软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2012,23(7):1796−1804 [doi: 10.3724/SP.J.1001.2012.04135]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

联盟规范系统及其规范能力极限*

王崇骏[1,2+], 吴 骏[3], 张 雷[1,2], 谢俊元[1,2]

[1](南京大学 计算机科学与技术系,江苏 南京 210093)
[2](计算机软件新技术国家重点实验室(南京大学),江苏 南京 210093)
[3](河海大学 计算机与信息学院,江苏 南京 210098)

On the Limitation of the Power of Coalitional Normative Systems

WANG Chong-Jun[1,2+], WU Jun[3], ZHANG Lei[1,2], XIE Jun-Yuan[1,2]

[1](Department of Computer Science and Technology, Nanjing University, Nanjing 210093, China)
[2](National Key Laboratory for Novel Software Technology (Nanjing University), Nanjing 210093, China)
[3](College of Computer and Information Engineering, Hohai University, Nanjing 210098, China)
+ Corresponding author: E-mail: chjwang@nju.edu.cn

Wang CJ, Wu J, Zhang L, Xie JY. On the limitation of the power of coalitional normative systems. *Journal of Software*, 2012,23(7):1796−1804 (in Chinese). http://www.jos.org.cn/1000-9825/4135.htm

Abstract: Coalitional Normative System (CNS) is an extension of a Normative System (NS) by enabling selective constraining a coalition's joint behavior. The study extends the semantics of ATL and proposes Coordinated ATL (Co-ATL) to support the formalizing of CNS. The paper classifies all the CNSs that control the same coalition into the same class and characterizes the power limitation of each such class by identifying two fragments of Co-ATL language corresponding to two types of system properties that are unchangeable. The relation between NS and CNS is discussed, and it turns out that by the result better characterize the power limitation of NS. Moreover, the study extends further the CNS by including a finite state machine encoding the execution history, and results show that the power limitation characterization for CNS is invariant under this extension.
Key words: normative system; ATL; coalition; concurrent game structure

摘 要: 联盟规范系统(coalitional normative system,简称 CNS)通过选择性地限制联盟的联合行动来对规范系统(normative system,简称 NS)进行扩展.扩展了 ATL 的语义,提出了 Coordinate-ATL(Co-ATL),用于对 CNS 进行形式化.为了刻画其规范能力的极限,确定了 Co-ATL 的两个语言片段,分别对应于两类不可改变的系统属性.对 NS 和 CNS 之间的关系进行了讨论,表明所得到的结果可以更好地界定 NS 的能力极限.此外,引入了对执行历史进行编码的有限状态机,进一步对 CNS 进行了扩展,提出了 CNS-M.可以证明,关于 CNS 能力极限的界定在该扩展下保持稳定.
关键词: 规范系统;交互时态逻辑;联盟;并发博弈结构
中图法分类号: TP18 **文献标识码**: A

为了以离线方式协同多 agent 系统,Shoham 和 Tennenholtz 在 1992 年首先提出了规范系统(normative system

* 基金项目: 国家自然科学基金(60503021, 60721002, 60875038, 61105069)
收稿时间: 2010-08-06; 定稿时间: 2011-10-17

或 social law,简称 NS)的概念[1,2].此后,有很多基于这一概念的扩展,例如,文献[3–7]通过引入形式化的模态逻辑和时间逻辑,尤其是交互时态逻辑(alternating-time temporal logic,简称 ATL)[8,9]及其变种来扩展规范系统.这些逻辑都可以用来描述与验证社会选择等机制.因此,这方面的工作也可以被视为自动化机制设计的逻辑研究的一部分[10,11].

虽然 NS 存在很多实现方法,但在技术细节上都存在很多相似之处:NS 是一组约束集,用于对多 agent 系统中的 agent 行为进行约束,以期实现某些需要的全局目标[12],对应于一个真值为假的逻辑公式.在这一约束下,真值变为真(称"这个公式被确立")或一个原本真值为真的逻辑公式在这一约束下真值变为假(称"这个公式被避免").其内在机理是:规范系统会导致语义模型的更新,从而使一些公式(相对于语义模型)的满足性得到改变.显然,规范系统不是万能的,确立或避免某些类别的公式不可避免地超越了所有规范系统的能力极限.如文献[5]确定了 ATL 的两个语言片段:$\mathcal{L}^e$ 和$\mathcal{L}^u$,并指出,不可能通过构造一个规范系统来确立$\mathcal{L}^e$ 的公式或避免$\mathcal{L}^u$ 中的公式.本文认为,片段$\mathcal{L}^e$ 和$\mathcal{L}^u$ 共同界定了规范系统的能力极限.

联盟规范系统(coalitional normative system,简称 CNS)是一种对 agent 联盟联合行为的约束,本文旨在提出一个基于 ATL 的联盟规范系统框架,并界定其规范能力的极限.本文的贡献表现在如下 4 个方面:

1) 扩展了 ATL 的语义,提出了 Coordinate-ATL(Co-ATL),并据此对联盟规范系统进行形式化;
2) 对于 agent 集合 C,通过确定 Co-ATL 的两个片段 $\mathcal{L}_c^+$ 和 $\mathcal{L}_c^-$,完备地界定了对于 C 的联盟规范系统的能力极限;
3) 证明了 NS 实际上是所有 agent 联盟(大联盟)的 CNS,本文的结果可以更好地表征 NS 的能力限度;
4) 将 CNS 扩展到 CNS-M,使它能够根据不同的"执行历史"在同一个状态施加不同的行为约束,并且 Co-ATL 的语言片段 $\mathcal{L}_c^+$ 和 $\mathcal{L}_c^-$ 同样能够界定对于 C 的 CNS-M 的能力极限.

本文第 1 节介绍 ATL 和 NS 的基本知识.第 2 节对 CNS 进行形式化并介绍 Co-ATL 的语法和语义.第 3 节证明关于 Co-ATL 语言片段 $\mathcal{L}_c^+$ 和 $\mathcal{L}_c^-$ 的一些结论,并讨论 CNS 的能力限度.第 4 节给出一种 CNS 的扩展方案.第 5 节对本文的工作进行总结.

1 背景:ATL 和规范系统

1.1 交互时态逻辑

交互时态逻辑[8,9]是对 CTL 的扩展,把路径选择量词∀和∃替换为路径选择量词《》,它可以表达社会选择论中的α-能力[4].ATL 的公式可由并发博弈结构[9]解释.

定义 1(并发博弈结构). 并发博弈结构是一个元组 $S=\langle k,Q,\Pi,\pi,d,\delta\rangle$,其中,

- k 为系统中 agent 的数目,本文用自然数 1,2,3,…来指代每个 agent;
- Q 为状态集合;
- Π为命题集合;
- π为标签函数,定义如下:对于每一个状态 $q\in Q$,存在一个集合$\pi(q)\subseteq\Pi$,使得$\pi(q)$中的命题在状态 q 下都为真;
- $d_a(q)$表示 agent a 在状态 q 下的非空行动集,其中,$a\in\{1,2,\ldots,k\}$,$q\in Q$.$\langle j_1,j_2,\ldots,j_k\rangle$则表示在状态 q 下的 agent 集合$\{1,2,\ldots,k\}$的一个联合行动,其中,$j_a\in d_a(q)$.同时,定义动作函数 $D(q)=d_1(q)\times d_2(q)\times\ldots\times d_k(q)$;
- $\delta(q,j_1,j_2,\ldots,j_k)$为状态转移函数,对于每一个状态 $q\in Q$ 和联合行动$\langle j_1,j_2,\ldots,j_k\rangle\in D(q)$,状态转移函数指定了系统在状态 q 下联合行动$\langle j_1,j_2,\ldots,j_k\rangle$导致的下一个状态,即$\delta(q,j_1,j_2,\ldots,j_k)\in Q$.

需要注意的是,每一个 agent 子集 $A\subseteq\{1,2,\ldots,k\}$均可看作是一个联盟($\{1,2,\ldots,k\}\backslash A$ 代表该联盟的环境).大联盟$\{1,2,\ldots,k\}$记为 Ag.本文中使用 $\vec{m}$ 指代所有 agent 的联合行动$\langle j_1,j_2,\ldots,j_k\rangle$,用 $\vec{m}_A$ 指代一个 agent 子集 $A\subset Ag$ 的联合行动(称为一个 A-行动),并且使用 $D_a(q)$指代状态 q 下所有可能的 A-行动.同时,引入表达式 $\vec{m}_A\,|\,A'$ 表示在集合 A 采取联合行动 $\vec{m}_A$ 的情况下,集合 $A\cap A'$的联合行动.

关于并发博弈结构的一些重要概念定义如下:

1) 后继:对于两个状态 q 和 q',q' 被称为 q 的后继,当且仅当存在一个联合行动 $\vec{m}\in D(q)$,使得 $q'=\delta(q,\vec{m})$;
2) 计算:一个无限的状态序列 $\lambda=q_0,q_1,\ldots$ 被称作 S 的一个计算,当且仅当对于每一个 $i\geqslant 0$,状态 q_{i+1} 都是 q_i 的后继;
3) q-计算:从状态 q 开始的一个计算称为一个 q-计算.对于计算的 λ 和位置 $i(i\geqslant 0)$,本文用 $\lambda[i]$,$\lambda[0,i]$ 和 $\lambda[i,\infty]$ 分别表示 λ 的第 i 个状态、λ 的前缀 $q_0,q_1,\ldots,q_i$ 和 λ 的后缀 $q_{i+1},q_{i+2},\ldots$;
4) 策略:Agent a 的策略 $f_a\in\Sigma$ 是从有限非空的状态序列 $\lambda\in Q^+$ 到行动的一个函数映射:如果 λ 的最后一个状态是 q,则有 $f_a(\lambda)\in d_a(q)$.对于一个 agent 集合 $A\subseteq Ag$,其联合策略被称作一个 A-策略;
5) 结果:A-策略 F_A 在状态 q 上的结果为一个计算的集合 $out(q,F_A)$.一个计算 $\lambda=q_0,q_1,\ldots\in out(q,F_A)$,当且仅当 $q_0=q$,且对于任意 $q_i=\lambda[i]$,存在联合行动 $\langle j_1,j_2,\ldots,j_k\rangle\in D(q_i)$,使得:(1) 对于所有的 agent $a\in A,f_a\in F_A$,$j_a=f_a(\lambda[0,i])$;(2) $\delta(q_i,j_1,j_2,\ldots,j_k)=q_{i+1}$.

定义 2(ATL 的语言). ATL 的语言 $\mathcal{L}$ 由下列语法产生:

$$\varphi ::= p\mid\neg\varphi\mid\varphi_1\vee\varphi_2\mid\langle\langle A\rangle\rangle\bigcirc\varphi\mid\langle\langle A\rangle\rangle\Box\varphi\mid\langle\langle A\rangle\rangle\varphi_1\mathcal{U}\varphi_2,$$

其中,$p\in\Pi$ 是一个命题,而 $A\subseteq Ag$ 是一个 agent 的集合.本文将 $\langle\langle A\rangle\rangle\top\,\mathcal{U}\varphi$ 缩写为 $\langle\langle A\rangle\rangle\Diamond\varphi$.

定义 3(ATL 的语义). 用 $S,q\vDash\varphi$ 来表示公式 φ 在并发博弈结构 S 中的状态 q 下成立.当上下文没有歧义时,本文将其缩写为 $q\vDash\varphi$.其中,关系 $\vDash$ 定义如下:

- 对于所有 $p\in\Pi,q\vDash p$ 当且仅当 $p\in\pi(q)$;
- $q\vDash\neg\varphi$ 当且仅当 $q\nvDash\varphi$;
- $q\vDash\varphi_1\vee\varphi_2$ 当且仅当 $q\vDash\varphi_1$ 或 $q\vDash\varphi_2$;
- $q\vDash\langle\langle A\rangle\rangle\bigcirc\varphi$ 当且仅当存在一个 A-策略,使得对于每个计算 $\lambda\in out(q,F_A)$,都有 $\lambda[1]\vDash\varphi$;
- $q\vDash\langle\langle A\rangle\rangle\Box\varphi$ 当且仅当存在一个 A-策略,使得对于每个计算 $\lambda\in out(q,F_A)$ 和位置 $i(i\geqslant 0)$,都有 $\lambda[i]\vDash\varphi$;
- $q\vDash\langle\langle A\rangle\rangle\varphi_1\mathcal{U}\varphi_2$ 当且仅当存在一个 A-策略,使得对于每个计算 $\lambda\in out(q,F_A)$ 存在一个位置 $i(i\geqslant 0)$,使得 $\lambda[i]\vDash\varphi_2$;并且对于所有的位置 $j(0\leqslant j<i)$,都有 $\lambda[j]\vDash\varphi_1$.

1.2 规范系统

定义 4(规范系统). 给定一个并发博弈结构 $S=\langle k,Q,\Pi,\pi,d,\delta\rangle$,一个规范系统(NS)是一个函数 η,使得对于所有 agent $a\in Ag$ 和所有状态 $q\in Q$,都有 $\eta(a,q)\subseteq 2^{d_a(q)}\setminus\{\varnothing\}$.

$\eta(a,q)$ 是 agent a 在状态 q 下“被允许”的行动集合(通常,规范系统定义对于每个 agent 的每个状态指定相应的“禁止”行动的函数.本文将规范系统定义为指定允许行动的函数,这两个定义是等价的).通过执行一个 NS η,在并发博弈结构 S 中,对所有的 agent $a\in Ag$ 和所有状态 $q\in Q$,令 $d_a(q)=\eta(a,q)$.本文将得到的结构记作 $S\dagger\eta$.这样限制了可允许的行动范围,显然,$S\dagger\eta$ 仍是一个并发博弈结构.

文献[5]中给出了全局性 ATL 子语言和存在性 ATL 子语言,分别记为 $\mathcal{L}^e$ 和 $\mathcal{L}^u$,由语法 e 和语法 u 分别定义为

$$e ::= p\mid e\wedge e\mid e\vee e\mid\langle\langle Ag\rangle\rangle\bigcirc e\mid\langle\langle Ag\rangle\rangle\Diamond e\mid\langle\langle Ag\rangle\rangle e\mathcal{U}e,$$

$$u ::= p\mid u\wedge u\mid u\vee u\mid\langle\langle\rangle\rangle\bigcirc u\mid\langle\langle\rangle\rangle\Diamond u\mid\langle\langle\rangle\rangle u\mathcal{U}u,$$

其中,$p\in\Pi$.并且证明了以下结果:

对于并发博弈结构 S、行为约束 η、状态 q 和 $e\in\mathcal{L}^e,u\in\mathcal{L}^u$,以下式子成立:

1) $S\dagger\eta,q\vDash e\Rightarrow S,q\vDash e$;
2) $S,q\vDash u\Rightarrow S\dagger\eta,q\vDash u$.

第 1 个结果表明,无法通过建立一个规范系统来确立 $\mathcal{L}^e$ 中的公式;第 2 个结果表明,无法通过建立一个规范系统来避免 $\mathcal{L}^u$ 中的公式.

2 联盟规范系统

联盟规范系统,即 CNS,通过“选择性”地限制联盟的联合行为,对 NS 进行了扩展.

定义 5(联盟规范系统). 对于一个并发博弈结构 $S=\langle k,Q,\Pi,\pi,d,\delta\rangle$,定义联盟规范系统为一个二元组 $\Gamma=\{C,\vartheta\}$,其中,

- C 表示一个联盟,其中,$C\subseteq\{1,2,\ldots,k\}$;
- ϑ 为协同函数,对于每一个 $q\in Q$, $\vartheta(q)\subseteq D_C(q)$ 表示联盟 C 中 agent 可以联合选择的行动集合.

值得注意的是,本文有时将一个联盟规范系统 $\Gamma=\langle C,\vartheta\rangle$ 称为一个 C-规范.当考虑 CNS 时,某些联合行动的选择和对应的计算将被排除.本文采用前缀“Γ-相容”来指代“被 Γ 允许”.

1) 一个联合行动 $\vec{m}\in D(q)$ 被称作 Γ-相容行动,当且仅当存在 $\vec{m}_C\in\vartheta(q)$,使得 $\vec{m}\,|\,C=\vec{m}_C$;
2) 一个状态 q' 被称为状态 q 的 Γ-相容后继,如果存在一个 Γ-相容动作向量 $\vec{m}\in D(q)$,使得 $q'=\delta(q,\vec{m})$;
3) S 的一个 Γ-相容计算是指一个无限序列 $\lambda=q_0,q_1,\ldots$,对于所有位置 $i(i\geqslant 0)$,状态 q_{i+1} 是状态 q_i 的一个 Γ-相容后继;
4) 对于每个状态 $q\in Q$,联盟 A 的一个 A-行动 $\vec{m}_A$ 被称作 Γ-相容 A-行动,当且仅当存在 $\vec{m}_C\in\vartheta(q)$,使得 $\vec{m}_A\,|\,A\cap C=\vec{m}_C\,|\,A\cap C$;
5) 联盟 A 的策略集合 $F_A=\{f_a|a\in A\}$ 被称作 Γ-相容 A-策略,当且仅当对于所有非空有限状态序列 $\lambda\in Q^+$,F_A 给出的 $\vec{m}_A$ 是 Γ-相容 A-行动;
6) 最后,在状态 $q\in Q$ 下,一个 Γ-相容 A-策略的 Γ-相容结果是一个满足下列条件的集合 $out_\Gamma(q,F_A)$:当 $q_0=q$ 时,Γ-相容 $\lambda=q_0,q_1,\ldots$ 属于集合 $out_\Gamma(q,F_A)$,并且存在一个 Γ-相容联合行动 $\vec{m}\in D(q_i)$,使得对于所有 agent,$a\in A,j_a=f_a(\lambda[0,i])$,且 $\delta(q_i,\vec{m})=q_{i+1}$.

同样,可以将 CNS $\Gamma=\langle C,\vartheta\rangle$ 作用后的并发博弈结构 S 记为 $S\dagger\Gamma$.在大多数情况下,$S\dagger\Gamma$ 不是一个传统的并发博弈结构,联盟 C 中的 agent 将预先“讨论”应如何选择 C-行动.因此,“联盟协同”在结构中得到显式的体现.

因此,在存在联盟规范系统的情况下,需要改进对 ATL 公式的语义解释.本文称这种新逻辑为协同 ATL (coordinate-ATL,简称 Co-ATL),它直接继承了 ATL 的语法,但语义略有不同.

定义 6(Co-ATL 语义). 用 $S,\Gamma,q\vDash\varphi$ 表示在联盟规范系统的约束下,公式 φ 在并发博弈结构 S 中的状态 q 下成立.当 S 和 Γ 在上下文中没有歧义时,缩写为 $q\vDash\varphi$.关系 $\vDash$ 的定义如下:

1) 对于所有 $p\in\Pi$,$q\vDash p$ 当且仅当 $p\in\pi(q)$;
2) $q\vDash\neg\varphi$ 当且仅当 $q\nvDash\varphi$;
3) $q\vDash\varphi_1\vee\varphi_2$ 当且仅当 $q\vDash\varphi_1$ 或 $q\vDash\varphi_2$;
4) $q\vDash\langle\langle A\rangle\rangle\bigcirc\varphi$ 当且仅当存在一个 Γ-相容的 A-策略,使得对于每个 Γ-相容的计算 $\lambda\in out(q,F_A)$,都有 $\lambda[1]\vDash\varphi$;
5) $q\vDash\langle\langle A\rangle\rangle\Box\varphi$ 当且仅当存在一个 Γ-相容的 A-策略,使得对于每个 Γ-相容的计算 $\lambda\in out(q,F_A)$ 和位置 i $(i\geqslant 0)$,都有 $\lambda[i]\vDash\varphi$;
6) $q\vDash\langle\langle A\rangle\rangle\varphi_1\mathcal{U}\varphi_2$ 当且仅当存在一个 A-策略,使得对于每个 Γ-相容的计算 $\lambda\in out(q,F_A)$ 存在一个位置 i $(i\geqslant 0)$,使得 $\lambda[i]\vDash\varphi_2$;并且对于所有的位置 $j(0\leqslant j<i)$,都有 $\lambda[j]\vDash\varphi_1$.

通过采用空的 CNS: $\Gamma_\forall^C=\langle C,\vartheta_\forall\rangle$, 其中,对于每一个 $q\in Q$ 都有 $\vartheta(q)=D_C(q)$,可以确定 Co-ATL 和 ATL 的关系如下:

命题 1. 给定一个并发博弈结构 $S=\langle k,Q,\Pi,\pi,d,\delta\rangle$,对于所有 $q\in Q,C\subseteq Ag$ 和 $\varphi\in\mathcal{L}$,下式成立:

$$S,\Gamma_\forall^C,q\vDash\varphi\Leftrightarrow S,q\vDash_{ATL}\varphi.$$

与 ATL 相比,Co-ATL 可以在更一般的结构中表示 α-能力,如在联盟规范系统作用后的并发博弈结构中.为了区分这两种语义,本文使用“$\vDash_{\mathrm{ATL}}$”来表示 ATL 中的满足关系.

如图 1 所示,图 1(a)是一个并发联盟结构 $S=\langle k,Q,\Pi,\pi,d,\delta\rangle$,其中,$k=2,Q=\{q_0,q_1,q_2,q_3,q_4\},\Pi=\{P_1,P_2,P_3,P_4\},\pi(q_i)=P_i(i=1,\ldots,4),d_1(q_0)=d_2(q_0)=2,d_1(q_i)=d_2(q_i)=1(i=1,\ldots,4),\delta(q_0,1,1)=q_1,\delta(q_0,1,2)=q_2,\delta(q_0,2,1)=q_3,\delta(q_0,2,2)=q_4,\delta(q_i,1,1)=q_i(i=1,\ldots,4)$,则有:

1) 给定一个如下的 NS:

$$\eta(1,q_0)=\{1\},\eta(2,q_0)=\{2\},\eta(i,q_j)=\varnothing(i\in\{1,2\},j\in\{1,\ldots,4\}).$$

2) 其与下述 CNS 具有同样的效果:

$$\Gamma=\{C,\vartheta\},$$

其中,$C=\{1,2\}$,$\vartheta(q_0)=\{\langle 1,1\rangle,\langle 1,2\rangle,\langle 2,2\rangle\}$,$\vartheta(q_i)=\varnothing(i\in\{1,\ldots,4\})$.

3) 给定一个 CNS 描述如下:

$$\Gamma'=\langle C',\vartheta'\rangle,$$

其中,$C'=\langle 1,2\rangle$,$\vartheta'(q_0)=\{\langle 1,2\rangle\}$,$\vartheta'(q_i)=\varnothing(i\in\{1,\ldots,4\})$,

则 CNS 可以将结构转变为如图 1(b)所示,但是不存在任何 NS 可以达到这种转变.

4) 图 1(b)描述的结构不能用并发博弈结构来描述.

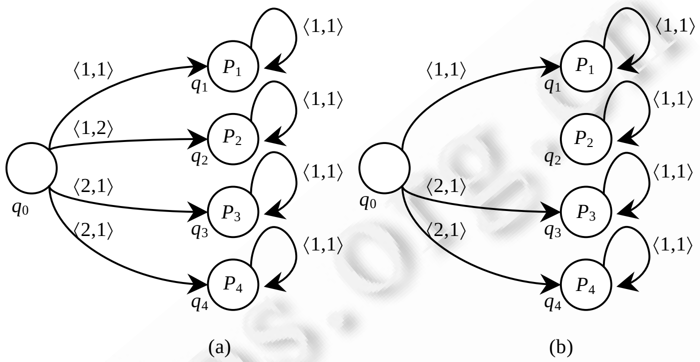

Fig.1 Example of CNS

图 1 CNS 示例

3 联盟规范系统的能力界定

对于任意公式$\varphi\in\mathcal{L}$,φ不能通过建立一个 C-规范而确立,当且仅当对所有并发博弈结构 S,不存在任何 C-规范Γ,使得存在一个状态 $q\in Q$,满足 $S,\Gamma_{\forall}^{C},q\nvDash\varphi$ 和 $S,\Gamma,q\vDash\varphi$.同样,φ不能通过建立一个 C-规范而避免,当且仅当对所有并发博弈结构 S,不存在任何 C-规范Γ,使得存在一个状态 $q\in Q$,满足 $S,\Gamma_{\forall}^{C},q\vDash\varphi$ 和 $S,\Gamma,q\nvDash\varphi$.

然后,C-规范的能力极限可以通过回答以下两个问题来刻画:

1) $\mathcal{L}$中哪一片段中的公式不能通过执行 C-规范得到满足;

2) $\mathcal{L}$中哪一片段中的公式不能通过执行 C-规范得到避免.

首先,本文确定了两种类型的子集,即 C 的超集(记为 C^+)与 $Ag\backslash C$ 的子集(记为 C^-)(例如,图 2 为一些可能的 C^+和 C^-),然后有以下结果:

引理 1. 给定一个任意的并发博弈结构 S、状态 q_0 和一个任意的 C-规范Γ,如果 C^+是一组满足 $C\subseteq C^+\subseteq Ag$ 的 agent,且 F_{C^+} 是一个Γ-相容的 C^+-策略,则存在一个 C^+-策略 F'_{C^+},使得

$$out(q_0,F'_{C^+})=out_{\Gamma}(q_0,F_{C^+}).$$

证明:始终可以定义 $F_{C^+}=F'_{C^+}$.而在所有的状态中,在 C^+中的 agent 选定一个 C^+-行动后,$Ag\backslash C^+$中 agent 的可用联合行动和Γ-相容联合行动是相同的,即所有的联合行动 $D_{Ag\backslash C^+}(q)$, 这是因为Γ没有在 $Ag\backslash C^+$设置任何约束.□

引理 2. 给定一个任意的并发博弈结构 S、状态 q_0 和一个任意的 C-规范Γ,如果 C^-是一组满足$\varnothing\subseteq C^-\subseteq Ag\backslash C$ 的 agent,且 F_{C^-} 是一个Γ-相容的 C^--策略,则存在一个 C^--策略 F'_{C^-},使得

$$out(q_0,F'_{C^-})=out_{\Gamma}(q_0,F_{C^-}).$$

证明:由于Γ其实没有对 C^-中 agent 的行为施加任何约束,所有的 C^--策略都是 C^-的Γ-相容的联合行动.因此,可以定义 $F_{C^+}=F'_{C^+}$.故在所有状态 q 中,对于 $Ag\backslash C^-$中的 agent,Γ-相容的联合行动的集合是联合行动的一个子集. □

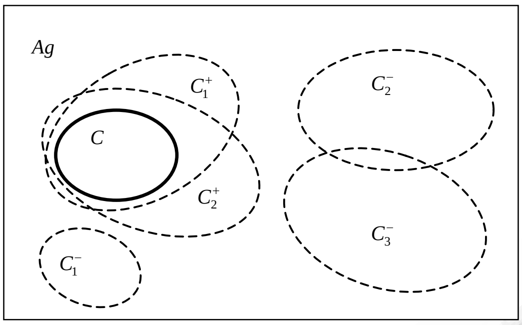

Fig.2 C^+ and C^-

图 2 C^+和 C^-

定义两个 Co-ATL 的语言片段 $\mathcal{L}_c^+$ 和 $\mathcal{L}_c^-$,分别由下列语法生成:

1) $\varphi ::= p \mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \vee \varphi_2 \mid \langle\!\langle C^+ \rangle\!\rangle \bigcirc \varphi \mid \langle\!\langle C^+ \rangle\!\rangle \Box \varphi \mid \langle\!\langle C^+ \rangle\!\rangle \varphi_1 \mathcal{U} \varphi_2 \mid \neg \psi$;

2) $\psi ::= p \mid \psi_1 \wedge \psi_2 \mid \psi_1 \vee \psi_2 \mid \langle\!\langle C^- \rangle\!\rangle \bigcirc \psi \mid \langle\!\langle C^- \rangle\!\rangle \Box \psi \mid \langle\!\langle C^- \rangle\!\rangle \psi_1 \mathcal{U} \psi_2 \mid \neg \varphi$.

其中,$p \in \Pi, C \subseteq C^+ \subseteq Ag, \varnothing \subseteq C^- \subseteq Ag \backslash C$.

此外,对 $\mathcal{L}_c^+$ 和 $\mathcal{L}_c^-$ 我们有以下结果:

定理 1. 给定任意一个并发博弈结构 S、任意一个 C-规范Γ和 S 中任意状态 q:

- 对所有的公式 $\varphi \in \mathcal{L}_c^+$,有

$$S,\Gamma,q \vDash \varphi \Rightarrow S,\Gamma_{\forall}^{C},q \vDash \varphi.$$

- 对所有的公式 $\varphi \in \mathcal{L}_C^-$,有

$$S,\Gamma_{\forall}^{C},q \vDash \psi \Rightarrow S,\Gamma,q \vDash \psi.$$

证明:通过对φ和ψ结构进行归纳证明.对于命题的情况,结论显然成立.对于其他情况下,假设一个任意$\varnothing \in \mathcal{L}_C^+$ 和一个任意 $\psi \in \mathcal{L}_C^-$ 结论成立,则对于$\varphi_1 \wedge \varphi_2, \psi_1 \wedge \psi_2, \varphi_1 \vee \varphi_2, \psi_1 \vee \psi_2, \neg\varphi, \neg\psi$的情况,显然也成立.

然后,对于所有的 $C \subseteq C^+ \subseteq Ag$ 和$\varnothing \subseteq C^- \subseteq Ag \backslash C$: $S,\Gamma,q \vDash \langle\!\langle C^+ \rangle\!\rangle \bigcirc \varphi \Rightarrow$ (通过 Co-ATL 的语义)存在一个Γ-相容 C^+-策略 F_{C^+}, 使得对于所有的Γ-相容的 q-计算 $\lambda \in out_{\Gamma}(q,F_{C^+})$, 都有 $S,\Gamma,\lambda[1] \vDash \varphi \Rightarrow$(通过归纳假设)对于所有的 $\lambda \in out_{\Gamma}(q,F_{C^+})$, 都有 $S,\Gamma,\lambda[1] \vDash \varphi \Rightarrow$(通过引理 2 和命题 1)存在一个 F'_{C^+}, 使得对于所有的 $\lambda' \in out(q,F'_{C^+})$, 都有

$$S,\Gamma_{\forall}^{C},\lambda'[1] \vDash_{\text{ATL}} \varphi \Rightarrow \text{(通过 ATL 语义和命题 1)} \; S,\Gamma_{\forall}^{C},\lambda[1] \vDash \langle\!\langle C^+ \rangle\!\rangle \bigcirc \varphi .$$

$S,\Gamma,q \vDash \langle\!\langle C^- \rangle\!\rangle \bigcirc \psi \Rightarrow$ (通过 Co-ATL 的语义和命题 1)存在一个 C^--策略 F_{C^-},使得对于所有的 q-计算 $\lambda \in out_{\Gamma}(q,F_{C^-})$, 都有 $S,\Gamma,\lambda[1] \vDash_{\text{ATL}} \varphi \Rightarrow$(通过引理 3)对于所有的 $\lambda' \in out(q,F'_{C^-})$,存在一个$\Gamma$-相容的 C^--策略 F'_{C^-}, $\lambda \in out_{\Gamma}(q,F_{C^+})$, 都有 $S,\Gamma,\lambda'[1] \vDash_{\text{ATL}} \psi \Rightarrow$(通过命题 1 和归纳假设)对于所有的 $\lambda' \in out(q,F'_{C^-})$,都有

$$S,\Gamma_{\forall}^{C},\lambda'[1] \vDash_{\text{ATL}} \varphi \Rightarrow \text{(通过 ATL 语义)} \; S,\Gamma,q \vDash \langle\!\langle C^- \rangle\!\rangle \bigcirc \psi.$$

类似地,可以证明结论对下列情况同样成立:$\langle\!\langle C^+ \rangle\!\rangle \Box \varphi, \langle\!\langle C^+ \rangle\!\rangle \varphi_1 \mathcal{U} \varphi_2, \langle\!\langle C^- \rangle\!\rangle \Box \psi$ 和$\langle\!\langle C^- \rangle\!\rangle \psi_1 \mathcal{U} \psi_2$. □

综上所述,可以得出结论,属于片段 $\mathcal{L}_c^+$ ($\mathcal{L}_c^-$)是一个公式不能通过建立一个 C-规范而确立(避免)的充分条件.接下来的另一个问题就是,属于片段 $\mathcal{L}_c^+$ ($\mathcal{L}_c^-$)是否也是一个公式不能通过建立一个 C-规范而满足(避免)的必要条件呢?本文对此给出了肯定的回答.

定理 2. 1) 如果 $\varphi \notin \mathcal{L}_c^+$,则φ可以通过建立一个 C-规范而确立;

2) 如果 $\varphi \notin \mathcal{L}_c^-$,则φ可以通过建立一个 C-规范而避免.

证明:本文只证明这个定理的第 1 部分,对第 2 部分的证明是类似的.

为了证明所有 $\mathcal{L}\setminus\mathcal{L}_c^+$ 中的的公式可以通过建立一个 C-规范而满足,本文的方法是通过为每个公式建立必要的并发博弈结构.设 $A\subseteq Ag$ 和 $A\cap C\neq C$(即 A 不是 C^+).对所有公式形如《A》γ,其中,γ形为 $\bigcirc\varphi,\square\varphi,\varphi_1\mathcal{U}\varphi_2$ 或 $\varphi,\varphi_1,\varphi_2$ 是任意的 Co-ATL 公式,构造一个包含状态 q 的并发博弈结构 S,满足 $S,q\nvDash_{\text{ATL}}$《A》γ 且 $S,q\vDash_{\text{ATL}}$《$A\cup C$》γ.因此,在这样的 S 中,$A\cup C$ 中的 agent 有一个联合策略 $F_{A\cup C}$,使得所有在 $out(q,F_{A\cup C})$中的计算满足φ.根据文献[9,13]可知,$F_{A\cup C}$ 可以是一种从状态映射到 $A\cup C$-行动的"无记忆"策略.通过构造 C-规范Γ,$A\cup C$ 中 agent 的联合行动限制为仅与 $F_{A\cup C}$ 相容.然后,由Γ有 $S,\Gamma,q\vDash$《A》γ.也就是说,《A》γ 可以通过建立一个 C-规范而满足.要证明"如果 $\varphi\notin\mathcal{L}_C^-$,$\neg\varphi$可以通过建立一个 C-规范而满足",可以转化成证明"如果 $\varphi\notin\mathcal{L}_C^-$,$\varphi$可以通过建立一个 C-规范而避免".

此外,可以直接证明,如果φ可以通过建立一个 C-规范而满足,那么下列公式同样可以通过建立一个 C-规范而满足:

$$\varphi\wedge\varphi',\varphi\vee\varphi',《A》\bigcirc\varphi,《A》\square\varphi,《A》\varphi'\mathcal{U}\varphi,《A》\varphi\mathcal{U}\varphi',\neg《A》\bigcirc\varphi,\neg《A》\square\varphi,\neg《A》\varphi'\mathcal{U}\varphi,\neg《A》\varphi\mathcal{U}\varphi',$$

其中,$A\subseteq Ag$,φ'是一个任意的 Co-ATL 公式.

据此可以得到结论: $\mathcal{L}_c^+$ 恰为不能通过建立 C-规范而满足的公式集合,且 $\mathcal{L}_c^-$ 恰为不能通过建立 C-规范而避免的公式集合.换句话说,$\mathcal{L}_c^+$ 及 $\mathcal{L}_c^-$ 有效地(定理 1)和完备地(定理 2)刻画了 C-规范的能力限制. □

下面的推论是定理 2 的必然结果.

推论 1. 1) $\mathcal{L}_\varnothing^+=\mathcal{L}_\varnothing^-$ 为如下语法产生的 Co-ATL 公式:

$$\varphi::=p\,|\,\varphi_1\wedge\varphi_2\,|\,\varphi_1\vee\varphi_2\,|\,《A》\bigcirc\varphi\,|\,《A》\square\varphi\,|\,《A》\varphi_1\mathcal{U}\varphi_2\,|\,\neg\varphi,$$

其中,$p\in\Pi$且$\varnothing\subseteq A\subseteq Ag$.

2) $\mathcal{L}_{Ag}^+$ 和 $\mathcal{L}_{Ag}^-$ 则分别是由以下语法产生的 Co-ATL 公式:

$$\varphi::=p\,|\,\varphi_1\wedge\varphi_2\,|\,\varphi_1\vee\varphi_2\,|\,《Ag》\square\varphi\,|\,《Ag》\square\varphi\,|\,《Ag》\varphi_1\mathcal{U}\varphi_2\,|\,\neg\psi,$$

$$\psi::=p\,|\,\psi_1\wedge\psi_2\,|\,\psi_1\vee\psi_2\,|\,《》\bigcirc\psi\,|\,《》\square\psi\,|\,《》\psi_1\mathcal{U}\psi_2\,|\,\neg\varphi,$$

其中,$p\in\Pi$.

综上所述,可以得到如下结论:

1) 关于 NS 与 CNS 之间的关系:一个任意的 NS η可以看作一个 CNS $\Gamma_\eta=\langle Ag,\vartheta_\eta\rangle$,其中,$\forall q\in Q$:$\vartheta_\eta(q)=\eta(1,q)\times\ldots\times(k,q)$.显然,CNS 并不一定是一个规范系统.所以,NS 是 Ag-规范类的一个真子集.由于 $\mathcal{L}_{Ag}^+$ 和 $\mathcal{L}_{Ag}^-$ 有效且完备地刻画了 Ag-规范的能力极限,因此得出结论: $\mathcal{L}_{Ag}^+$ 和 $\mathcal{L}_{Ag}^-$ 有效地刻画了规范系统的能力极限.虽然 $\mathcal{L}_{Ag}^+$ 和 $\mathcal{L}_{Ag}^-$ 不能完备地刻画 NS,但是 $\mathcal{L}_{Ag}^+$ 与 $\mathcal{L}_{Ag}^-$ 相比$\mathcal{L}^e$,$\mathcal{L}^u$ 可以更精确地刻画 NS 的能力极限;
2) 可以比较每对 CNS 类之间的能力.本文提及的类 C_1-规范比类 C_2-规范能力更强,当且仅当 $\mathcal{L}_{C_1}^+\subseteq\mathcal{L}_{C_2}^+$ 且 $\mathcal{L}_{C_1}^-\subseteq\mathcal{L}_{C_2}^-$.然后可以得出,任意两个联盟 C_1 和 C_2,如果 $C_1\subseteq C_2$,则有类 C_1-规范比类 C_2-规范能力更强.因此,类$\varnothing$-规范是最弱的 CNS 类.

4 带记忆的联盟规范系统

本节考虑对联盟规范系统进行扩展,使它可以在一个有限状态机中"记住"执行历史.同时,根据不同执行状态的历史,提出不同的行为约束.

定义 7(CNS-M)给定一个并发博弈结构 $S=\langle k,Q,\Pi,\pi,d,\delta\rangle$,一个带记忆的联盟规范系统(CNS-M)是一个元组 $P=\langle C,E,q_0,e_0,\vartheta,\tau\rangle$,其中,

1) $C\subseteq\{1,2,\ldots,k\}$是受到控制的联盟;
2) E 是一组(执行)上下文;
3) $q_0\in Q$ 是初始状态;
4) $e\in E$ 为初始上下文;

5) ϑ为行动函数,对所有的 $q\in Q$ 和 $e\in E$,有 $\vartheta(q,e)\subseteq D_C(q)$;

6) $\tau:Q\times E\times Q\to E$ 是上下文函数.

本文将控制联盟 C 的 CNS-M P 称为一个 C-规范*.在状态 q 和上下文 e 下,本文称一个联合行动是 P-相容的,当且仅当 $\exists\vec{m}_C\in\alpha(q,e)$, 使得 $\vec{m}|C=\vec{m}_C$.一个符合 P 的计算称为 P 的执行,这是由一个循环反复地感知状态,计算出当前执行的上下文,选择并执行一个符合 P 的联合行动,并不断重复的过程.因此,CNS-M 的执行可以描述成"状态-上下文"这种有序对之间的转移.在形式上,P 的一次执行是状态-上下文的一个无穷序列$\lambda=\langle q_0,e_0\rangle$, $\langle q_1,e_1\rangle,\langle q_2,e_2\rangle,\ldots$,并且满足:对于所有的位置 $i(i>0)$,都存在一个 P-相容的联合行动 $\vec{m}$,使得 $\delta(q_i,\vec{m})=q_{i+1}$ 且

$$\tau(q_i,e_i,q_{i+1})=e_{i+1}.$$

在并发博弈结构 S 中实施一个 CNS-M P 所得到的结构,指的是在 $\hat{S}(S,P)=\langle\hat{k},\hat{Q},\hat{\Pi},\hat{\pi},\hat{d},\hat{\delta}\rangle$ 实施一个 CNS $\hat{\Gamma}(S,P)=\langle\hat{C},\hat{\vartheta}\rangle$ 所得到的结构,即 $\hat{S}(S,P)\dagger\hat{\Gamma}(S,P)$, 其中,

1) $\hat{k}=k,\hat{\Pi}=\Pi,\hat{C}=C$;

2) $\hat{Q}$ 和 $\hat{\delta}$ 是由如下规则所产生的极小集:(1) $\langle q_0,e_0\rangle\hat{Q}$;(2) 如果 $\langle q,e\rangle\hat{Q}$ 且 $\exists\vec{m}\in D(q)$ 使得 $\delta(q,\vec{m})=q'$, $\tau(q,e,q')=e'$ 且 $\langle q',e'\rangle\in\hat{Q},\delta(\langle q,e\rangle,\vec{m})=\langle q',e'\rangle$;

3) $\forall\langle q,e\rangle\in\hat{Q}:\hat{\pi}(\langle q,e\rangle)=\pi(q),\hat{d}_i(\langle q,e\rangle)=d_i(q)$ 且 $\hat{\vartheta}(\langle q,e\rangle)=\alpha(q,e)$.

引理 3. 一个状态-上下文序列λ是 CNS-M P 的一次执行,当且仅当λ是 $\hat{S}(S,P)$ 的一个 $\hat{\Gamma}(S,P)$- 相容$\langle q_0,e_0\rangle$-计算.

证明:假设$\lambda=\langle q_0,e_0\rangle,\langle q_1,e_1\rangle,\langle q_2,e_2\rangle,\ldots$是 P 的一次执行.对于任意一个位置 $i\geqslant 0$,存在一个 P-相容联合行动 $\vec{m}$,使得 $\delta(q_i,\vec{m})=q_{i+1}$ 且 $\tau(q_i,e_i,q_{i+1})=e_{i+1}$.由 $\hat{\delta}$ 在 $\hat{S}(S,P)$ 中的定义可知,这等价于 $\hat{\delta}(\langle q_i,e_i\rangle,\vec{m})=\langle q_{i+1},e_{i+1}\rangle$.

因此,S 中所有被 P 所允许的计算可以被显式地编码进结构 $\hat{S}(S,P)\dagger\hat{\Gamma}(S,P)$ 中.

对于所有的公式$\varphi\in\mathcal{L}$,不能建立一个 C-规范来满足φ,当且仅当对于所有并发博弈结构 S,不存在任何 C-规范$^*P$,使得存在 $\langle q,e\rangle\in\hat{Q}$ 满足 $S,\Gamma_{\forall}^{C},q\nvDash\varphi$ 且 $\hat{S}(S,P),\hat{\Gamma}(S,P),\langle q,e\rangle\vDash\varphi$;不能建立一个 C-规范来避免φ当且仅当若对于所有并发博弈结构 S,不存在任何 C-规范$^*P$,使得存在 $\langle q,e\rangle\in\hat{Q}$ 满足 $S,\Gamma_{\forall}^{C},q\vDash\varphi$ 且

$$\hat{S}(S,P),\hat{\Gamma}(S,P),\langle q,e\rangle\nvDash\varphi. \qquad \square$$

定理 3. 给定一个 Co-ATL 公式φ,则有:

1) φ可以通过建立一个 C-规范*而满足,当且仅当φ可以通过建立一个 C-规范而满足;

2) φ可以通过建立一个 C-规范*而避免,当且仅当φ可以通过建立一个 C-规范而避免.

证明:本文只证明第 1 部分.假设对于一个并发博弈结构 S,存在一个 CNS $\Gamma=\langle C,\vartheta\rangle$,使得 $S,\Gamma_{\forall}^{C},q\forall=\varphi$ 且 $S,\Gamma,q\vDash\varphi$.定义一个 *CNS-M* $P=\langle C,\{e\},q,e,\vartheta',\tau\rangle$,其中,对于所有的 $q',q\in Q,\vartheta'(q,e)=\vartheta(q,e)$,且 $\tau(q,e,q')=e$.故,显然有

$$\hat{S}(S,P),\hat{\Gamma}(S,P),\langle q,e\rangle\vDash\varphi. \qquad \square$$

定理 3 本质指的是 C-规范*和 C-规范的能力限度是相同的.因此,类 C-规范*能力的限制也同样可以由 Co-ATL 的两个语言片段 $\mathcal{L}_c^+$ 和 $\mathcal{L}_c^-$ 来刻画.注意,这并不意味着扩展的联盟规范结构 CNS-M 是没有用的,因为对就一个固定的并发博弈结构 S 来说,C-规范*比 C-规范更有效:给定一个 Co-ATL 公式φ,对于 $\langle q,e\rangle\in\hat{S}(S,P)$, 存在一个 C-规范Γ,使得 $S,\Gamma,q\vDash\varphi\Rightarrow\exists C$-规范$^*P$: $\hat{S}(S,P),\hat{\Gamma}(S,P),\langle q,e\rangle\vDash\varphi$.但很容易给出一个反例表明,而反过来这一结果并不成立.

5 结 论

本文提出一种基于扩展 ATL 语义的联盟规范系统 CNS,并对其规范能力进行了形式化的界定.本文证明,对于任意联盟 C,关于 C 的联盟规范系统的能力极限可以由 Co-ATL 的两个语言片段 $\mathcal{L}_c^+$ 和 $\mathcal{L}_c^-$ 来界定.此外,本文认为,一个规范系统在本质上是对所有 agent(记为 Ag)的一个联盟规范系统.因此,规范系统的能力极限可以由两个语言片段 $\mathcal{L}_{Ag}^+$ 和 $\mathcal{L}_{Ag}^-$ 来刻画.容易看出: $\mathcal{L}^u\subset\mathcal{L}_{Ag}^+$ 和 $\mathcal{L}^e\subset\mathcal{L}_{Ag}^-$,所以可以得出结论:相比$\mathcal{L}^e$和$\mathcal{L}^u$, $\mathcal{L}_{Ag}^+$ 和 $\mathcal{L}_{Ag}^-$ 可以更

好地刻画规范系统的能力极限.同时,本文进一步对 CNS 进行了扩展,显式地包含执行上下文,直观上是指联盟可以在执行上下文中记录执行历史.这样,即使在同一状态,也可以根据所处的不同上下文做出不同的行为限制.本文证明,对 CNS 规范能力极限的界定,在这个扩展下同样有效.虽然 C-规范*比 C-规范在改变系统属性上更严格、更有效,但 CNS-M 的规范能力极限却仍和 CNS 是等同的.

References:

[1] Shoham Y, Tennenholtz M. On the synthesis of useful social laws for artificial agent societies. In: Proc. of the AAAI'92. 1992. 276−281. http://dl.acm.org/citation.cfm?id=1867178

[2] Shoham Y, Tennenholtz M. On social laws for artificial agent societies: Off-Line design. In: Agre PE, Rosenschein SJ, eds. Proc. of the Computational Theories of Interaction and Agency. 1996. 597−618. [doi: 10.1016/0004-3702(94)00007-N]

[3] Wooldridge M, van der Hoek W. On obligations and normative ability: Towards a logical analysis of the social contract. Journal of Applied Logic, 2005,3(3-4):396−420. [doi: 10.1016/j.jal.2005.04.006]

[4] van der Hoek W, Roberts M, Wooldridge M. Knowledge and social laws. In: Proc. of the AAMAS 2005. 2005. 674−681. [doi: 10.1145/1082473.1082576]

[5] van der Hoek W, Roberts M, Wooldridge M. Social laws in alternating time: Effectiveness feasibility, and synthesis. Synthese, 2007,156(1):1−19. [doi: 10.1007/s11229-006-9072-6]

[6] Ågotnes T, van der Hoek W, Juan A, Rodriguez-Aguilar CS, Wooldridge M. On the logic of normative systems. In: Proc. of the IJCAI 2007. 2007. 1175−1180. https://www.aaai.org/Papers/IJCAI/2007/IJCAI07-190.pdf

[7] Ågotnes T, van der Hoek W, Tennenholtz M, Wooldridge M. Power in normative systems. In: Proc. of the AAMAS 2009. 2009. 145−152. http://dl.acm.org/citation.cfm?id=1082576

[8] Alur R, Henzinger TA, Kupferman O. Alternating-time temporal logic. In: Proc. of the FOCS'97. 1997. 100−109. http://www-sst.informatik.tu-cottbus.de/~db/doc/People/LNCS/papers/15360023.pdf [doi: 10.1145/585265.585270]

[9] Alur R, Henzinger TA, Kupferman O. Alternating-Time temporal logic. Journal of the ACM, 2002,49(5):672−713. [doi: 10.1145/585265.585270]

[10] Pauly M, Wooldridge M. Logic for mechanism design—A manifesto. In: Proc. of the 2003 Workshop on Game Theory and Decision Theory in Agent Systems (GTDT 2003). 2003. http://www.csc.liv.ac.uk/~mjw/pubs/gtdt2003.ps.gz

[11] Wooldridge M, Agotnes T, Dunne PE, van der Hoek W. Logic for automated mechanism design—A progress report. In: Proc. of the AAAI 2007. 2007. 9−16. http://www.aaai.org/Papers/AAAI/2007/AAAI07-002.pdf

[12] Ågotnes T, Wooldridge M, van der Hoek W. Normative system games. In: Proc. of the AAMAS 2007. 2007. 876−883. [doi: 10.1145/1329125.1329284]

[13] Schobbens P. Alternating-Time logic with imperfect recall. Electronic Notes in Theoretical Computer Science, 2004,85(2):82−93. [doi: 10.1016/S1571-0661(05)82604-0]

王崇骏(1975-),男,江苏淮安人,博士,副教授,CCF 会员,主要研究领域为多 Agent 系统,人工智能.

张雷(1987-),男,博士生,主要研究领域为多 Agent 系统,人工智能.

吴骏(1981-),男,博士,讲师,CCF 学生会员,主要研究领域为多 Agent 系统逻辑及其应用.

谢俊元(1961-),男,教授,博士生导师,主要研究领域为人工智能.