

# 基于角色协同的角色扮演 逻辑及其代数模型

滕少华<sup>1</sup>, 张如琪<sup>1</sup>, 刘冬宁<sup>1</sup>, 刘林源<sup>2</sup>, 朱咸军<sup>3</sup>

(1. 广东工业大学计算机学院, 广东广州 510006; 2. 南京审计大学电子商务系, 江苏南京 211815;  
3. 南京大学工程管理学院, 江苏南京 210093)

**摘要:** 基于角色协同(Role-Based Collaboration, RBC)的软件工程与协同计算方法以角色为基础进行研究,能有效促进信息系统的抽象与分类,有利于单元协作与协同计算.为支撑RBC及其角色引擎,本文从元理论角度出发,运用子结构逻辑对RBC的关键元素:群组、角色和代理进行了考量与印证,并提出了系列多项式时间复杂度的角色扮演逻辑系统(Role Playing Logic, RPL)及相关代数模型.通过逻辑强刻画与形式化方法对RBC系统基本关联性和层次结构的剖析,系列角色扮演逻辑可因不同的设计需求做扩充与变化,以支撑RBC做各类工程应用与研发.

**关键词:** 角色协同; 群组; 角色; 代理; 角色扮演逻辑

**中图分类号:** TP301 **文献标识码:** A **文章编号:** 0372-2112 (2017)12-3012-07

**电子学报 URL:** <http://www.ejournal.org.cn> **DOI:** 10.3969/j.issn.0372-2112.2017.12.025

## Role Playing Logic and Its Algebra Model in Role-Based Collaboration

TENG Shao-hua<sup>1</sup>, ZHANG Ru-qi<sup>1</sup>, LIU Dong-ning<sup>1</sup>, LIU Lin-yuan<sup>2</sup>, ZHU Xian-jun<sup>3</sup>

(1. School of Computer Science and Technology, Guangdong University of Technology, Guangzhou, Guangdong 510006, China;  
2. Department of Electronic Commerce, Nanjing Audit University, Nanjing, Jiangsu 211815, China;  
3. School of Management and Engineering, Nanjing University, Nanjing, Jiangsu 210093, China)

**Abstract:** Role-based collaboration (RBC) is a software engineering and cooperative computing method. It can extract and classify information systems. Therefore, it can be applied to the fields such as system unit collaborative computing and working. From meta-theoretical perspective, the specification of groups, roles and agents is a critical element of RBC. In consideration of the relationships and hierarchies faced by groups, roles and agents, we propose a series of role playing logic systems (RPL) and their algebra models through by using substructural logic. The proposed logic systems are polynomial in complexity and in support of RBC. One can extend RPL systems and models to suit other appropriate applications, to support RBC to do all kinds of engineering applications and research development.

**Key words:** role-based collaboration; group; role; agent; role playing logic

## 1 引言

基于角色协同(RBC)是由加拿大学者朱海滨提出的一种软件工程与协同计算方法,其以角色(Role)为基础进行研究,能有效促进信息系统的抽象与分类,从而有利于系统单元协作与协同计算.目前,RBC及其计算模型E-CARGO已被广泛应用于组织管理、工业工程与生产管理等多个领域<sup>[1]</sup>.

角色引擎(role engine)是构建RBC系统并进行分

析与设计的关键,它类似于Prolog推理机<sup>[2]</sup>(在使用Prolog系统时,人们只需写规则和事实,Prolog推理机就能搜索出结果).在RBC中,群组(group)、角色、代理(agent)及其之间的关系由角色引擎指定.角色引擎为代理协同提供平台,在该平台上,代理遵循一定的生产关系,通过被指派角色并执行该角色来为系统工作.为使RBC系统有效工作,所有这些活动都必须形式化和符号化.

国内外很多学者都讨论过角色和代理的形式

收稿日期:2016-11-11;修回日期:2016-12-21;责任编辑:覃怀银

基金项目:国家自然科学基金(No. 61402118);广东省产学研合作项目(No. 2015B090901016);广州市科技计划项目(No. 2016201604030034);江苏省自然科学基金(No. BK20130735);江苏省高校自然科学研究项目(No. 13KJB520011)

化<sup>[3-5]</sup>,这些方法大多是基于 CASE 工具的,如类图、对象图等,容易被设计师和程序员所理解.但在元理论上,却缺乏对 RBC 及其构成元素(群组、角色和代理)的形式化和综合性的数学讨论,从而无法解释 RBC 的高层次抽象,更无法揭示 RBC 的主要元素所存在的最基本的关联性和层次结构.另一方面,尽管当前存在着大量的多代理系统(Multi-Agent System,简称 MAS)形式化方法,如 BDI 逻辑<sup>[6-8]</sup>,认知逻辑<sup>[9]</sup>,动态逻辑<sup>[10]</sup>等,但这些方法大多是针对代理自身的认知和推理的.相反,系统性地探讨角色和角色引擎的研究很少.

因此,本文以子结构逻辑(Sub-structural Logic)及其代数模型为基础,对 RBC 及其计算模型 E-CARGO 进行了元理论建模与性质证明,提出了系列时间复杂度仅为多项式时间的角色扮演逻辑系统(Role Playing Logic,简称 RPL).RPL 系统能对 RBC 做逻辑深刻刻画,且能做多种扩充与变化,以适应并支撑各类工程应用与研发.

选择子结构逻辑及其代数模型(偏序半群)作为形式化工具的主要原因是:首先,本文强调的是检验 RBC 主要元素(群组、角色和代理)之间最基本的关联性和层次结构,因此我们希望在构建 Gentzen's 风格的极小的逻辑系统<sup>[11-13]</sup>时,能用最少的逻辑连接词和结构规则来描述角色扮演(代理的认知和推理并非本文重点);其次,相对于句法逻辑,偏序半群作为模型语义,可以抽象地反映角色指派,并用于创建、删除和修改角色、代理和群组,而所有这些活动都直接关系到角色引擎的设计.

根据 E-CARGO 模型,本文假设一个角色可以由多个代理来扮演,但一个代理一次只能扮演一个角色.在前期工作中,刘冬宁等<sup>[14]</sup>提出了基于 RBC 的极小角色扮演逻辑系统,文献[15]针对诊疗排班系统的具体应用进行了阐述和分析.在此基础上,本文进一步对角色扮演逻辑进行了论证,并完成了相关逻辑与代数证明,从元理论角度剖析了 RBC 的主要元素所存在的最基本的关联性和层次结构.

## 2 基于角色协同的角色扮演逻辑

### 2.1 E-CARGO 模型

在 E-CARGO 中,系统  $\Sigma$  被描述为九元组  $\Sigma ::= \langle C, O, \mathcal{A}, \mathcal{M}, \mathcal{R}, \mathcal{E}, \mathcal{G}, s_0, \mathcal{H} \rangle$ . 其中,  $C$  是类的集合,  $O$  是目标对象的集合,  $\mathcal{A}$  是代理的集合,  $\mathcal{M}$  是消息的集合,  $\mathcal{R}$  是角色的集合,  $\mathcal{E}$  是环境,  $\mathcal{G}$  是群组的集合,  $s_0$  是协同系统的初始状态,  $\mathcal{H}$  是用户的集合<sup>[16]</sup>. 据 E-CARGO 模型,软件开发被认为是基于角色的协同活动.  $\Sigma$  则被视为一个软件开发团队:在同一环境中,随着人类用户  $\mathcal{H}$  的参与,加入一个团队  $\Sigma$ , 访问团队的对象,通过角色发

送消息,构建群组. 团队工作的结果是一个由  $C, O, \mathcal{A}, \mathcal{M}, \mathcal{E}, \mathcal{G}$  和  $\mathcal{H}$  值表示的  $\Sigma$  的新状态,其中  $\langle C, O, \mathcal{M}, \mathcal{R} \rangle$  形成基于角色的软件产品. 在 E-CARGO 中,代理被认为只有一个中央处理单元,在一个时间点上只能处理消息或最多扮演一个角色.

在 E-CARGO 模型中,群组是在由角色和对象构成的共同环境中工作的代理集合. 作为系统需求和设计规范的发展基础,良定义角色引擎并形式化角色、代理,能为软件的管理、设计与实现形成支撑并带来效益.

### 2.2 极小角色扮演逻辑

本文重点在于揭示角色和代理之间最基本的关联性和层次结构,并使用子结构逻辑作为形式化工具进行阐述. 为与传统的代理逻辑区分开来,本文将新建立的系列逻辑系统称为角色扮演逻辑. 在系列角色扮演逻辑系统中,存在一极小系统,其 Gentzen 式演算表达如下:

公理

$$p \Rightarrow p(\text{id})$$

结构规则

$$\frac{\Delta \Rightarrow A \quad \Gamma, A, \Pi \Rightarrow B}{\Gamma, \Delta, \Pi \Rightarrow B} (\text{cut})$$

操作规则

$$\frac{\Gamma, A, B \Rightarrow C}{\Gamma, A \bullet B \Rightarrow C} (\bullet L) \quad \frac{\Gamma \Rightarrow A \quad \Delta \Rightarrow B}{\Gamma, \Delta \Rightarrow A \bullet B} (\bullet R)$$

其中,用小写字母  $p, q, r \dots$  表示代理(原子公式),用大写字母  $A, B, C \dots$  表示由“ $\bullet$ ”连接的代理组合(命题公式). 因为主要研究的是群组、角色和代理之间基本的关联性和层次结构,所以选择忽略代理的内部细节. 在此基础上,将  $\Gamma, \Delta$  和  $\Pi$  表示为代理的序列性多重集(sequential multiset). 例如“ $\Delta \Rightarrow A$ ”表示代理  $A$  在代理序列性多重集  $\Delta$  中是协同工作的. 同时假设在群组中,代理最基本的关联性是协同,因此使用连接词“ $\bullet$ ”作为对代理协同工作的抽象.

实际上,结构规则中的(cut)规则是可以被消除的. 切割消除定理是形式逻辑系统中一项非常重要的属性,也是证明该系统可判定性的关键. 这表明,没有切割消除定理的逻辑系统,就相当于没有发动机的汽车. 注意到该逻辑系统除了一个连接符“ $\bullet$ ”(和它的左,右规则)外,没有其他任何结构规则. 这是因为本文要将角色和代理之间的关联做到“最简单、最抽象”,并称该逻辑为极小角色扮演逻辑(MRPL). 以下证明,切割消除定理在 MRPL 中是成立的,MRPL 是可判定的,且其时间复杂度是多项式时间(P-time).

**定义 1**(切割度). 令  $d(x)$  表示类型公式  $A$  中出现的连接词“ $\bullet$ ”的数量. 且  $d(A_1, A_2, \dots, A_n) = d(A_1) + d(A_2) + \dots + d(A_n)$ . 则一个切割的度为:  $d(\Delta) + d(\Gamma)$

$+d(\Pi) + d(A) + d(B)$ .

**定理 1** (切割消除定理). 在系统的任意切割中, 当所有切割的前提不是由切割推演得到时, 该切割的结论存在且只存在两种可能: 该切割与结论前提中的某一个一致; 该切割可以被一个或两个切割度更小的切割替换.

证明 施归纳于证明步骤, 可分五种情况讨论.

(1)  $\Delta \Rightarrow A$  是公理的实例;

(2)  $\Gamma, A, \Pi \Rightarrow B$  是公理的实例;

(3) 证明  $\Delta \Rightarrow A$  的最后一步推演中没有引入  $A$  中的主连接词;

(4) 证明  $\Gamma, A, \Pi \Rightarrow B$  的最后一步推演中没有引入  $A$  中的主连接词;

上述 4 种情况的切割消除证明与文献 [17] 中的经典子结构逻辑系统 Lambek 演算相似, 我们需要证明的是第五种情况:

(5) 两个切割前提推演的最后一步都引入了  $A$  的主连接词“ $\bullet$ ”, 即  $A = A' \bullet A''$ . 可将原切割

$$\frac{\frac{\Gamma, A' \text{ (or } A'') , \Pi \Rightarrow B}{\Gamma, A' \bullet A'' , \Pi \Rightarrow B} (\bullet L) \quad \frac{\Delta \Rightarrow A' \quad \Delta \Rightarrow A''}{\Delta \Rightarrow A' \bullet A''} (\bullet R)}{\Gamma, \Delta, \Pi \Rightarrow B} (\text{cut})$$

置换为

$$\frac{\Delta \Rightarrow A' \text{ (or } A'') \quad \Gamma, A' \text{ (or } A'') , \Pi \Rightarrow B}{\Gamma, \Delta, \Pi \Rightarrow B} (\text{cut})$$

显然, 新的切割比原切割有更小的度.

据情况 (1) ~ (5), 证明了在 MRPL 系统中, 切割规则是可以被消除的 (切割消除定理). 证毕.

根据切割消除定理, 在上述系统中, 给定一个序列  $\Gamma \Rightarrow A$ , 可尝试对其构造一棵自底向上的证明树, 该树的每一分支为一证明. 对于该树的构造, 可使用 ( $\bullet L$ ) 和 ( $\bullet R$ ) 规则. 在树中, 每一个向上的步骤, 都能消除连接词“ $\bullet$ ”的一次出现, 且这样的回溯总是有穷的, 因此该树是有穷的. 据此, 序列  $\Gamma \Rightarrow A$  是可判定的当且仅当该树的某一分支能被证明.

据此可进一步证明上述逻辑的可判定性证明的时间复杂度是多项式时间的.

**引理 1**  $\vdash_{\text{MRPL}} \Gamma \Rightarrow A$  iff 存在  $p$ , 使得  $F(\Gamma) = p \bullet F(\Gamma')$ ,  $A = p \bullet A'$  and  $\Gamma' \Rightarrow A'$ , 其中  $F(\Gamma)$  指的是从结构公式  $\Gamma$  中通过将“ $,$ ”替换为“ $\bullet$ ”而得到的公式.

**证明** 从右到左的证明是平凡的, 以下证明从左到右.

(1)  $\Gamma \Rightarrow A$  是公理, then  $\Gamma = A = p$ ;

(2)  $\Gamma \Rightarrow A$  是通过 ( $\bullet L$ ) 从前提  $\Gamma_1, B, C \Rightarrow A$  中得到. 据归纳假设, 存在  $p$ , 使得  $F(\Gamma_1, B, C) = p \bullet F(\Gamma'_1, B, C)$ ,  $A = p \bullet A'$  and  $\Gamma'_1, B, C \Rightarrow A'$ . 通过 ( $\bullet L$ ), 得到  $\Gamma'_1, B \bullet C \Rightarrow A'$ ;

(3)  $\Gamma \Rightarrow A$  是通过 ( $\bullet R$ ) 从前提  $\Gamma_1 \Rightarrow A_1$  and  $\Gamma_2 \Rightarrow A_2$  中得到的. 据归纳假设, 存在  $p$ , 使得  $F(\Gamma_1) = p \bullet F(\Gamma'_1)$ ,  $A_1 = p \bullet A'_1$  and  $\Gamma'_1 \Rightarrow A'_1$ . 则  $F(\Gamma) = p \bullet F(\Gamma'_1) \bullet F(\Gamma_2) = p \bullet F(\Gamma'_1, \Gamma_2)$  且  $B = p \bullet A'_1 \bullet A_2$ . 证毕.

**定理 2** (可判定性时间复杂度的证明). MRPL 可判定性的证明的复杂度为 P-time.

**证明** 由引理 1, 可构建以下算法判定序列  $\Gamma \Rightarrow A$  是否可证.

(1) 令  $C_1 = A$  且  $C_2 = F(\Gamma)$ ;

(2) 若  $C_1, C_2$  为空, 则返回 true, 否则选择出现在  $C_1$  的第一个原子公式  $p$ , 同理, 可检验  $C_2$ ;

(3) 若  $p$  是出现在  $C_2$  的第一个原子公式, 则  $C_1 = C_1 - p, C_2 = C_2 - p$ , 且返回 Step 2, 否则返回 false. 证毕.

显然, 上述算法的复杂度是  $O(n)$ , 因此该过程是 P-time 的.

### 3 代数模型

#### 3.1 极小角色扮演逻辑的半群模型

MRPL 系统是句法层面上的表示, 可表达逻辑后承. 在语义层面, 本文将构造其半群模型, 并证明了 MPRL 相对于该半群是可靠的和完全的.

**定义 2** 偏序半群的代数结构为  $G = \langle \mathcal{A}, \bullet, \leq \rangle$ , 其中,  $\langle \mathcal{A}, \leq \rangle$  是偏序集,  $(\mathcal{A}, \bullet)$  是半群.

可靠性和完全性是逻辑和模型理论中的首要性质, 用于作为判断该逻辑系统是否符合语义模型的依据.

**定义 3** (模型指派).  $\mu$  是 MRPL 对  $G = \langle \mathcal{A}, \bullet, \leq \rangle$  的一个指派, 使得 MRPL 中所有的原子公式  $p$ , 都有  $\mu(p) = a_p$ , 其中  $a_p \in A$ , 且通过归纳 MRPL 中的所有公式  $A, \mu(A) = a_{p_1} \bullet a_{p_2} \bullet \dots \bullet a_{p_n}$ , 其中  $p_1, p_2, \dots, p_n$  是  $\mathcal{A}$  中连续出现的原子公式.

**定理 3** (可靠性定理). MRPL 相对于  $G = \langle \mathcal{A}, \bullet, \leq, \mu \rangle$  是可靠的.

**证明** 据定理 2,  $\Gamma \Rightarrow A$  是 MRPL 中的一个逻辑后承, 当且仅当  $\forall p_i$  在  $A$  中连续出现, 并在  $\Gamma$  中连续出现. 据此可得:  $\mu(\Gamma) = \mu(F(\Gamma)) = \mu(A)$ . 证毕.

**定理 4** (完全性定理). MRPL 相对于  $G = \langle \mathcal{A}, \bullet, \leq, \mu \rangle$  是完全的.

**证明** 设  $\mu(\Gamma) = \mu(A)$ , 施归纳公式  $A$  复杂度的证明  $\Gamma \Rightarrow A$ .

(1) 若  $A = p$ , 且  $\mu(\Gamma) = \mu(A) = \mu(p)$ , 则  $\Gamma = p$  and  $\Gamma \Rightarrow p$ ;

(2) 若  $A = B \bullet C$ , 且存在  $\Gamma_1$  和  $\Gamma_2$ , 其中  $\Gamma = \Gamma_1, \Gamma_2$ , 使得  $\mu(\Gamma) = \mu(\Gamma_1) \bullet \mu(\Gamma_2)$ . 据归纳假设,  $\Gamma_1 \Rightarrow B$  且  $\Gamma_2 \Rightarrow C$ , 由 ( $\bullet R$ ) 得  $\Gamma \Rightarrow A$ . 证毕.

#### 3.2 MRPL 的代数语义

作为一个抽象概念, 偏序半群  $G = \langle \mathcal{A}, \bullet, \leq, \mu \rangle$  具

有三个层次的应用意义.

首先,关于模型指派  $\mu$ , MRPL 相对于  $G$  是可靠的和完全的. 实际上,  $G = \langle \mathcal{A}, \bullet, \leq \rangle$  是进行角色扮演的代理的代数结构, 由偏序集  $\langle \mathcal{A}, \bullet, \leq \rangle$  和运算符“ $\bullet$ ”组成的. 对于  $\mathcal{A}$  中的任意元素, 例如  $a, b, c \dots$  等, 分别指代了代理  $a, b, c \dots$ , 而  $a, b, c \dots$  的成分和值实际上是代理的内部细节 (内涵). 这是 MRPL 代数模型在全局层面的意义.

其次, RBC 中, 代理是角色扮演者. 在代数逻辑中, 角色提供具体流程规范, 以约束代理的行为. 且角色的模型具有与代理相同的代数类型, 表示为  $G' = \langle \mathcal{R}, *, \leq \rangle$ . 其中,  $G'$  是半群,  $\leq$  是  $\mathcal{R}$  的偏序结构,  $\langle \mathcal{R}, \leq \rangle$  是偏序集, “ $*$ ”是  $\mathcal{R}$  的二元运算符 (类似  $\mathcal{A}$  中的“ $\bullet$ ”). 基于此, 可将角色指派表述为:  $\nu: \mathcal{R} \rightarrow \mathcal{A}$ . 在 RBC 的角色指派中, 假设一个代理只能扮演一个角色, 但一个角色可被多个代理所扮演. 故  $\nu: \mathcal{R} \rightarrow \mathcal{A}$  并非代数意义上的函数 (因为对于定义域中的一个元素, 值域中有多个元素与其对应). 但相反地, 取其反  $\pi: \mathcal{A} \rightarrow \mathcal{R}$  则是一个函数 (对于空闲的代理, 可在  $\mathcal{R}$  中假拟一虚角色, 并将空闲代理指派给了虚角色, 则能使该函数定义域中任意元素都能在值域中找到一个元素与之对应且唯一对应). 因此, 角色指派可看作是  $G = \langle \mathcal{A}, \bullet, \leq \rangle$  到  $G' = \langle \mathcal{R}, *, \leq \rangle$  的函数, 该函数是同态映射的. 对于任何  $a, b \in \mathcal{A}, \pi(a) \bullet \pi(b) = \pi(a * b)$ , 都能由规则 ( $\bullet$ L) 和 ( $*$ L) 得到. 此外, 该函数还是满射的, 因为建立在良定义角色引擎基础上的系统, 每个角色都必须有一个扮演者, 这就意味着每个角色都必须指派至少一个代理.

这是代数中一个非常重要的性质, 当  $\pi$  是满射同态映射时, 可确保运算符“ $\bullet$ ”和“ $*$ ”均满足交换律、幂等律、可逆性、分配律、幺元律和同一性. 因此 MRPL 代数模型能有效衔接角色和代理, 起到衔接层面的意义.

第三, MRPL 还具有操作层面的意义.  $\mathcal{A}$  中相互关联的元素和协同运算符“ $\bullet$ ”组成 RBC 的群组. 在数域  $\mathcal{A}$  范围内,  $G = \langle \mathcal{A}, \bullet, \leq \rangle$  是最大的 RBC 群组, 且对于  $\mathcal{A}$  中的任何元素  $a, b, c \dots$ , 若  $a \bullet b \bullet c \dots$ , 则它们是 E-CARGO 模型在  $G$  中一个协同工作的群组. 以下结合案例对上述模型进行阐述.

### 3.3 案例分析

某 X 通信公司负责 S 省移动基站代维业务, 为移动基站运营提供保障. 由于基站的物理分布特性, 基站代维工作量大、区域范围广, 工作内容包括各类故障处理、投诉处理、应急保障、设备巡检、抢修发电等. 目前 X 公司某驻点收到了新一季度运维任务与人员需求 (体现为人数需求) 如表 1 所示:

根据人力资源部门评估, 每个技术工人对各岗位技能评分如表 2 所示:

表 1 岗位需求表

| 岗位 | 基站设备 | 铁塔天馈 | 直放站 | 传输线路 | 集客家宽 |
|----|------|------|-----|------|------|
| 需求 | 1 人  | 2 人  | 2 人 | 2 人  | 3 人  |

表 2 岗位技能评分与志愿表

| 岗位<br>工人 | 基站设备       | 铁塔天馈        | 直放站         | 传输线路        | 集客家宽        |
|----------|------------|-------------|-------------|-------------|-------------|
| 李峰       | 0.97       | 0.49        | 0.54        | <b>0.99</b> | 0.54        |
| 陈江       | <b>0.9</b> | 0.52        | 0.49        | 0.47        | 0.81        |
| 赵强       | 0.24       | 0.69        | 0.46        | <b>0.95</b> | 0.87        |
| 王军       | 0.05       | 0.54        | 0.2         | 0.45        | 0.61        |
| 段俊       | 0.85       | 0.78        | 0.25        | 0.4         | <b>0.98</b> |
| 曾溪       | 0.62       | 0.65        | 0.51        | 0.38        | 0.8         |
| 于梁       | 0.73       | 0.65        | 0.79        | 0.91        | 0.84        |
| 李小璐      | 0.3        | 0.73        | <b>0.82</b> | 0.43        | 0.83        |
| 向青       | 0.77       | 0.38        | 0.69        | 0.87        | <b>0.91</b> |
| 李玲       | 0.65       | 0.88        | 0.63        | 0.69        | 0.85        |
| 朱辉       | 0.39       | 0.57        | <b>0.76</b> | 0.94        | 1.0         |
| 万郑       | 0.7        | 0.59        | 0.71        | 0.68        | 0.68        |
| 李刚       | 0.56       | 0.12        | 0.65        | 0.52        | 0.75        |
| 武娟       | 0.42       | 0.85        | 0.75        | 0.59        | 0.33        |
| 李洁       | 0.53       | 0.78        | 0.59        | 0.79        | 0.27        |
| 马天       | 0.26       | <b>0.92</b> | 0.69        | 0.31        | 0.16        |
| 袁强       | 0.71       | 0.77        | 0.62        | 0.75        | 0.37        |
| 邓思思      | 0.64       | <b>0.96</b> | 0.75        | 0.83        | 0.94        |
| 姚华       | 0.06       | 0.8         | 0.71        | 0.47        | 0.92        |
| 施文华      | 0.29       | 0.81        | 0.71        | 0.58        | <b>0.99</b> |

据此, 可形成代理偏序半群  $G = \langle \mathcal{A}, \bullet, \leq \rangle$ , 其中  $\mathcal{A}$  为表 2 中人员集合. 首先据全局层, “李峰”、“陈江”、“赵强”……“姚华”、“施文华”等为各代理, 其在每个角色“岗位”上的分值为其模型赋值 (指派  $\mu$ ), 即为其内涵, 通常由人力资源部门负责评估. “基站设备”、“铁塔天馈”、“直放站”、“传输线路”、“集客家宽”等形成角色集  $\mathcal{R}$ , 角色需求数为其模型赋值.

其次, 据衔接层, 代理是角色扮演者, 每个角色及其需求数均需要满足. 即将  $\mathcal{R}$  指派给  $\mathcal{A} (\nu: \mathcal{R} \rightarrow \mathcal{A})$ , 然而这并非是一函数. 根据本例, 如果一个代理只能代理一个角色, 那么根据角色需求数, 只有 10 名代理需参加本驻点工作, 另有 10 名代理不安排工作. 但其反向  $\pi: \mathcal{A} \rightarrow \mathcal{R}$  则为函数, 并为满射. 例如在本例中, 可将代理指派如表 2 中 **带框所示** 代理, 其 **带框分值** 为被指派角色. 通常用作指派的算法为匈牙利算法 (Kuhn-Munkres 算法)、线性规划算法等<sup>[18-20]</sup>.

再次,在操作层,据本例指派可得出“李峰 • 陈江 • 赵强 • 段俊 • 李小璐 • 向青 • 朱辉 • 马天 • 邓思思 • 施文华”为协同工作群组. 其中,运算符“•”是语义模型和句法逻辑的核心. 其所具备的运算性质将影响逻辑系统的运算结构与关系,最终施影响于协同工作. 针对此,将在操作层面对 MRPL 进行扩展.

### 4 MRPL 扩展

#### 4.1 结构规则扩展

本文对 MRPL 系统的结构规则扩展主要体现在交换规则、弱化规则与收缩规则上,形如表 3 所示:

表 3 结构规则和及其代数性质

| 结构规则<br>Structure Rules | 代数性质<br>Algebra Properties                                  |
|-------------------------|---|
| 弱化<br>(weakening)       | $a \bullet b \leq a$ and $a \bullet b \leq b$<br>(integral) |
| 收缩<br>(contraction)     | $a \leq a \bullet a$<br>(contractive)                       |
| 交换<br>(exchange)        | $a \bullet b = b \bullet a$<br>(commutative)                |

如果逻辑系统没有弱化规则,那么它是非单调的,且所有的计算资源都必须至少用一次. 反之亦然,如果逻辑系统有弱化规则,则它是单调的. 收缩规则是弱化规则的逆规则. 如果逻辑系统没有收缩规则,那么所有的计算资源都至多用一次. 反之,如果将收缩规则添加到逻辑系统中,则若初始时,由  $\Gamma$  和两个  $\Delta$  可推导出  $A$ , 那么由  $\Gamma$  和  $\Delta$ , 同样可以推导出  $A$ , 这意味着资源  $\Delta$  是可重复使用的. 特别地,如果逻辑系统既没有弱化规则,也没有收缩规则,那么所有的计算资源都至少用一次,且至多用一次,因此其对资源是相当敏感的,该逻辑系统被称为资源敏感逻辑系统.

弱化规则与收缩规则体现了在协作过程中,代理作为一种资源的敏感性(如果协作的代理是人,则可推断该人是否是多余的,对其使用是否可以不计次数等). 而交换规则体现了协同的灵活性,  $a \bullet b = b \bullet a$  意味着代理间的协作顺序是可调换的.

弱化规则,收缩规则和交换规则分别对应代数中的 integral、contractive 和 commutative 性质. 由此组合产生的代数分别称为 Integral-偏序半群(W), Contractive-偏序半群(C), 偏序阿贝尔半群(E), Integral-偏序阿贝尔半群(EW), Contractive-偏序阿贝尔半群(EC), 和 Integral-Contractive-偏序半群(CW).

可简证得,假设某代数存在 integral 与 contractive 性质,则 commutative 性质也成立. 此外,Integral-Contractive-偏序半群还是一个上交半格. 因此,根据 ECW 的组

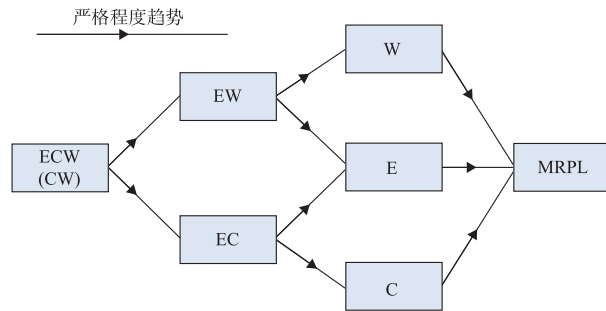


图1 结构规则组合成的角色扮演逻辑

合,可以得到 7 种角色扮演逻辑,如图 1 所示. 对于这些模型的变换,根据从左到右的箭头,系统在代数逻辑上,变得更加严格,规则越少. 而针对不同的协同应用,应需求适当放松这些约束.

#### 4.2 性质证明

上述的角色扮演逻辑(RPL)均是可切割消除,且时间复杂度为 P-time,相对于规则对应的偏序半群是可靠的和完全的. 在前文中,我们已证明了最严格的极小系统 MRPL 性质,以下对其相对最宽松的满足交换、弱化和收缩的 RPL<sub>ECW</sub> 系统的性质做例证(限于篇幅,对中间各系统则略去证明).

我们证明 RPL<sub>ECW</sub> 系统相对于其半群是可靠的、完全的,且这两个属性是框架可靠的和框架完全的(相对于 MRPL 的模型可靠性和完全性,框架可靠性和完全性更为严格).

RPL<sub>ECW</sub> 句法逻辑系统如下:

公理

$$p \Rightarrow p(\text{id})$$

结构规则

$$\frac{\Delta \Rightarrow A \quad \Gamma, A, \Pi \Rightarrow B}{\Gamma, \Delta, \Pi \Rightarrow B} (\text{cut})$$

$$\frac{\Gamma, A, B, \Delta \Rightarrow C}{\Gamma, B, A, \Delta \Rightarrow C} (\text{exchange})$$

$$\frac{\Gamma \Rightarrow A}{\Gamma, \Delta \Rightarrow A} (\text{weakening}) \quad \frac{\Gamma, \Delta, \Delta \Rightarrow A}{\Gamma, \Delta \Rightarrow A} (\text{contraction})$$

操作规则

$$\frac{\Gamma, A, B \Rightarrow C}{\Gamma, A \bullet B \Rightarrow C} (\bullet L) \quad \frac{\Gamma \Rightarrow A \quad \Delta \Rightarrow B}{\Gamma, \Delta \Rightarrow A \bullet B} (\bullet R)$$

定义 4 RPL<sub>ECW</sub> 的偏序半群的代数结构为  $G = \langle \mathcal{A}, \bullet, \leq \rangle$ , 其中,  $\langle \mathcal{A}, \leq \rangle$  是偏序集,  $(\mathcal{A}, \bullet)$  是半群, 对于任意的  $a, b, c \in \mathcal{A}$ , 均满足:

表 1 仿真参数.

|   |               |
|---|---------------|
| $a \bullet b = b \bullet a$                         | (commutative) |
| $a \leq a \bullet a$                                | (contractive) |
| $a \bullet b \bullet a$ and $a \bullet b \bullet b$ | (integral)    |

引理 2 RPL<sub>ECW</sub> 的代数半群中,  $c \leq a$  and  $c \leq b$  iff  $c$

$\leq a \bullet b$  iff  $c \leq b \bullet a$ .

**证明**

(1)  $c \leq a \bullet b$  iff  $c \leq b \bullet a$ , 由交换规则可得;

(2)  $c \leq a$  and  $c \leq b$  iff  $c \leq a \bullet b$ .

左 $\Rightarrow$ 右.  $\because c \leq a$  and  $c \leq b, \therefore c \bullet c \leq a \bullet b$ , 根据 contractive,  $c \leq c \bullet c$ , 所以有  $c \leq a \bullet b$ ;

右 $\Rightarrow$ 左.  $\because a \leq a$ , 通过 integral,  $a \bullet b \leq a$ , 又  $\because c \leq a \bullet b, \therefore c \leq a$ . 同理,  $c \leq b$ .

根据(1)、(2),  $c \leq a$  and  $c \leq b$  iff  $c \leq a \bullet b$  iff  $c \leq b \bullet a$  在  $G$  中成立. 简称其为规则(I). 证毕.

**定理 5**(可靠性定理).  $RPL_{ECW}$  相对于  $G = \langle \mathcal{A}, \bullet, \leq \rangle$  是可靠的.

**证明** 由引理 2 易证得.

**定理 6**(完全性定理).  $RPL_{ECW}$  相对于  $G = \langle \mathcal{A}, \bullet, \leq \rangle$  是完全的.

**证明**

(1) 令  $\mathcal{L} = (W, \bullet)$  为 Integral-Contractive-偏序半群. 在幂集  $\mathcal{W}(P(W))$  上定义二元运算符  $\otimes$ , 如下:

$$U \otimes V = \{a \bullet b \mid a \in U \text{ and } b \in V\}.$$

易证得,  $\langle P(W), \otimes, \subseteq \rangle$  是偏序半群.

(2) 给定序列  $\Gamma_0 \Rightarrow A_0$ , 令  $S$  表示该序列中所有公式的子公式集合. 显然,  $S$  是有穷集合.  $S$  上的语言为  $S^*$  的子集. 定义  $G = \langle \mathcal{A}, \bullet, \leq \rangle$  上的典范赋值  $f$  如下所示:

$$f(p) = \{\Gamma \in S^* : \Gamma \Rightarrow p\}.$$

简称为规则(II).

(3) 证明规则(II)  $f(A) = \{\Gamma \in S^* : \Gamma \Rightarrow A\}$  如下.

施归纳于  $A$  的复杂度:

情况 1 若  $A = p$ , 则结论成立;

情况 2 若  $A = B \bullet C$ , 则

左 $\Rightarrow$ 右. 设  $\Gamma \in f(B \bullet C)$ . 据规则(I),  $\Gamma \in f(B)$  且  $\Gamma \in f(C)$ , 据归纳假设  $\Gamma \Rightarrow B$  且  $\Gamma \Rightarrow C$ . 据规则( $\bullet R$ ),  $\Gamma \Rightarrow B \bullet C$ ;

右 $\Rightarrow$ 左. 设  $\Gamma \Rightarrow B \bullet C$ . 据规则( $\bullet L$ )和弱化规则, 易得  $\Gamma \Rightarrow B$  且  $\Gamma \Rightarrow C$ , 据归纳假设  $\Gamma \in f(B)$  且  $\Gamma \in f(C)$ , 据规则(I), 有  $\Gamma \in f(B)$  且  $\Gamma \in f(C)$ .

依据情况 1、2, 则规则(II)得证.

(4) 证明完全性. 设  $\Gamma_0 \not\vdash A_0$ , 据规则(I),  $\Gamma_0 \notin f(A_0)$ . 又因为  $\Gamma_0 \Rightarrow \Gamma_0$ , 据规则(II),  $\Gamma_0 \in f(\Gamma_0)$ .  $\therefore \mu(\Gamma_0) \notin f(A_0)$ ;  $\therefore \Gamma_0 \not\vdash A_0$ . 证毕.

现在我们证明  $RPL_{ECW}$  句法逻辑系统可判定性证明的复杂度是 P-time( $O(n^2)$ ).

**定理 7**(可判定性时间复杂度的证明).  $\vdash_{ECW} \Gamma \Rightarrow A$  当且仅当每个原子公式  $p_i$  出现在该序列的右边时, 那么  $p_i$  必须出现在该序列的左边.

以下通过构造性证明, 验证该序列是否满足上述定理. 从右边开始.

Step 1 若对  $\forall p_i \in A$ , 存在  $p_i \notin \Gamma$ , 则  $\Gamma \Rightarrow A$  是不可证的, 故结束;

Step 2 令  $A = A - p_i$  and  $\Gamma = \Gamma - p_i$ , 返回 Step 1 直到  $A$  为空, 或者  $\Gamma \Rightarrow A$  是可证明的;

Step 3 若  $A = \emptyset$ , 则  $\Gamma \Rightarrow A$  是可证的, 结束.

假设存在  $n$  公式,  $m = \text{Max}(\text{Length}(G), \text{Length}(A))$ , 则该方法的时间复杂度至多是  $O(mn)$ , 或者说  $O(n^2)$ , 即 P-time. 使用同样的方法, 不难证明, 所有子结构逻辑及其代数可判定性的证明时间复杂度都是 P-time.

## 5 结论

基于角色协同(RBC)是一种软件工程与协同计算方法, 其以角色为基础进行研究, 能有效促进信息系统的抽象与分类, 从而有利于系统单元协作与协同计算. 角色引擎能有效对 RBC 形成支撑, 为支撑角色引擎, 并对群组、角色和代理进行形式化建模, 本文根据它们之间的必要关联性和层次结构, 提出了系列角色扮演逻辑(RPL)及相关代数模型, 对 RBC 进行了高度化抽象.

角色扮演逻辑为角色协作提供三个层次的应用指导, 分别是全局层面: 组织代理, 并形成群组; 衔接层面: 群组角色指派; 操作层面: 管理协同系统设计中应该出现的协作特性、关系和结构. 可通过遵循具体的设计要求, 对 RPL 进行扩展, 以满足相应的协同计算.

本文主要研究基于角色协同的角色扮演逻辑的模型论(理论)部分内容. 在下一步工作中, 我们将结合复杂工程应用, 依据全局层与操作层群组模型论做建模, 依据衔接层的群组角色指派完成分工指派, 并对复杂、多约束环境下协作协同做知识表达与指派计算, 影响并指导基于 RBC 的协同计算与协同工作.

**致谢** 谨对朱海滨教授对本文的帮助与支持表示衷心感谢.

## 参考文献

- [1] ZHU Hai-bin. Role-based collaboration and E-CARGO: revisiting the developments of the last decade role-based collaboration (RBC) is an emerging computational methodology that uses roles as the prim[J]. IEEE Systems Man & Cybernetics Magazine, 2015, 1(3): 27-36.
- [2] ZHU Hai-bin. Fundamental issues in the design of role engine[A]. Proceedings of the 6th International Symposium on Collaborative Technologies and Systems [C]. Irvine, CA, USA: ACM, 2008. 399-407.
- [3] GALITSKY B. Exhaustive simulation of consecutive mental states of human agents[J]. Knowledge-Based Systems, 2013, 43(2): 1-20.
- [4] GONCALVES E J T, CORTÉS M I, CAMPOS G A L, et

- al. MAS-ML 2.0: supporting the modelling of multi-agent systems with different agent architectures [J]. *Journal of Systems & Software*, 2015, 108: 77 – 109.
- [5] HOSNY A A, SAMIR I S, MOHAMMED H A. Organization of multi-agent systems: an overview [J]. *International Journal of Intelligent Information Systems*, 2015, 4(3): 46 – 57.
- [6] RAO A S, GEORGEFF M P. BDI-agents: from theory to practice [A]. *Proceedings of the First International Conference on Multi-Agent Systems (ICMAS'95)* [C]. San Francisco, CA: AAAI Press, 1995. 312 – 319.
- [7] FORTINO G, RANGO F, RUSSO W, et al. Translation of statechart agents into a BDI framework for MAS engineering [J]. *Engineering Applications of Artificial Intelligence*, 2015, 41(C): 287 – 297.
- [8] VISSER S, THANGARAJAH J, HARLAND J, et al. Preference-based reasoning in BDI agent systems [J]. *Autonomous Agents and Multi-agent Systems*, 2015, 30(2): 291 – 330.
- [9] MEYER J J C, HOEK W V D. *Epistemic Logic for AI and Computer Science* [M]. Cambridge: Cambridge University Press, 1995.
- [10] HAREL D, KOZEN D, TLUY J. *Foundations of Computing Series: Dynamic Logic* [M]. Cambridge: MIT Press, 2000.
- [11] 刘冬宁, 汤庸, 滕少华, 林哲. 基于时态数据库的极小子结构逻辑系统 [J]. *计算机学报*, 2013, 36(8): 1592 – 1601.  
LIU Dong-ning, TANG Yong, TENG Shao-hua, LIN Zhe. A Minimal Substructural Logic in Temporal Database [J]. *Journal of Computer*, 2013, 36(8): 1592 – 1601. (in Chinese)
- [12] BUSZKOWSKI W. Type logics and pregroups [J]. *Studia Logica*, 2007, 87(2): 145 – 169.
- [13] BUSZKOWSKI W, LIN Zhe, MOROZ K. Pregroup grammars with letter promotions: complexity and context-freeness [J]. *Journal of Computer and System Science*, 2012, 78(6): 1899 – 1909.
- [14] LIU Dong-ning, ZHU Hai-bin, TENG Shao-hua, TANG Yong. Minimal role playing logic in role-based collaboration [A]. *IEEE International Conference on Systems, Man and Cybernetics (SMC)* [C]. San Diego, CA, USA: IEEE, 2014. 1420 – 1425.
- [15] ZHU Hai-bin, HOU Ming, WANG Chun, ZHOU Meng-chu. An efficient outpatient scheduling approach [J]. *IEEE Transactions on Automation Science and Engineering*, 2012, 9(4): 701 – 709.
- [16] ZHU Hai-bin, ZHOU Meng-chu. Role-based collaboration and its kernel mechanisms [J]. *IEEE Transactions on Systems Man & Cybernetics Part C*, 2006, 36(4): 578 – 589.
- [17] LAMBEK J. The mathematics of sentence structure [J]. *American Mathematical Monthly*, 1958, 3(65): 154 – 170.
- [18] ZHU Hai-bin, ZHOU Meng-chu. Roles in information systems: a survey [J]. *IEEE Transactions on Systems Man & Cybernetics Part C: Applications and Reviews*, 2008, 38(3): 377 – 396.
- [19] ZHU Hai-bin. Collective group role assignment and its complexity [A]. *IEEE International Conference on System, Man and Cybernetics* [C]. Anchorage, AK: IEEE, 2011. 351 – 356.
- [20] ZHU Hai-bin, LIU Dong-ning, Zhang Si-qin, et al. Solving the Many to Many Assignment Problem by Improving the Kuhn-Munkres Algorithm with Backtracing [J]. *Theoretical Computer Science*, 2016, (618): 30 – 41.

### 作者简介



**滕少华** 男, 1962 年生于江西南昌. 现为广东工业大学计算机学院教授、硕士生导师. 主要研究方向为协同计算、数据挖掘、网络安全、大数据.  
E-mail: shteng@gdut.edu.cn



**张如琪** 女, 1991 年生于广东茂名. 现为广东工业大学计算机学院硕士研究生. 主要研究方向为协同计算、数据挖掘.  
E-mail: zhangrq002@outlook.com



**刘冬宁 (通信作者)** 男, 1979 年生于广东广州. 现为广东工业大学计算机学院副教授、硕士生导师. 主要研究方向为协同计算、人工智能逻辑.  
E-mail: liudn@gdut.edu.cn



**刘林源** 男, 1981 年生于湖南长沙, 博士, 现为南京审计大学电子商务系讲师. 主要研究方向为协同计算, 系统可靠性与安全.  
E-mail: liulinyuan@nau.edu.cn

**朱咸军** 男, 1977 年生于江苏盐城, 现为南京大学工程管理学院博士研究生. 主要研究方向为协同计算.  
E-mail: mymailzxj@126.com