

## 基于层次图变换的多 Agent 组织结构动态重组机制\*

王正光, 梁晓辉<sup>+</sup>, 赵沁平

(北京航空航天大学 计算机科学与工程学院 虚拟现实技术与系统国家重点实验室,北京 100191)

### Dynamic Reorganization Mechanisms in Multi-Agent Organizational Structure with Hierarchical Graph Transformation

WANG Zheng-Guang, LIANG Xiao-Hui<sup>+</sup>, ZHAO Qin-Ping

(State Key Laboratory of Virtual Reality Technology and System, School of Computer Science and Engineering, BeiHang University, Beijing 100191, China)

+ Corresponding author: E-mail: lxh@vrlab.buaa.edu.cn

Wang ZG, Liang XH, Zhao QP. Dynamic reorganization mechanisms in multi-agent organizational structure with hierarchical graph transformation. *Journal of Software*, 2009,20(1):41-53. <http://www.jos.org.cn/1000-9825/3334.htm>

**Abstract:** The self-adaptation to the environmental changes is one of the key issues of organization-based multi-agent systems. Dynamic reorganization of organizational structures provides an effective approach for multi-agent systems to realize organizational objectives flexibly. Based on the structural characteristics of agent organizations, this paper presents a single-rooted hierarchical graph model describing social structure, role enactment and agent coordination of the organizational structures. This model decreases effectively the complexity of reorganization for large-scale agent organizations by maintaining their structural elements based on the single-rooted and hierarchical graph approach. It formalizes the reorganization process of agent organizational structures by extending the algebraic graph transformation with the DPO (double-pushout) approach. In this formal specification, the single-rooted hierarchical graphs characterize different states of organizational structures and the derivation sequences of transformation rules formulate the transition process of organizational structures. Finally, the experimental results on reorganization simulation and matching algorithm of organization transformation rules indicate that this hierarchical graph transformation approach defines formally the reorganization process of agent organizations, and supports the graph-based design of organizational elements during the reorganization process and the reorganization computation of large-scale agent organizations.

**Key words:** dynamic reorganization mechanism; organizational structure; multi-agent system; hierarchical graph transformation; single-rooted hierarchical graph

**摘要:** 如何动态适应环境是基于组织计算的多 Agent 系统的关键研究内容之一.组织结构的动态重组为多 Agent 系统柔性地实现组织目标提供了有效途径.结合 Agent 组织结构特点,给出了一种描述组织结构的社会结

\* Supported by the National High-Tech Research and Development Plan of China under Grant No.2006AA01Z333 (国家高技术研究发展计划(863))

Received 2007-05-18; Accepted 2008-03-27

构、角色指定和 Agent 协调的单根节点层次图模型.通过单根节点和层次化地维护组织结构内元素的拓扑关系,有效地降低了大规模 Agent 组织重组问题的复杂性;扩展 DPO(double-pushout)代数图变换,形式定义了 Agent 组织结构的重组过程.单根节点层次图描述了重组过程中给定时刻的组织结构状态,图变换规则序列定义了组织结构的变化过程.Agent 组织重组和图匹配算法实验结果表明,该层次图变换方法有效地刻画了多 Agent 组织动态重组过程,并支持图形化重组过程要素设计和大规模 Agent 组织的重组计算.

关键词: 动态重组组织机制;组织结构;多 Agent 系统;层次图变换;单根节点层次图

中图法分类号: TP18 文献标识码: A

Agent 组织是基于多 Agent 计算问题的有效求解机制.在开放的环境中,Agent 组织需要更柔性地求解问题.例如,组织内 Agent 个体的知识、能力随环境的变化动态地更新;Agent 之间的关系、组织内角色关系随着任务目标的变化动态调整.为了提高 Agent 组织的自适应性以及问题求解效率,组织的演化或重组问题就成为基于 Agent 计算的关键.组织重组反映了 Agent 组织对外部环境变化和内部状态变化的一种适应策略.外部环境变化体现在组织任务目标的开放性和求解过程的开放性上<sup>[1]</sup>,内部状态变化表现在组织内 Agent 个体的行为变化和组织结构元素关系的变化上<sup>[2]</sup>.目前,组织重组问题研究分为组织变化的要素、实现组织重组的机制及过程、重组产生的时刻、重组产生的动机和可能性等方面<sup>[3]</sup>.

已有的 Agent 群体演化问题研究途径可以分为两类<sup>[4]</sup>:一类是自底向上的基于 Agent 角度方法,Agent 之间的协作行为依赖于多个 Agent 自发行为,Agent 群体的重组过程是一种突现的行为;另一类是自上而下的基于组织角度方法,Agent 之间的关系依赖于各自承担的角色之间的关系,角色关系在组织设计阶段预先设定并随环境变化重新设计,组织的重组过程具有主动性、可控性等特点.在基于组织的多 Agent 计算问题中,组织结构定义了组织内 Agent 个体的任务结构和交互方式.Agent 组织结构及其变化是重组问题的重要研究方面.

## 1 相关研究

近年来,Agent 组织重组的相关问题得到了广泛关注.Dignum 等人<sup>[2,3]</sup>提出了重组过程中组织变化的形式及重组问题的 5 个方面的研究内容,已实现了实验环境 VILLA 研究重组过程仿真,取得了很好的效果.Hübner 等人<sup>[5]</sup>提出了基于 MOISE+的组织模型的重组过程,将组织模型分为结构维度、功能维度和义务维度.功能维度描述了组织任务层次,结构维度定义了组织角色关系,义务维度定义了角色对应给定任务允许的行为及义务,将功能维度和结构维度联系起来.在组织结构中指定管理角色集中方式管理重组过程,承担该角色的 Agent 可以自主地修改组织状态,并应用于小规模机器人团队 TeamBots 仿真.从软件工程角度,DeLoach 等人<sup>[6]</sup>提出了基于 MaSE 的多 Agent 系统分析和设计方法,该方法允许小规模 Agent 系统重组计算.在 MaSE 方法中,组织内 Agent 没有分组,不支持组织的分层结构,对于大规模组织重组存在组合爆炸问题.Durfee 等人<sup>[7]</sup>强调了任务环境和组织结构影响组织的性能,分析了不同的树结构组织在特定环境下对大规模 Agent 系统的性能影响,并给出了基于组织自设计的重组过程.徐晋晖等人<sup>[8]</sup>给出了面向结构的组织结构模型、组织设计原则和非形式定义的组织形成演化过程.张伟等人<sup>[9]</sup>针对 Agent 组织的设计过程给出了 Pi 演算形式描述和基于化学抽象机的形式语义.詹剑锋等人<sup>[10]</sup>基于软件体系结构描述语言形式定义了 Agent 个体演化过程中内部组件的交互,从微观角度分析了演化过程中 Agent 个体内部状态的变化.已有工作一般都没有对 Agent 组织重组过程进行严格的形式定义和对重组过程效率深入研究.本文在分析多 Agent 计算问题及组织特征的基础上,针对这些问题从组织结构角度提出了一种描述组织结构的社会结构、角色指定和 Agent 协调 3 个维度的单根节点层次图模型,给出了 Agent 组织重组过程的基于层次图变换的形式定义,包括将层次图模型刻画重组过程的组织结构状态和扩展 double-pushout 代数图变换规则,形式定义重组过程.给出了重组过程中组织图规则应用的图匹配算法.本文的实验结果表明,该层次图变换方法有效地定义了重组过程,并避免了大规模组织重组带来的图节点组合爆炸问题.相对于 Hübner,Dignum 和 DeLoach 等人的工作,该层次图变换方法形式定义了 Agent 组织的重组过程,并支持图形化重组过程要素设计和大规模 Agent 组织的重组计算.

## 2 Agent 组织结构及重组过程语义

### 2.1 Agent 组织结构的3个维度特征

多 Agent 系统是分布式人工智能的重要研究内容<sup>[11]</sup>.在多 Agent 系统中,Agent 群体通过个体的自主计算及个体间的交互实现系统预定的全局目标.个体的自治与交互是多 Agent 系统的重要特征.在开放的问题求解环境中,由于任务目标的开放性及求解过程的开放性,Agent 系统需要面临环境自适应及矛盾冲突等问题<sup>[1]</sup>.相应地,解决自适应及矛盾冲突等问题需要系统中大量的计算资源,例如,Agent 个体之间的通信资源.近年来,基于组织的 Agent 计算有望成为多 Agent 系统求解问题的新途径<sup>[12,13]</sup>.为了实现基于组织的多 Agent 问题求解,组织计算过程可以分为 4 个阶段:(1) 对问题领域进行分析和定义,确定任务目标及任务相关的知识、资源约束;(2) 进行组织结构设计,定义组织内角色关系、角色之间交互方式和组织元素的约束规则;(3) 组织实例化,Agent 通过承担角色来完成角色指定的任务;(4) 进行问题求解,组织内 Agent 个体根据承担角色的任务和相应的能力、资源约束形成具体的行为规划.在问题求解过程中,Agent 之间的行动会存在冲突,因而阶段(4)会根据组织的重组策略返回阶段(2)或阶段(3),分别进行组织结构的重新设计和重新实例化.这两者是 Agent 组织重组研究的重要方面.

在基于组织的问题求解中,组织结构是重要的研究内容<sup>[12]</sup>.组织结构定义了 Agent 个体之间的任务结构和交互方式,任务结构定义了组织任务之间的父-子层次关系及资源约束,这是组织结构中角色关系设计的基础.角色关系反映了组织的任务结构,并定义了承担相应角色的 Agent 个体的交互.为了体现组织结构元素的这种相互关系,我们在文献[4,14]中给出了组织结构的社会结构、角色指定和 Agent 协调 3 个维度的定量描述及基于组织结构的 3 个维度将图变换方法应用于组织结构演化过程的描述方法.在文献[4,14]工作的基础上,针对具有显式层次结构的 Agent 组织从宏观角度的组织结构重组问题,本文进一步讨论了组织结构的 3 个维度模型相关约束特性、组织结构图变换规则的有效性和重组过程中图规则应用算法.组织任务之间的时间、资源约束在微观层次影响了 Agent 个体的决策行为.在显式层次结构的 Agent 组织中,组织的角色关系显现了显式的层次特征.一方面,有效地降低了承担相应角色的 Agent 个体的交互复杂性,另一方面,在相应的基于图变换的组织结构重组过程中,有效地降低了图规则匹配的计算复杂度.第 3 节将详细分析组织结构重组过程的图匹配算法.

下面首先介绍 Agent 组织结构的 3 个维度模型的相关概念.

### 2.2 组织结构的单根节点层次图模型

在具有显式层次结构的 Agent 组织中,组织的任务结构决定了组织角色及角色关系的设计,下面定义给出了任务图及任务之间约束的表示.

**定义 2.2.1(任务图).** 给定组织的有限任务集合  $TASK$ ,任务之间的层次关系形成一个有向非循环图,定义为  $G_{task} = (V_{task}, E_{task})$ .其中: $V_{task}$  是任务图的节点集,定义了任务集合, $E_{task}$  是任务图的边集,定义了任务之间的层次关系 $\leq_i$ .也即任意两个任务节点  $t_1$  和  $t_2$ ,若有  $t_2 \leq_i t_1$ ,则  $t_2$  称为  $t_1$  的子任务节点.

由于本文关注的是组织结构元素的约束关系,在本文中仅讨论组织任务结构的层次特性及任务结构导出的组织结构元素的约束,不涉及任务概念特定领域含义以及如何分解任务结构. $\leq_i$  具有以下约束特性:

**约束 2.2.1(任务层次关系约束).** 设  $G_{task} = (V_{task}, E_{task})$  为一个任务图,对任意  $t_1, t_2, t_3 \in V_{task}$ ,满足

$$\text{若存在 } (t_1, t_2) \in E_{task}, (t_2, t_3) \in E_{task}, \text{ 则有 } t_3 \leq_i t_1 \quad (\text{Transitivity})(1)$$

$$\text{若存在 } (t_1, t_2) \in E_{task}, \text{ 则有 } \neg(t_1 \leq_i t_2) \quad (\text{Dissymmetry})(2)$$

通过对组织目标的分解和合并,任务图可以生成一个特定的根节点,并满足下面的约束特性:

**约束 2.2.2(任务图单根节点约束).** 一个任务图  $G_{task} = (V_{task}, E_{task})$  存在单个根节点,记为  $Root(G_{task})$ .对任务图中任意节点  $t \in V_{task} \setminus \{Root(G_{task})\}$ ,则存在一条与根节点相连接的路径.

为了简化考虑,我们假定组织任务已经按照预期的粒度分解完毕,并假定组织结构的角色与组织的任务或子任务一一对应.本文的角色和 Agent 内部组成要素不作定义限制,为便于宏观组织结构的描述,仅假定为相应的角色关系结构和协作关系结构图节点.基于组织任务,下面给出组织结构的 3 个维度描述.

定义 2.2.2(角色关系结构或组织的社会结构). 给定两个符号集合  $\Sigma_r$  和  $\Delta_r$  分别表示节点和边的标识符集合, 组织的角色关系结构定义为一个单根节点有向非循环图, 记为  $G_{role} = (V_r, E_r, s_r, t_r, l_r, m_r)$ . 其中:  $\Sigma_r \cap \Delta_r = \emptyset$ ,  $V_r$  是节点集合, 定义了组织角色集合,  $E_r$  是边集合, 定义了角色之间基于任务的层次关系.  $s_r: E_r \mapsto V_r$  和  $t_r: E_r \mapsto V_r$  分别是给定边的源节点和目的节点映射.  $l_r: V_r \mapsto \Sigma_r$  和  $m_r: E_r \mapsto \Delta_r$  分别是节点和边的标识符映射, 角色关系结构的根节点记为  $Root(G_{role})$ . 任意角色节点  $r_1 \in V_r$ , 对应的任务记为  $r_1^t$ .

通过承担角色, 相应的 Agent 个体之间形成具有任务的委托与合作关系, 即 Agent 协作关系结构.

定义 2.2.3(Agent 协作关系结构). 给定两个符号集合  $\Sigma_a$  和  $\Delta_a$  分别表示节点和边的标识符集合, Agent 协作关系结构定义为一个有向非循环图, 记为  $G_{agent} = (V_a, E_a, s_a, t_a, l_a, m_a)$ . 其中:  $\Sigma_a \cap \Delta_a = \emptyset$ ,  $V_a$  和  $E_a$  分别是节点和边集合.  $s_a, t_a, l_a$  和  $m_a$  分别是边的源节点、边的目的节点、节点标识符和边的标识符映射.

定义 2.2.4(角色指定关系). 给定  $\Sigma_c = \Sigma_r \cup \Sigma_a$  和  $\Delta_c$  分别表示节点和边的标识符集合,  $\Sigma_r$  和  $\Sigma_a$  分别是角色和 Agent 节点标识符集合, 角色指定关系定义为一个二分图, 记为  $G_{assign} = (V_c, E_c, s_c, t_c, l_c, m_c)$ . 其中:  $\Sigma_c \cap \Delta_c = \emptyset$ ,  $V_c = V_r \cup V_a$  和  $E_c$  分别是节点和边集合,  $V_r$  和  $V_a$  分别是角色和 Agent 节点集合,  $s_c: E_c \mapsto V_a$  和  $t_c: E_c \mapsto V_r$  分别是边的源节点和目的节点映射.  $l_c: V_c \mapsto \Sigma_c$  和  $m_c: E_c \mapsto \Delta_c$  分别是节点和边的标识符映射, 对任意  $v \in V_c, l_c$  满足:

$$l_c(v) = \begin{cases} l_r(v), & v \in V_r \\ l_a(v), & \text{否则} \end{cases} \quad (3)$$

为了实例化给定的组织角色关系结构, Agent 需要承担组织内全部角色, 根据角色的能力、资源等约束, 单个 Agent 会承担多个角色. 因此, 角色指定关系满足以下约束:

约束 2.2.3(完全性约束). 给定一个角色指定关系的二分图  $G_{assign}$ , 由边集  $E_c \subseteq V_a \times V_r$  确定的映射  $f_{assign}: V_a \rightarrow V_r$  为满射, 其中, 对任意  $e \in E_c$ ,  $t_c(e) = f_{assign}(s_c(e))$ .

约束 2.2.4(无环性约束). 给定一个角色指定关系的二分图  $G_{assign}$ , 对任意节点  $a \in V_a$  承担的角色集  $S_{assign} = \{r \mid (a, r) \in E_c\}$ ,  $S_{assign}$  中任意两个角色节点  $r_1$  和  $r_2$  对应的任务不满足父子层次关系.

完全性约束保证了组织内所有角色均有 Agent 承担, 无环性约束表明承担角色的 Agent 之间不存在循环任务委托关系, 无环性约束避免承担存在资源冲突的任务对应角色的 Agent 之间循环任务委托.

基于上述 3 个方面的定义, 组织结构模型的定义如下:

定义 2.2.5(组织结构的单根节点层次图). 一个 Agent 组织结构形成一个单根节点层次图, 定义为一个三元组  $G_{org} = (G_{role}, G_{agent}, G_{assign})$ . 其中:  $G_{role} = (V_r, E_r, s_r, t_r, l_r, m_r)$  是 Agent 组织结构的角色关系结构,  $G_{agent} = (V_a, E_a, s_a, t_a, l_a, m_a)$  是 Agent 的协作关系结构,  $G_{assign} = (V_c, E_c, s_c, t_c, l_c, m_c)$  是 Agent 的角色指定关系,  $V_c = V_r \cup V_a$ . 组织结构层次图的根节点记为  $Root(G_{org})$ .

角色关系结构定义了一个 Agent 组织中角色处于整个组织内的位置层次, 因而将角色关系结构的根节点  $Root(G_{role})$  等同为组织结构层次图的根节点, 记为  $Root(G_{org})$ , 组织结构层次定义如下:

定义 2.2.6(组织结构的层次). 给定一个组织结构层次图  $G_{org}$  和组织结构的根节点  $Root(G_{org})$ , 组织结构层次关系定义为  $Lev_{Root(G_{org})}: V_r \setminus \{Root(G_{org})\} \mapsto Lay$ , 简记为  $Lev$ . 其中:  $V_r$  是  $G_{org}$  中角色关系结构的节点集,  $Lay$  是一个有限自然数集合. 对任意除根节点以外的角色节点,  $Lay$  中存在唯一的一个元素与其对应.

组织层次关系映射  $Lev$  选为给定角色节点到根节点的路径距离. 若两个角色节点存在一个直接邻接的有向边, 则该两个角色的路径距离记为 1. 当某个角色节点与根节点之间存在多个路径时, 选取最大距离作为该角色节点的层次数. 对给定角色节点  $i$ , 该节点在组织层次图中的层次数记为  $Lev(i)$ .

约束 2.2.5(角色层次约束). 给定一个组织结构图  $G_{org}$  及根节点  $Root(G_{org})$ , 任意角色节点  $i \in V_r \setminus \{Root(G_{org})\}$  的父角色节点集  $par(i) = \{j \mid j \text{ 满足 } i^t \leq j^t\}$ ,  $i^t$  和  $j^t$  分别为角色节点  $i$  和  $j$  对应的组织任务. 对任意父角色节点  $r \in par(i)$ , 满足  $Lev(i) - Lev(r) = 1$ .

角色层次约束避免了角色关系结构中给定角色节点的直接邻接的子角色节点跨多层的情况. 角色层次约束用于降低 Agent 之间的通信和活动的协调复杂性.

### 2.3 基于层次图变换的组织结构重组过程语义

Agent 组织结构的动态重组过程反映了组织内 Agent 个体随环境变化执行任务的策略调整过程.由第 2.2 节给出了组织结构的 3 个维度描述,相应地,组织结构状态演化包括组织结构的角色关系结构、角色指定关系和 Agent 之间的协作关系随环境变化的演变过程.图变换文法是 Chomsky 文法基于图重写的扩展,广泛用于程序语言操作语义、数据库、操作系统、基于规则的系统及分布式系统<sup>[15]</sup>.基于范畴论,采用图描述组织结构状态,图变换文法定义了图的对象之间的关系以及基于图变换规则的图的对象之间的关系演化.Agent 组织结构的重组过程可以采用基于规则的图变换方法进行严格的语义定义.在本文的图变换方法中,图变换系统的图的对象需要扩展为包含组织结构 3 个维度的图状态,相应的图变换规则需要扩展为 3 个维度的图的对象变换规则.为了反映组织结构的层次特性,引入的图变换方法中图类需要保持组织结构图的层次特性,并且经过图规则变换后,图的对象仍然保持组织结构的相关约束.我们将在第 2.4 节详细讨论组织结构图变换规则的这些特性.

本文所涉及到的图的态射、范畴、导出(pushouts)及图规则概念可以参考文献[15,16].给定两个图  $G, H$  和一个图规则  $r$ ,将规则  $r$  应用于图  $G$  生成图  $H$ ,记为  $G \Rightarrow^r H$ ,  $G$  称为宿主图,  $H$  称为  $G$  基于规则  $r$  的一个直接生成.为方便起见,定义 2.3.1 列出了图变换系统的组成要素.

定义 2.3.1(图变换系统)<sup>[15,16]</sup>. 一个图变换系统定义为五元组  $A = (\Gamma, \mathfrak{R}, \Rightarrow, C, \varepsilon)$ .其中:

- (1)  $\Gamma$  是一类图的对象集合;
- (2)  $\mathfrak{R}$  是一类图的规则集合,本文中讨论基于双导出的规则形式;
- (3)  $\Rightarrow$  是规则应用操作符,对所有的图规则  $r \in \mathfrak{R}$ ,规则  $r$  的语义由  $\Rightarrow$  定义的二元关系  $\Rightarrow^r \subseteq \Gamma \times \Gamma$  描述;
- (4)  $C$  是一类控制条件集合,对所有的控制条件  $c \in C, c$  的语义由二元关系  $SEM_c(C) \subseteq \Gamma \times \Gamma$  描述;
- (5)  $\varepsilon$  是一类图的表达式集合,对所有的表达式  $\chi \in \varepsilon, \chi$  的语义由集合  $SEM_\varepsilon(\chi) \subseteq \Gamma$  描述.

根据定义 2.3.1,当  $\Gamma$  分别是由组织结构的角色关系结构、Agent 协作结构和角色指定关系图形成的图类,我们分别得出组织结构的 3 个维度的图变换系统,记为  $A_{role}, A_{agent}$  和  $A_{assign}$ .从而组织的层次图变换系统可以由 3 个维度的图变换系统组合得到,定义如下:

定义 2.3.2(组织结构的层次图变换系统). 组织结构的层次图变换系统定义为一个五元组  $A_{org} = (\Gamma_{org}, \mathfrak{R}_{org}, \Rightarrow_{org}, C_{org}, \varepsilon_{org})$ .其中:

- (1)  $\Gamma_{org} = (\Gamma_{role}, \Gamma_{agent}, \Gamma_{assign})$  是组织结构的单根节点层次图的对象集合,  $\Gamma_{role}, \Gamma_{agent}$  和  $\Gamma_{assign}$  分别是组织结构的角色关系、Agent 协作结构和角色指定关系图的对象集合;
- (2)  $\mathfrak{R}_{org} = \mathfrak{R}_{role} \times \mathfrak{R}_{agent} \times \mathfrak{R}_{assign}$  是一类图的规则集合,定义为角色关系结构图变换系统规则  $\mathfrak{R}_{role}$ , Agent 协作结构图变换系统规则  $\mathfrak{R}_{agent}$  和角色指定关系图变换系统规则  $\mathfrak{R}_{assign}$  的积.对所有的规则  $r = (\alpha, \beta, \gamma) \in \mathfrak{R}_{org}$ ,规则  $r$  的语义由二元关系  $\Rightarrow^r \subseteq \Gamma_{org} \times \Gamma_{org}$  描述,也即对  $G_{org} = (G_{role}, G_{agent}, G_{assign})$ ,  $G'_{org} = (G'_{role}, G'_{agent}, G'_{assign}) \in \Gamma_{org}$ ,若  $G_{org} \Rightarrow^r G'_{org}$ ,当且仅当  $G_{role} \Rightarrow^\alpha G'_{role}$ ,  $G_{agent} \Rightarrow^\beta G'_{agent}$  和  $G_{assign} \Rightarrow^\gamma G'_{assign}$ ;
- (3)  $C_{org} = C_{role} \times C_{agent} \times C_{assign}$  是一类控制条件集合,对所有的控制条件  $c_{org} = (c_{role}, c_{agent}, c_{assign}) \in C_{org}$ ,  $c_{org}$  的语义由下式描述:

$$SEM_{org}(C_{org}) = \{((G_{role}, G_{agent}, G_{assign}), (G'_{role}, G'_{agent}, G'_{assign})) \in \Gamma_{org} \times \Gamma_{org} \mid (G_{role}, G'_{role}) \in SEM_{role}(C_{role}) \wedge (G_{agent}, G'_{agent}) \in SEM_{agent}(C_{agent}) \wedge (G_{assign}, G'_{assign}) \in SEM_{assign}(C_{assign})\} \quad (4)$$

- (4)  $\varepsilon_{org} = \varepsilon_{role} \times \varepsilon_{agent} \times \varepsilon_{assign}$  是一类图的表达式集合,对所有的表达式  $\chi_{org} = (\chi_{role}, \chi_{agent}, \chi_{assign}) \in \varepsilon_{org}$ ,  $\chi_{org}$  的语义由下式描述:

$$SEM_{org}(\chi_{org}) = SEM_{role}(\chi_{role}) \times SEM_{agent}(\chi_{agent}) \times SEM_{assign}(\chi_{assign}) \cap \Gamma_{org}, \text{其中 } SEM_{org}(\chi_{org}) \subseteq \Gamma_{org}, \\ SEM_{role}(\chi_{role}) \subseteq \Gamma_{role}, SEM_{agent}(\chi_{agent}) \subseteq \Gamma_{agent}, SEM_{assign}(\chi_{assign}) \subseteq \Gamma_{assign} \quad (5)$$

组织结构的层次图变换系统形式定义了基于图描述的自治对象的组织结构状态随环境变化的迁移过程,一系列图规则的应用刻画了组织结构状态的转换历史过程,因而由组织结构的图对象及相互之间的转换构成

了组织结构的图状态转移系统,由下面定义描述.

**定义 2.3.3(组织结构的图状态转移系统).** 给定组织的层次图变换系统  $A_{org}$ , 基于该图变换系统生成的组织图状态转移系统定义为一个三元组  $GTS_{org} = (S, L, \rightarrow)$ . 其中:  $S \subseteq \Gamma_{org}$  是组织结构状态集, 给定初始组织状态  $s_0 \in S$ , 任意状态  $s' \in S \setminus \{s_0\}$  从初始状态可达.  $L$  是图变换系统  $A_{org}$  规则的标识符集.  $\rightarrow \subseteq S \times S$  是转移集. 对  $\forall s_1, s_2 \in S$ , 若  $s_1 \rightarrow s_2$ , 则当且仅当标识符  $l \in L$  对应的  $A_{org}$  中图规则  $p \in \mathfrak{R}_{org}$  满足  $s_1 \Rightarrow^p s_2$ .

#### 2.4 重组过程的组织图变换规则特性

为了保证经过组织的图规则操作, Agent 组织结构初始状态在图变换后生成的中间状态和目标状态图仍为单根节点层次图, 我们进一步讨论层次图变换规则的特征和约束. 下面介绍本小节涉及到的几个引理.

**引理 2.4.1.** 给定初始有向非循环图  $G$ 、图规则  $r = (L, i, K, j, R)$ , 其中  $L, K$  和  $R$  是有向非循环图,  $i$  和  $j$  是两个单射. 若  $G$  基于规则  $r$  的一个直接生成图  $H$  是有向非循环图, 当且仅当对  $\forall u, v \in K$ , 若右边图  $R$  中存在从  $u$  到  $v$  的路径, 则左边图  $L$  中必然存在一条从  $u$  到  $v$  的路径.

**引理 2.4.2.** 给定初始单根有向非循环图  $G$ 、图规则  $r = (L, i, K, j, R)$ , 其中  $L, K$  和  $R$  是单根有向非循环图,  $i$  和  $j$  是两个单射. 若  $G$  基于规则  $r$  的一个直接生成图  $H$  是单根有向非循环图, 当且仅当规则  $r$  除满足引理 2.4.1 以外, 对于  $L, K$  和  $R$  的根节点  $\rho_L, \rho_K$  和  $\rho_R$  还满足  $\rho_L = i(\rho)$  且  $\rho_R = j(\rho)$ .

在给出引理 2.4.3 之前, 简要介绍几个必要的概念. 假定二分图  $B$  的节点标识符集  $\Sigma = \Sigma_s \cup \Sigma_t$ 、节点集  $V_B$  和边集  $E_B$ , 其中  $\Sigma_s$  和  $\Sigma_t$  分别为二分图源节点和目的节点的标识符集. 对任意节点  $n \in V_B$ , 定义  $n$  的入度和出度分别为  $indeg_B(n) = |\{e \mid n = t_B(e)\}|$  和  $outdeg_B(n) = |\{e \mid n = s_B(e)\}|$ . 给定  $\lambda \in \Sigma$ , 定义集合  $V_B^\lambda = \{n \in V_B \mid l_B(n) = \lambda\}$ . 给定一个非空集合  $M \subseteq \Sigma$ , 定义集合  $V_B^M = \bigcup_{\lambda \in M} V_B^\lambda$ . 对于一个二分图的图规则  $r = (L, i, K, j, R)$ , 定义 3 个图  $NEW(r) = (V_R \setminus V_K, \phi, \phi, \phi, l_R \upharpoonright_{(V_R \setminus V_K)}, \phi)$ ,  $DEL(r) = (V_L \setminus V_K, \phi, \phi, \phi, l_L \upharpoonright_{(V_L \setminus V_K)}, \phi)$  和  $PRES(r) = (V_K, \phi, \phi, \phi, l_K, \phi)$  分别为规则  $r$  中新增、删除和保留的图节点.

**引理 2.4.3.** 给定基于标识符集  $\Sigma = \Sigma_s \cup \Sigma_t$  的初始二分图  $G$ 、图规则  $r = (L, i, K, j, R)$ , 其中  $L, K$  和  $R$  是基于  $\Sigma$  上有向图,  $i$  和  $j$  是两个单射. 若  $G$  基于规则  $r$  的一个直接生成图  $H$  是基于  $\Sigma$  上的二分图, 当且仅当对任意节点  $n \in NEW(r) \cap V_R^{\Sigma_t}$  满足  $indeg_R(n) > 0$  且对任意节点  $n \in PRES(r) \cap V_R^{\Sigma_t}$ , 若  $indeg_R(n) = 0$ , 则  $indeg_L(n) = 0$ .

以上各引理的证明在文献[17]已给出, 这里略去.

**定义 2.4.1(层次图规则的保持单根节点层次性).** 给定一个组织的图规则  $r = (\alpha, \beta, \gamma) \in \mathfrak{R}_{org}$ , 称图规则  $r$  是一个保持单根节点层次性的图规则, 当且仅当任意组织结构图  $G_{org} = (G_{role}, G_{agent}, G_{assign}) \in \Gamma_{org}$ , 经过图规则  $r$  操作生成的直接导出  $G'_{org} = (G'_{role}, G'_{agent}, G'_{assign}) \in \Gamma_{org}$ .

**定义 2.4.2.** 给定两个图的偏态射  $p: S_1 \mapsto S_3$  和  $q: S_2 \mapsto S_4$ , 两个关系  $\cong^{upper} \subseteq S_1 \times S_2$  和  $\cong^{lower} \subseteq S_2 \times S_4$ , 称四元组  $(p, q, \cong^{upper}, \cong^{lower})$  是可交换的, 如果满足下列条件: 对所有的  $s_1 \in S_1$  和  $s_2 \in S_2$ , 若  $s_1 \cong^{upper} s_2$ , 则

- (1)  $s_1 \notin dom(p)$  且  $s_2 \notin dom(q)$ ; 或者
- (2)  $s_1 \in dom(p)$ ,  $s_2 \in dom(q)$  且  $p(s_1) \cong^{lower} q(s_2)$ .

其中,  $dom(p)$  和  $dom(q)$  分别是态射  $p$  和  $q$  的定义域.

在组织结构的图变换规则中, 我们将定义 2.4.2 中的两个关系分别定义为对应两个规则的左边图的公共元素和右边图的公共元素, 因此, 将定义 2.4.2 规定的可交换性约束作为角色关系结构和角色指定关系的图规则之间约束, 可以保证对应的图规则应用过程中角色关系结构与角色指定关系之间元素的一致性.

**约束 2.4.1(组织图规则的一致性条件).** 给定一个组织结构的图规则  $r = (\alpha, \beta, \gamma)$ ,  $\alpha = (L_\alpha, i_\alpha, K_\alpha, j_\alpha, R_\alpha)$ ,  $\beta = (L_\beta, i_\beta, K_\beta, j_\beta, R_\beta)$  和  $\gamma = (L_\gamma, i_\gamma, K_\gamma, j_\gamma, R_\gamma)$ . 其中  $L_x, K_x$  和  $R_x$  分别是规则  $x$  的左边图、接口图和右边图.  $i_x: K_x \mapsto L_x$ ,  $j_x: K_x \mapsto R_x$  为单射,  $x \in \{\alpha, \beta, \gamma\}$ . 称组织的图规则  $r$  是操作上一致的, 如果满足下列条件:

- (1) 存在偏态射  $\alpha_p: L_\alpha \mapsto R_\alpha$ ,  $\gamma_p: L_\gamma \mapsto R_\gamma$ , 关系  $\cong_\alpha^L \subseteq L_\alpha \times L_\gamma$ ,  $\cong_\alpha^R \subseteq R_\alpha \times R_\gamma$  有  $(\alpha_p, \gamma_p, \cong_\alpha^L, \cong_\alpha^R)$  可交换;
- (2) 且存在偏态射  $\beta_p: L_\beta \mapsto R_\beta$ , 关系  $\cong_\beta^L \subseteq L_\beta \times L_\gamma$ ,  $\cong_\beta^R \subseteq R_\beta \times R_\gamma$  有  $(\beta_p, \gamma_p, \cong_\beta^L, \cong_\beta^R)$  可交换.

由约束 2.4.1 的定义可知,  $\cong_{\alpha}^L \subseteq L_{\alpha} \times L_{\gamma}$  和  $\cong_{\alpha}^R \subseteq R_{\alpha} \times R_{\gamma}$  是集合上的二元关系,表示对应的集合之间的公共元素,  $(\alpha_p, \gamma_p, \cong_{\alpha}^L, \cong_{\alpha}^R)$  的可交换性表示该二元关系  $\cong_{\alpha}^L \subseteq L_{\alpha} \times L_{\gamma}$  经过一个或多个分别建立在规则  $\alpha, \gamma$  对应的左边图与右边图的偏态射序对操作后仍然满足二元关系  $\cong_{\alpha}^R \subseteq R_{\alpha} \times R_{\gamma}$ .

下面的命题给出组织的图规则保持单根节点层次性的等价条件.

**定理 2.4.1.** 给定 Agent 组织结构图  $G_{org} \in \Gamma_{org}$  和组织的图规则  $r = (\alpha, \beta, \gamma) \in \mathfrak{R}_{org}$ , 若  $G_{org}$  基于规则  $r$  的一个直接生成图  $G'_{org}$  满足  $G'_{org} \in \Gamma_{org}$ , 当且仅当规则  $\alpha$  满足引理 2.4.2, 规则  $\beta$  满足引理 2.4.1, 规则  $\gamma$  满足引理 2.4.3, 且规则  $r$  满足性质 2.4.1 的约束.

**证明:** 给定组织结构的初始状态图  $G_{org} \in \Gamma_{org}$  及  $G_{org} \Rightarrow^r G'_{org}$ . 在满足引理 2.4.1、引理 2.4.2 和引理 2.4.3 的情况下,  $G'_{org}$  的各组成子部分  $G'_{role}$ ,  $G'_{agent}$  和  $G'_{assign}$  依次保持为单根有向非循环图、有向非循环图和二分图. 因而, 定理 2.4.1 需要进一步证明  $G'_{role}$  与  $G'_{assign}$ ,  $G'_{agent}$  与  $G'_{assign}$  之间的元素保持一致性, 可以由定义 2.4.2、约束 2.4.1 以及 double-pushouts 规则应用得出定理 2.4.1 的结论, 由于篇幅所限, 证明过程此处略去.

### 3 Agent 组织结构重组过程算法

Agent 组织结构重组反映了组织对外部环境变化和内部结构变化的一种自适应策略. 图变换系统形式地定义了基于图描述的组织结构状态随各种因素变化的演变过程. 外部环境变化要求组织任务目标的开放性和问题求解过程的开放性. 后者要求在求解过程中 Agent 知识、能力不断更新, 从微观角度反映了 Agent 个体的自适应过程. 本文从宏观角度分析, 在组织任务的开放性因素下, 组织结构的动态重组过程, 具体如图 1 所示.

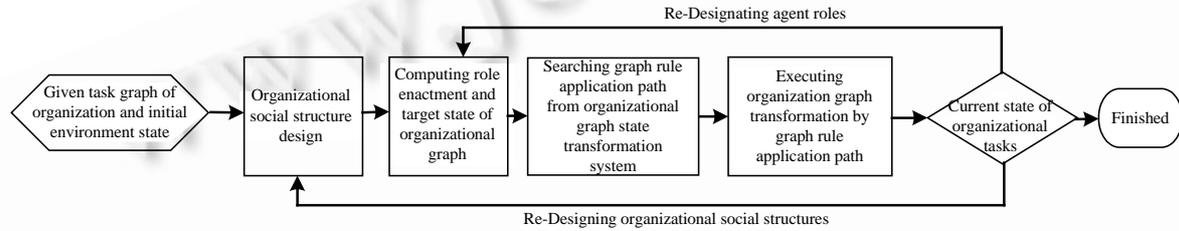


Fig.1 Reorganization process of organizational structures

图 1 组织结构的重组过程

在图 1 中, 组织的任务目标根据领域知识进行分解与合并形成任务图. 为简便起见, 每个任务节点对应一个角色, 因而可以由任务图导出组织的角色关系结构. 通过角色指定, 可以得到当前条件下的组织目标状态. 为了实施组织结构的变化过程, 搜索组织结构的图状态转移系统, 可以得出从初始状态到目标状态的组织图规则应用序列. 第 4 节详细介绍了我们实现的集中方式管理的层次图变换系统原型. 组织任务的状态可以采用集中或分布的方式进行检测. 在组织任务执行的过程中, 当组织任务无法完成时, 根据特定的重组策略, Agent 组织重新进行角色指定或组织结构的重新设计. 在重组过程中, 图规则应用的图匹配往往成为基于图变换方法应用的瓶颈. 通过单根节点和层次化地维护组织结构内元素的拓扑关系, 从而避免了大规模 Agent 组织重组中图匹配带来的图节点的组合爆炸问题. 定义 2.3.2 给出组织结构的层次图变换规则由角色关系结构、Agent 协作结构和角色指定关系图规则组合得到, 组织的图规则应用相应地包括以上 3 个方面的图规则应用. 考虑到图规则与宿主图之间采用单射和非单射具有等效模拟特性<sup>[16]</sup>, 为了简便起见, 本文考虑的图规则与宿主图之间的态射为单射. 下面给出了组织的图规则应用算法.

**算法 1.** 组织图规则应用过程算法 (arithmetic for organization graph rule application).

Input: Initial organization graph  $G_{org} = (G_{role}, G_{agent}, G_{assign}) \in \Gamma_{org}$ ; graph rule  $r = (\alpha, \beta, \gamma) \in \mathfrak{R}_{org}$

Output: Set of organization graph  $Set_{org} \subseteq \Gamma_{org}$

01. initialize morphism set of organization graph  $M = \{\}, Set_{org} = \{\}$

02. construct the sets of morphisms:  $M_{role}$  from  $\alpha$  to  $G_{role}$ ;  $M_{agent}$  from  $\beta$  to  $G_{agent}$  by Algorithm 2
03. construct the set of morphisms:  $M_{assign}$  from  $\gamma$  to  $G_{assign}$  by Algorithm 3
04. construct the set  $M$  of morphisms from  $r$  to  $G_{org}$ :  $M \leftarrow M_{role} \times M_{agent} \times M_{assign}$
05. for each  $m \in M$  do
06. if  $m$  不满足性质 2.3.1 then  $M = M \setminus \{m\}$  end if
07. end for
08. for each  $m = (m_{role}, m_{agent}, m_{assign}) \in M$  do
09. apply  $G_{role} \Rightarrow^{\alpha, m_{role}} G'_{role}$ ,  $G_{agent} \Rightarrow^{\beta, m_{agent}} G'_{agent}$ , and  $G_{assign} \Rightarrow^{\gamma, m_{assign}} G'_{assign}$
10. construct  $G'_{org} = (G'_{role}, G'_{agent}, G'_{assign})$ ,  $Set_{org} = Set_{org} \cup \{G'_{org}\}$
11. end for
12. return  $Set_{org}$

基于组织任务的层次约束,在 Agent 组织结构中角色关系结构相应地反映了这个层次关系.通过角色指定,Agent 之间实例化角色之间的层次约束并形成相互之间的委托与合作关系.性质 3.1 给出了角色关系结构与 Agent 协作关系结构的图规则满足的特性.

性质 3.1. 给定有向非循环图的规则  $p = (L, i, K, j, R)$ , 其中  $L, K$  和  $R$  是有向非循环图,在左边图  $L$  与宿主图进行图匹配时,  $L$  满足以下性质:

- (1) 按照定义 2.2.6 的组织层次,对图  $L$  的节点进行分层.其根节点集记为  $Rset(L)$ ,每个根节点层次设为 0,图  $L$  即划分为  $|Rset(L)|$  个连通子图.从相应子图的根节点到该子图的任意节点均可达;
- (2) 假定  $|Rset(L)| \leq k$ ,  $k$  为常数.  $k$  可以设为图规则集中最大的根节点集的势;
- (3) 在  $L$  任意节点  $v \in V_L \setminus Rset(L)$ , 存在一个节点  $u \in V_L$ , 满足  $(u, v) \in E_L$  且  $Lev(u) < Lev(v)$ .

当  $p$  为角色关系结构的图规则时,  $|Rset(L)| = 1$ . 下面给出了有向非循环图的图规则匹配算法.

算法 2. 有向非循环图的图规则匹配算法(arithmetic for graph rule match of directed acyclic graph).

Input: Initial directed acyclic graph  $G$ ; graph rule  $p = (L, i, K, j, R)$

Output: Set  $M$  of morphisms from  $L$  to  $G$

01. initialize the layers of  $L$  with Definition 2.2.6, its root set  $Rset(L)$  and  $M = \{\}$
02. construct partial morphisms only matching the nodes of  $Rset(L)$ :  $M_0 = \{M_{0,j}\}$  with  $M_{0,j}$  as a morphism from all root nodes in  $Rset(L)$  to  $G$
03. for each  $M_{0,j} \in M_0$  do
04. for each root node  $\mathcal{R} \in Rset(L)$  do
05. set  $E_0 = \{e_j\}$  with  $e_j$  outgoing edge from  $\mathcal{R}$
06. while any  $e \in E_0$  not matched do
07. if target node  $t(e)$  not matched then
08. extend  $M_{0,j}$  by the edge  $e$  and the node  $t(e)$  based on injective matching
09.  $E_0 \leftarrow (E_0 \setminus \{e\}) \cup \{\text{outgoing edges from } t(e)\}$
10. else
11. extend  $M_{0,j}$  by the edge  $e$  based on injective matching,  $E_0 \leftarrow E_0 \setminus \{e\}$
12. end if
13. end while
14. end for
15.  $M = M \cup M_{0,j}$
16. end for
17. check each morphism in  $M$  with dangling and identification conditions of double-pushout rule
18. return  $M$

由定义 2.2.4 可知,角色指定关系为不存在环路的二分图,是一类特殊的有向非循环图,也即在该图中所有边的方向均是由 Agent 节点指向角色节点,因而相应的图规则满足下面的性质 3.2.

性质 3.2. 给定二分图的规则  $p = (L, i, K, j, R)$ , 其中  $L, K$  和  $R$  是有向图, 在左边图  $L$  与宿主图进行图匹配时,  $L$  满足以下性质:

- (1) 按照定义 2.2.6 的组织层次, 对图  $L$  的节点进行分层. 将所有的 Agent 节点集记为根节点集  $Rset(L)$ , 每个根节点层次设为 0;
- (2) 图  $L$  的层次数等于 2, 也即对任意节点  $v \in V_L \setminus Rset(L)$ , 满足  $Lev(v) = 1$ ;
- (3) 对任意边  $e \in E_L$ , 存在节点  $u \in Rset(L)$ ,  $v \in V_L \setminus Rset(L)$ , 满足  $e = (u, v)$ .

基于上述性质, 角色指定关系的图规则匹配算法如下:

算法 3. 二分图的图规则匹配算法(arithmetic for graph rule match of bipartite graph).

Input: Initial bipartite graph  $G$ ; graph rule  $p = (L, i, K, j, R)$

Output: Set  $M$  of morphisms from  $L$  to  $G$

01. initialize the layers of  $L$  with Definition 2.2.6, its root set  $Rset(L)$  and  $M = \{\}$
02. construct partial morphisms only matching the nodes of  $Rset(L)$ :  $M_0 = \{M_{0,j}\}$  with  $M_{0,j}$  as a morphism from all root nodes in  $Rset(L)$  to  $G$
03. for each  $M_{0,j} \in M_0$  do
04. for each root node  $\mathcal{S} \in Rset(L)$  do
05. for each the node  $v \in V_L \setminus Rset(L)$  do
06. if  $v$  not matched then
07. if  $e = (\mathcal{S}, v) \in E_L$  and  $e$  not matched then
08. extend  $M_{0,j}$  by the edge  $e$  and the node  $v$  based on injective matching
09. else
10. extend  $M_{0,j}$  by the node  $v$  based on injective matching
11. end if
12. end if
13. end for
14. end for
15.  $M = M \cup M_{0,j}$
16. end for
17. check each morphism in  $M$  with dangling and identification conditions of double-pushout rule
18. return  $M$

在上述算法中, 图的节点匹配需要满足下面的条件: (1) 左边图节点的出度和入度分别小于对应宿主图节点的出度和入度; (2) 左边图节点对应的任务应与宿主图对应节点的任务相同. 图的边匹配需要满足左边图的边源节点和目的节点分别与宿主图的边源节点和目的节点相匹配. 由节点和边的匹配条件可知, 左边图在宿主图中最多有一个匹配的态射. 在算法复杂度方面, 假定初始图  $G$  的节点数为  $N$ , 边数为  $E$ , 算法 2 执行的最坏时间复杂度为  $\mathcal{O}(N^3 + NE(N + E))$ , 最坏空间复杂度为  $\mathcal{O}(N + E)$ . 相应地, 算法 3 的最坏时间复杂度为  $\mathcal{O}(N^2(N + E))$ , 最坏空间复杂度为  $\mathcal{O}(N + E)$ . 算法 1 的时间和空间复杂度取决于算法 2 和算法 3 的执行情况.

## 4 实验结果

我们基于二维平面网格环境下的追逐问题<sup>[18]</sup>及 Repast<sup>[19]</sup>仿真环境给出了原型系统的实验例子, 将基于组织的方法应用于追逐者之间的协作. 通过组织结构的演变实现追逐者对环境变化的自适应. 本文在实现 Agent 组织结构的层次图变换系统原型的基础上, 将该图变换系统反映为追逐者之间的组织结构演变. 实验原型的系

统结构如图 2 所示。

在图 2 中,图 2(a)显示了基于组织结构图状态的 Agent 系统的自省反射结构,其中的组织结构图状态维护了系统运行过程中所有 Agent 的状态.本文采用集中方式维护组织结构图状态,而在分布情况下,该状态分布于系统的各 Agent 个体中.同时,Agent 的行为变化也体现在组织结构图状态中.图 2(b)显示了基于这种自省的反射结构实现的 Agent 组织系统自适应结构,在监控模块中,采用集中方式对组织结构图状态及系统层 Agent 的任务执行情况进行监控,并将监控结果发送给自适应调节模块.在自适应调节模块中,实现利用当前环境条件预测下一时刻的组织目标状态,按照如图 1 所示的组织结构重组流程,在图状态转移系统中搜索从组织的当前状态到目标状态的一条图变换规则应用路径,利用该路径上的图变换规则依次修改组织结构图状态并反馈给系统层的 Agent 个体。

实验测试了 5 个追逐者和 1 个猎物的情景,将网格环境分成不同的区域,并设置不同数量的障碍.在无障的情况下,1 个猎物需要 4 个追逐者从 4 个不同的方向进行围堵时才能被捕.当在不同区域内时,根据障碍的分布可以只需 2 个或 3 个追逐者进行围堵.因而,当猎物在不同区域活动时,追逐者之间的组织角色结构发生了变化,通过重新作角色指定,相应的组织结构状态也发生变化.图 3 显示了由图 4 转变为图 5 过程的图规则,图 4 显示出猎物在无障区域 4 个追逐者分别承担 4 个不同方向的角色,图 5 显示出猎物在边缘区域时只有 1 个追逐者承担从猎物北方向的角色。

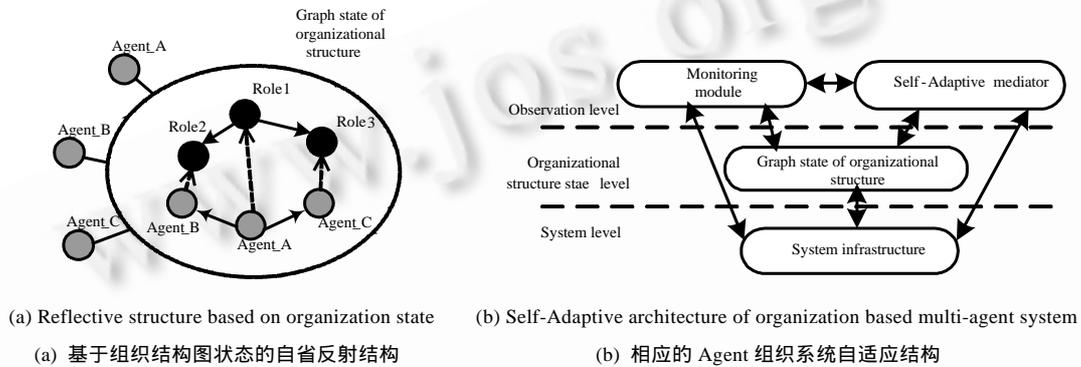
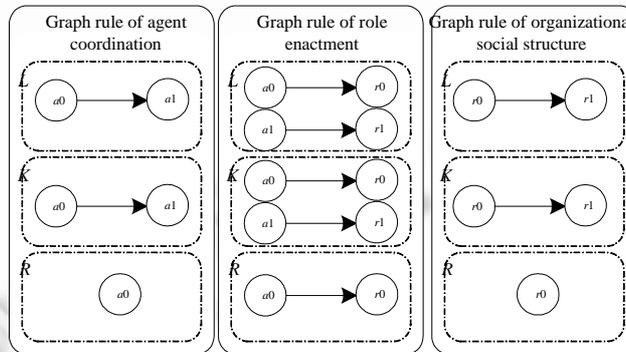


Fig.2 Experiment system prototype architecture

图 2 实验原型系统结构



Organization graph rule with deleting a role assigned to an agent

Fig.3 Graph rule from no obstacles to corner 1

图 3 从无障碍区域到边缘区域 1 情形的图规则示例

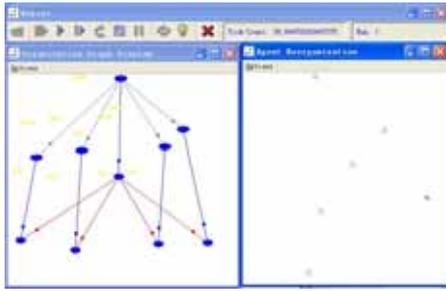


Fig.4 Predator's organization structure when preys near no obstacles

图 4 猎物在无障碍时追逐者组织结构状态

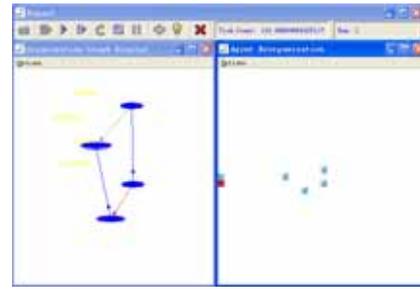


Fig.5 Predator's organization structure when preys near in corner 1

图 5 猎物在障碍边缘区域 1 时追逐者组织结构状态

我们将本文的算法 2 和算法 3 分别与 VF2<sup>[20]</sup>和 Ullman 算法<sup>[21]</sup>进行比较.为生成上述算法的测试数据,首先设置图的层数  $C_l$ 和每层最大节点数  $C_w$ ,通过随机方式生成图的节点,并根据满足性质 3.1 随机生成图节点之间的边.二分图作为有向非循环图的特例,在生成测试数据时,参数  $C_l$ 设置为 2.图 6 显示了有向非循环图的节点在由 0~1 000 范围变化时,上述算法的测试结果,图 7 显示了二分图的节点由 0~1 000 范围变化时上述算法的测试结果.由图 6 和图 7 可以看出:

(1) 算法 2 和算法 3 在计算时间上与 VF2 算法的计算性能接近,Ullman 算法随节点数的增长其计算时间有显著增加.

Ullman 算法和 VF2 算法都是图同态及子图同态的精确匹配算法.在 Ullman 算法中,首先建立左边图  $L$  与宿主图  $G$  节点的度比较矩阵  $M'=(m_{i,j})_{n_L \times n_G}$ ,其中  $n_L$  和  $n_G$  分别是图  $L$  和  $G$  的节点数.当左边图  $L$  节点  $i$  的度小于或等于宿主图  $G$  节点  $j$  的度时, $M'$ 中相应的元素  $m_{ij}=1$ ,否则, $m_{ij}=0$ .将矩阵  $M'$ 的元素按照行逐个进行匹配,相应地形成状态匹配树,树的每个节点对应  $M'$ 的每个元素匹配时的中间状态.Ullman 算法通过条件  $(\forall x)_{1 \leq x \leq n_L} ((a_{i,x}=1) \Rightarrow (\exists y)_{1 \leq y \leq n_G} (m_{x,y} \cdot b_{y,j}=1))$  判断图  $L$  的节点  $i$  与图  $G$  节点  $j$  的匹配状态,并对状态匹配树进行剪枝,其中  $a_{i,x}$  和  $b_{y,j}$  分别是图  $L$  和  $G$  对应的邻接矩阵的元素.在 VF2 算法中,利用深度优先策略,对状态匹配树进行遍历,与 Ullman 算法的不同点是,在状态匹配树的剪枝策略应用 5 类启发式规则,利用这些规则对问题空间的分支进行约简,并且 VF2 算法考虑了图节点之间的属性匹配.本文的算法 2 和算法 3 利用显式层次结构的组织结构图的单根节点和层次化的特性对图匹配问题空间的分支进行剪枝.与 VF2 相比,算法 2 和算法 3 更为直接、简便.其不足在于,本文的算法仅适用于具有层次性的图结构匹配.

(2) 随着图节点数的增长,算法的计算时间呈现一定的抖动.

由于 Ullman 算法中的矩阵  $M'$ 的元素是基于左边图  $L$  和宿主图  $G$  节点之间的度比较来构造的,因此,图的边数变化不改变矩阵  $M'$ 的计算量,但是,在状态匹配树的剪枝过程中,却影响了问题空间规模的约简.另一方面,当图的边数减小时,VF2 算法和本文的算法 2 和算法 3 在构建状态匹配树的过程中同时对问题空间进行化简.因此,当测试数据中图的边数减少时,Ullman 算法效率没有显著的提高,而 VF2 算法、算法 2 和算法 3 的效率变化明显.由于测试数据中图的节点和边是采用随机方式生成的,因而相应的算法计算时间呈现一定的抖动.

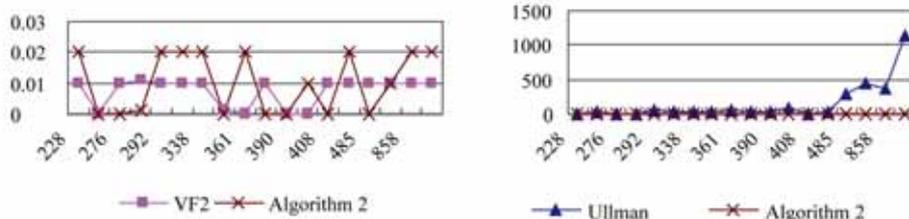


Fig.6 Comparison of Algorithm2, VF2 and Ullman based on directed acyclic graph match

图 6 基于有向非循环图匹配的算法 2,VF2 算法和 Ullman 算法比较结果

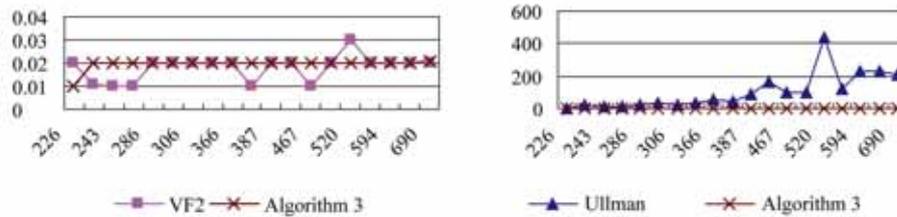


Fig.7 Comparison of Algorithm3, VF2 and Ullman based on bipartite graph match

图 7 基于二分图匹配的算法 3, VF2 算法和 Ullman 算法比较结果

## 5 总结和进一步的工作

本文基于层次图变换定义了多 Agent 组织结构动态重组过程语义.针对 Agent 组织结构的特点给出了描述组织结构的社会结构、角色指定和 Agent 协调的单根节点层次图模型,基于该模型,扩展 DPO 代数图变换,形式定义了组织的重组过程,并给出了重组过程的组织图变换规则操作满足的一致性条件及组织的图规则匹配关键算法.通过单根节点和层次化维护组织结构内元素拓扑,避免了大规模 Agent 组织的重组过程中图规则匹配时图节点的组合爆炸问题,有效地降低了大规模 Agent 组织重组过程的计算复杂性.未来的研究工作包括:Agent 组织动态重组过程中相关的矛盾处理,基于重组过程的大规模 Agent 组织快速协商技术;考虑重组过程中 Agent 组织维持组织目标实现的稳定性,组织内 Agent 的微观行为对组织行为的影响;图规则对图属性的扩展及大量的规则集合的处理等.

## References:

- [1] Peng H, Shi CY, Wang KL. Open distributed artificial intelligence. Journal of Software, 1995,6(10):607-613 (in Chinese with English abstract). <http://www.jos.org.cn/1000-9825/6/607.htm>
- [2] Dignum V, Dignum F, Sonenberg L. Towards dynamic reorganization of agent societies. In: Vouros G, ed. Workshop on Coordination in Emergent Agent Societies, ECAI 2004. Washington: IOS Press, 2004. 70-78.
- [3] Dignum F, Dignum V, Sonenberg L. Exploring congruence between organizational structure and task performance: A simulation approach. In: Coordination, Organization, Institutions and Norms in Agent Systems I, Proc. of the ANIREM 2005/OOP 2005. LNAI 3913, Berlin, Heidelberg: Springer-Verlag, 2006. 213-230.
- [4] Wang ZG, Liang XH, Zhao QP. Adaptive mechanisms of organizational structures in multi-agent systems. In: Shi Z, Sadananda R, eds. PRIMA 2006. LNAI 4088, Berlin, Heidelberg: Springer-Verlag, 2006. 471-477.
- [5] Hübner JF, Sichman JS, Boissier O. Using the MOISE+ for a cooperative framework of MAS reorganisation. In: Bazzan ALC, Labidi S, eds. Proc. of the 17th Brazilian Symp. on Artificial Intelligence (SBIA 2004). LNAI 3171, Berlin: Springer-Verlag, 2004. 506-515.
- [6] DeLoach SA, Oyen WH. An organizational model and dynamic goal model for autonomous, adaptive systems. Technical Report, MACR-TR-2006-01, Department of Computing and Information Sciences, Kansas State University, 2006.
- [7] So YP, Durfee EH. Designing tree-structured organizations for computational agents. Computational and Mathematical Organization Theory, 1996,2(3):219-246.
- [8] Xu JH, Zhang W, Shi CY, Hou BH. A structure-oriented mechanism of agent organization formation and evolution. Journal of Computer Research and Development, 2001,38(8):897-903 (in Chinese with English abstract).
- [9] Zhang W, Shi CY. A formal semantics of agent organization structure design. Journal of Software, 2002,13(3):447-452 (in Chinese with English abstract). <http://www.jos.org.cn/1000-9825/13/447.htm>
- [10] Zhan JF, Cheng H. Specification and evolution of agent from a perspective of software architecture. Journal of Computer Research and Development, 2002,39(12):1543-1549 (in Chinese with English abstract).

- [11] Durfee EH, Rosenschein JS. Distributed problem solving and multi-agent systems: Comparisons and examples. In: Klein M, ed. Proc. of the 13th Int'l Workshop on Distributed Artificial Intelligence (IWDAI'94). 1994. 94–104.
- [12] Zambonelli F, Jennings NR, Wooldridge M. Organizational abstractions for the analysis and design of multi-agent systems. In: Ciancarini P, Wooldridge MJ, eds. Proc. of the 1st Int'l Workshop on Agent-Oriented Software Engineering. LNAI 1957, Limerick: Springer-Verlag, 2001. 127–141.
- [13] Bryan H, Victor L. A survey of multi-agent organizational paradigms. The Knowledge Engineering Review, 2005,19(4):281–316.
- [14] Wang ZG, Liang XH, Zhao QP. A graph transformation system model of dynamic reorganization in multi-agent systems. In: Corchado E, *et al.*, eds. Proc. of the IDEAL 2006. LNCS 4224, Berlin, Heidelberg: Springer-Verlag, 2006. 1182–1190.
- [15] Corradini A, Montanari U, Rossi, Ehrig H, Heckel R, Löwe M. Algebraic approaches to graph transformation-Part I: Basic concepts and double pushout approach. In: Rozenberg G, ed. Handbook of Graph Grammars and Computing by Graph Transformation. Vol.1: Foundations. Singapore: World Scientific, 1997. 163–246.
- [16] Annegret H, Urgen J, Uller M, Plump D. Double-Pushout graph transformation revisited. mathematical structures. Computer Science, 2001,11(5):637–688.
- [17] Busatto G. An abstract model of hierarchical graphs and hierarchical graph transformation [Ph.D. Thesis]. Paderborn: Universitat Paderborn, 2002.
- [18] Tanev I, Brzozowski M, Shimohara K. Evolution, generality and robustness of emerged surrounding behavior in continuous predators-prey pursuit problem. Genetic Programming and Evolvable Machines, 2005,6(3):301–318.
- [19] Repast. 2001. <http://repast.sourceforge.net>
- [20] Cordella LP, Foggia P, Sansone C, Vento M. A (Sub) graph isomorphism algorithm for matching large graphs. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2004,26(10):1367–1372.
- [21] Ullman JR. An algorithm for subgraph isomorphism. Journal of the ACM, 1976,23(1):31–42.

## 附中文参考文献:

- [1] 胡蓬,石纯一,王克宏.开放的分布式人工智能.软件学报,1995,6(10):607–613. <http://www.jos.org.cn/1000-9825/6/607.htm>
- [8] 徐晋晖,张伟,石纯一,侯保华.面向结构的 Agent 组织形成和演化机制.计算机研究与发展,2001,38(8):897–903.
- [9] 张伟,石纯一. Agent 组织结构设计的一种形式语义.软件学报,2002,13(3):447–452. <http://www.jos.org.cn/1000-9825/13/447.htm>
- [10] 詹剑锋,程虎.基于软件体系结构的 Agent 规约和演化.计算机研究与发展,2002,39(12):1543–1549.



王正光(1980 - ),男,江西鄱阳人,博士生,主要研究领域为虚拟现实,人工智能.



赵沁平(1948 - ),男,博士,教授,博士生导师,CCF 高级会员,主要研究领域为虚拟现实,分布式应用系统,人工智能.



梁晓辉(1970 - ),男,博士,副教授,CCF 高级会员,主要研究领域为计算机图形学,虚拟对象建模方法.