江苏技术师范学院 计算机工程学院, 江苏 常州 213001; 2 江苏技术师范学院 电气信息工程学院, 江苏 常州 213001)

摘 要: 为了使组织适应动态、自治的多 agent 系统环境, 需要改变组织结构. 通过在原有的多 agent 系统上面额外增加一层的方法, 建立一个提供协助服务的 MAS 的两层抽象模型. 新增加的一层用来负责组织适应; 为了协助组织, 给出了组织适应的相关函数. 通过在对等网络中的模拟, 验证了在原有系统上面增加一层建立的模型具有更好的适应性能.

关键词: 适应; 组织; 协调; 多 agent 系统

中图分类号: TP18

文献标识码: A

文章编号: 1000-7180(2012)09-0055-04

An Organizational Adaptation Model Study of MAS

CAO Feng-xue¹, HUANG Cheng²

(1 School of Computer Engineering, Jiangsu Teachers University of Technology, Changzhou 213001, China; 2 School of Electricity and Information Engineering, Jiangsu Teachers University of Technology, Changzhou 213001, China)

Abstract: The organization structure must change to make the organization adapt to the dynamic, autonomy multi-agent system (MAS) environment. Through adding an additional layer in the original MAS, we build a two-level assisted MAS model. The new added layer is in charge of organization adaptation. In order to help the organization, the paper gives the correlation function of organization adaptation. Through the simulation in a peer-to-peer network, it is proved that the proposed model in which an additional layer is added in the original system has better adaptation performance.

Key words: adaptation; organization; coordination; MAS

1 引言

多 agent 系统是由在一定环境下的一组相互交互的 agent 构成, agent 的自治性以及环境的动态性需要整个多 agent 系统组织的改变来适应新的环境, 因此多 agent 系统的组织适应就成为一个重要的研究课题. 为了解决组织的动态适应性问题^[1-3], 基本上是以 agent 为中心的多 agent 系统 (AC-MAS)^[4] 的角度来研究的, 组织变化来自于 agent 的行为^[5-7]. 整个系统的控制通过引入监督 agent 负责在 agent 中重新分配任务来达到适应. 然而, 任务分配会限制 agent 的自治性. 因此, 在处理自私的自治 agent 时, 这是一个不合适的方法. 本文以组织为中

心的多 agent 系统 (OCMAS)^[8] 的角度来研究组织适应, 通过设置规范, 来调节 agent 的行为并且保留大部分自治和个人利益. 在变化的环境中, 这些规范或 agent 的关系能够更好的完成系统目标.

2 提供援助服务的 MAS 两层模型

文中提出一个两层协助 MAS 体系模型 (2-LAMM), 它能够在变化的环境下适应多 agent 系统有秩序的社会协定. 把原来的多 agent 系统组织称为领域层 (DL), 在 DL 上额外增加一层适应层 (AL), 它具有自己的社会协定. 另外, 在两层之间, 定义一个通信接口 (Int). 模型表示成: $M = \{DL, AL, Int\}$.

收稿日期: 2012-02-25; 修回日期: 2012-03-27

基金项目: 江苏技术师范学院项目 (KYY10043)

每一层都有一组 agent(Ag_{xL})和由社会结构或组织(Org_{xL})定义的社会协定以及一组规范($Norm_{xL}$)组成. 每一层都可以定义为: $xL = \{Ag_{xL}, Org_{xL}, Norm_{xL}\}$. 另外, 社会结构中有一些角色($Role_{xL}$)和扮演一定角色的 agent 之间关系(Rel_{xL}). 组织结构表示成: $Org_{xL} = \{Role_{xL}, Rel_{xL}\}$. 虽然在每层的角色不同, 但是一个 agent 在不同的层中可以担当不同的角色. 规范可以用一阶逻辑公式定义许可、禁止和义务.

给部分系统提供援助来完成领域行为, 如图 1, 两层通过接口进行通信. 此外, 假定, 一个多 agent 系统由组织内一组 agent 参与的环境组成(见等式 1, 通过等式 2, 3, 4, 表示出这个结构层). 在 AL 内的 agent 提供了协助服务, 以后, 把它们看作是助手. 值得注意的是, 可能给后来的更新之前的层组织的 AL 奠定了基础. 关于系统设计意图($Goals \in Org$), 在两层之间的共享是可扩展的. 由于它们被指定在领域层($Goals_{DL} \in Org_{DL}$), 及 AL 的目标($Goals_{AL} \in Org_{AL}$)下来帮助领域层去完成它们的目标.

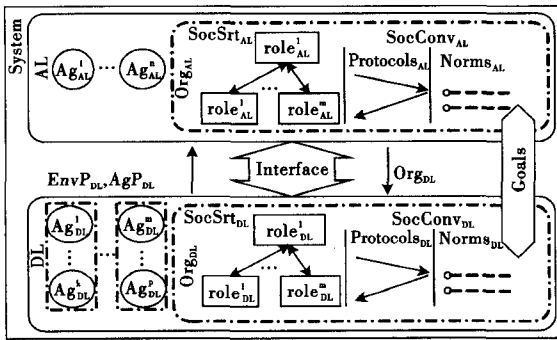


图 1 提供协助服务的 MAS 的两层抽象模型(2-LAMM)

$$MAS = Ag \times Org \times Env \quad (1)$$

$$2LAMM = AL \times DL \quad (2)$$

$$AL = Ag_{AL} \times Org_{AL} \times Env_{AL} \quad (3)$$

$$DL = Ag_{DL} \times Org_{DL} \times Env_{DL} \quad (4)$$

由于 ML 需要感知 DL 的行为, 那么整个 DL 可以认为是 ML 环境的一部分, 通过接口来传输的仅仅是能够在 agent 或环境中的可见的属性($AgP_{DL}^S/EnvP_{DL}^S$)以及组织规范(Org_{DL}).

$$Env_{AL} = EnvP_{AL} \times AgP_{DL}^S \times Org_{DL} \times EnvP_{DL}^S \quad (5)$$

然而, 任何一个多 agent 系统都能应用于这个模型. 如此, 助手恰好是“负责”援助领域层 agent 的一个子集(后面称之为族群)($cluster \subset Ag_{DL}$). 这会

导致部分信息假象. 其中, 助手仅仅感知到受助 agent 和处于本地环境下的可见属性. 而族群可以依靠具体领域的标准来定义. 以交通为例, AL 相当于一个交通管理部门, 它规定(agent 所在的)整个交通网的交通规则, 即领域层. 在这个框架中, 族群定义为—组车辆 agent 正在通过一个给定的公路网区域. 因而, 一个助手能够观察到它所协助的汽车的的速度属性(AgP_{DL})和在它的区域中($EnvP_{DL}$)的交通流密度.

如等式 6, agent 可以描述成一组属性. 所有的 agent 具有相同的属性($AgProp_i \in AgP$), 但是属性值不同(也就是在交通情景中, 汽车以不同的速度行驶 $AgP = \{Speed\}$). 因而, 一个助手能够观察在它的族群中所有 agent 的属性(见等式 7). 同样, 等式 8 计算这个可扩展的系统的可见属性作为在族群中的可见属性的联合. 关于环境属性, 使用一个类似符号来表示环境区域的属性, 一群 agent 位于此区域内($EProp_i \in EnvP^C$), 以及它们如何聚集在系统层中($EnvP^S$) (见等式 9, 10). 此外, 在交通例中, 一个助手能够测量它所负责的公路网区域的平均交通密度. 然后, 如果助手们都共享他们的数据, 就能够计算出总的平均值.

$$AgP = AgProp_1 \times \dots \times AgProp_{\#ag_prop} \quad (6)$$

$$AgP^C = \bigcup_{j=1}^{m_i} AgP_j \quad (7)$$

$$AgP^S = \bigcup_{i=1}^n AgP_i^C \quad (8)$$

$$EnvP^C = EProp_1 \times \dots \times EProp_{\#env_prop} \quad (9)$$

$$EnvP^S = \bigcup_{i=1}^n EnvP_i^C \quad (10)$$

3 组织协助

3.1 适应函数

把组织适应定义为一个函数 α^O , 依靠系统的可见性属性($EnvP^S$ 和 AgP^S)和当前的组织, 这个函数提供了一个更新过的组织(Org). 公式 11 定义了这个函数的领域和范围.

$$\alpha^O: EnvP^S \times AgP^S \times Org \rightarrow Org \quad (11)$$

根据组织设计, 组织构件的适应性将是完全非独立的、部分关联的或完全独立的. 当做出适应决策时, 越是非独立, 越是需要信息. 在适应性上的驱动力就是完成目标, 所以, 目标 Goals 认为是所有组织构件的适应函数的目标. 因而, 作了一个部分关联的假设. 如果所有构件是非独立的, 整个组织当适应时将是必须的. 很明显, 这样一个情况也意味着增加适

应函数的复杂度.

$$\alpha^{\text{ss}}; \text{EnvP}^S \times \text{AgP}^S \times \text{Goals} \times \text{SocStr} \rightarrow \text{SocStr} \quad (12)$$

$$\alpha^{\text{sc}}; \text{EnvP}^S \times \text{AgP}^S \times \text{Goal} \times \text{SocConv} \rightarrow \text{SocConv} \quad (13)$$

$$\alpha^{\text{p}}; \text{EnvP}^S \times \text{AgP}^S \times \text{Goals} \times \text{Prot} \rightarrow \text{Prot} \quad (14)$$

$$\alpha^{\text{N}}; \text{EnvP}^S \times \text{AgP}^S \times \text{Goals} \times \text{Norms} \rightarrow \text{Norms} \quad (15)$$

$$\alpha^{\text{G}}; \text{EnvP}^S \times \text{AgP}^S \times \text{Goals} \rightarrow \text{Goal} \quad (16)$$

因此,把社会体系结构(α^{ss} ,见公式 12)的适应性定义成一个函数,即依靠系统状况如已定义的目标和当前的社会结构,这个函数提供一个更新过的社会结构,即它的可见性属性.类似地,定义社会协定的适应性(α^{sc} ,见公式 13),以及一个适应每个构件的函数.具体来讲,交互协议适应函数(α^{p})和另一个适应规范(α^{N})在公式 14~15 中被定义.利用同样的方式,形式化定义了目标适应函数(公式 16).总之,当有一个特定的组织 $\text{org} \in \text{Org}$,它的适应性形式化定义如下:

定义 1 给定一个组织 $\text{org} = (\text{socstr}, \text{socconv}, \text{goals}) \in \text{Org}$. 其中, $\text{socconv} = (\text{prot}, \text{norms})$, 环境属性的当前值是 envp^s 以及 agent 属性当前值为 agp^s , 定义 org 的组织适应性为:

$$\alpha^{\text{O}}(\text{envp}^s, \text{agp}^s, \text{org}) = \text{org}'$$

其中:

$$\text{org}' = (\text{socstr}', (\text{prot}', \text{norm}'), \text{goals}')$$

$$\text{socstr}' = \alpha^{\text{ss}}(\text{envp}^s, \text{agp}^s, \text{goals}, \text{socstr})$$

$$\text{prot}' = \alpha^{\text{p}}(\text{envp}^s, \text{agp}^s, \text{goals}, \text{prot})$$

$$\text{norms}' = \alpha^{\text{N}}(\text{envp}^s, \text{agp}^s, \text{goals}, \text{norm})$$

$$\text{goals}' = \alpha^{\text{G}}(\text{envp}^s, \text{agp}^s, \text{goals})$$

这些适应函数为了修改特定的组织构件,对当前系统状况做出了评价.如上所述,这些改变由系统目标驱动,因此改变将诱导完成当前更高的目标.当在领域层的 agent 未能得到预期的性能时,适应层的助手就申请这些适应函数.从这个意义上来讲,可以解释成一个自上而下的适应方法,然而,在 agent 入口处可能需要改变时,例如目标适应,适应层也能灵敏的改变.此案例接近于一种自底向上的方法,一种混合的适应方法可能更加灵活.

3.2 组织适应举例

在交通场景中,对适应函数提供一些简短说明.此场景有一个社会结构适应函数(α^{ss}),能够增加角色(举例, $\text{rol}' = \text{rol} \cup \{\text{Ambulance}\}$)以及它们的关系(举例, $\text{rel}' = \text{rel} \cup \{\text{PickInjured}(\text{ambulance}, \text{car})\}$)

来处理冲突,并且有助于恢复交通流量.有一个协议适应函数(α^{p}),能够更新当前的转向程序包括安全距离(举例 $\text{prot}' = \{\text{TurningSafely}\}$)防止冲突.另外,当有大量的汽车时,标准适应函数(α^{N})能够降低速度限制,避免冲突的发生.最后,相对交通流量来讲,目标适应函数(α^{G})可以更新目标,相对交通流更加重视冲突.

4 2-LAMM 在对等网络中的应用

4.1 对等网络中的 2-LAMM

在对等网络中,定义两层:站点(peer)间相互分享数据,即 DL;负责更新社会协定来提高系统的性能,即 AL.

在 DL 层,参与的 agent(Ag_{DL})担当角色 peer,因此, $\text{Role}_{\text{DL}} = \{\text{peer}\}$ 通过互联,每个 agent 可以和其他 agent 相互联系.而在整个网络中,站点通常是只联系一部分相近的站点.把这种情况下定义一个子网络,子网络是整个 agent 组织的一部分,而且子网络中的站点之间的关连由考虑了系统状态后的 AL 层来更新.为了简单化,假定延迟不随时间变化.增加一个规范 Norm_{DL} ,来限制站点可以使用的带宽,并且限制了一个站点在每个时间内发送信息单位的数量.这样,假定 agent 遵循当前社会协定的话,站点就不能把网络看作是一个无穷的资源.

在 AL 层,有一种特别的角色,即助手, $\text{Role}_{\text{AL}} = \{\text{assistant}\}$,在 AL 层中担当这个角色的每个 agent 收集一个族群中的站点的信息(cluster 属于 Ag_{DL} ,族群之间是不相连的),并且调节它们本地的组织(Org_{DL}).助手的决策是基于它所关连的族群的信息(例如反应时间(EnvP)或拥有数据的站点(AgP))和其他族群从它们的邻居中获得的信息.另外,规范(Norm_{AL})也存在于这一层,因此,把 AL 层的社会协定定义为 Org_{AL} .

最后,在 AL 层上的通信比在 DL 层的通信要快,因为,在对等网络中,助手可以看作是网络提供商.

4.2 2-LAMM 在对等网络中的模拟

利用网络时间和网络的使用来评价这个模型,假定:在 DL 层, $\text{ping} = 1$, $\text{data} = 10$ and $\text{control} = 2$;在 AL 层,站点之间的信息和助手个数都是 2.

由于提出增加一层的方法,建立一个两层协助多 agent 组织模型,为了更好的说明它的性能,与传统的 agent 系统方法(称 alltoall)以及只有一个 agent 助手的情况(称为 Centralized)进行比较.方法

一(alltoall),开始时,所有的站点相互连接,并且它们通过任何可能的路径请求数据,但保证最短的执行时间(c_t),但是它没有 AL 层,所有的站点可能利用所有的连接,这样就不能防止要消耗最大的网络成本(c_n).方法二(Centralized),它有一个 AL 层,但是只有一个助手,这个助手有一个全局的信息,它能够计算出最短的路径,这样它就把最理想的邻居站点发送给每个站点,这样保证了最小的网络成本(c_n).但是,比第一种方法,执行时间可能要长.因为在方法一中,所有的站点同时发送请求,而方法二中,连接是一次对话,请求接收到,才会发送应答.

通过限制不同带宽,对这三种方法进行实验.对于方法二和方法三(Distributed),在助手之间,假定有相同的反应时间(L_{x2a})的情况下,也进行了不同的模拟.每个站点一旦有数据就会执行一次.表 1 中显示了执行时间(c_t)和网络成本(c_n)的平均值.

表 1 模拟结果

		All to All		Centralized		Distributed	
BW	L_{x2a}	c_t	c_n	c_t	c_n	c_t	c_n
1	1	525	25480	512	2896	648	5926
1	5	—	—	512	2984	688	6348
4	1	476	25053	488	2896	618	5971
4	5	—	—	488	2984	653	6394
∞	1	464	24976	481	2896	610	5979
∞	5	—	—	481	2984	645	6402
∞	30	—	—	481	3534	861	8939

结果证实,方法一,站点需要找到通往数据资源的最短的路径,需要最少的时间,但是需要最大的网络成本.方法二,助手拥有了所有的信息,需要最小的网络成本.这就表明,增加一 AL 层是值得的.方法三,增加 AL 层会节省时间,相对于方法一来讲,需要的时间就比方法一多 30%的时间,而网络成本却节省了 76%.在方法三中,助手已经拥有了它族群的知识,但是站点需要发现其他族群的数据资源的最短路径.即使把通信的反应时间,增加到 AL 的最大值,它依旧使用的网络要比没有 AL 少.例如, $BW=\infty, L_{x2a}=30$,方法三消耗的网络比方法一要少 64.2%.

5 结束语

本文提出了一个多 agent 系统组织适应的双层模型.通过增加的一层(AL)来负责动态系统的适

应.这种适应方法是分布式的,不需要全局的信息.通过在对等网络中的模拟,证明了这种两层适应模型的相对于传统的模型,可以节省网络成本.

参考文献:

- [1] 贾利民,刘刚,秦勇.基于智能 Agent 的动态协作任务求解[M].北京:科学出版社,2007:83-104.
- [2] 修保新,张维明,刘忠,等. C2 组织结构的适应性设计方法[J]. 系统工程与电子技术,2007,29(7):1103-1108.
- [3] 王正光,梁晓辉,赵沁平.基于层次图变换的多 Agent 组织结构动态重组机制[J]. Journal of Software,2009,20(1):41-53.
- [4] Ferber J, Gutknecht O, Michel F. From agents to organizations: an organizational view of multi-agent systems[C]//Giorgini P, Müller JP, Odell, et al. Agent-oriented software engineering IV, Melbourne: Springer,2004:214-230.
- [5] Deloach S A, Oyenon W H, Matson E T. A capabilities-based model for adaptive organizations[J]. Auton Agents Multi-Agent Systems, 2008,16(1):13-56.
- [6] 张莉,孙达,姚潇.基于复杂适应系统的组织学习过程研究[J]. 工业工程与管理,2011,16(3):75-80.
- [7] 夏晓,胡华平,董攀,等. BSS: 基于 BDI 对手模型的 MANET 自组织安全分族[J]. 微电子学与计算机,2007,24(10):157-160.
- [8] Scott A Deloach, Walamitien H Oyenon, Eric T Matson. A capabilities-based model for adaptive organizations [J]. Autonomous Agents and Multi-Agent Systems, 2008, 16(1):13-56.

作者简介:



曹凤雪 女,(1978—),硕士,讲师.研究方向为人工智能、多 agent 系统.



黄成 男,(1976—),硕士,讲师.研究方向为电子技术、智能控制.