

基于情境演算的智能体结构*

李斌^{1,2+}, 吕建¹, 朱梧楨¹

¹(南京大学 计算机软件新技术国家重点实验室,江苏 南京 210093)

²(扬州大学 计算机科学与技术系,江苏 扬州 225009)

An Agent Architecture Based on Situation Calculus

LI Bin^{1,2+}, LÜ Jian¹, ZHU Wu-Jia¹

¹(State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210093, China)

²(Department of Computer Science and Technology, Yangzhou University, Yangzhou 225009, China)

+ Corresponding author: Phn: 86-25-3303169, Fax: 86-25-3593670, E-mail: libin@softlab.nju.edu.cn

<http://www.nju.edu.cn>

Received 2002-05-15; Accepted 2002-08-14

Li B, Lü J, Zhu WJ. An Agent architecture based on situation calculus. *Journal of Software*, 2003,14(4): 733~742.

Abstract: Agent architecture is an important topic in the current research of Agent. Based on BDI model and situation calculus, an Agent architecture called AASC (Agent architecture based on situation calculus), which can depict various features, especially autonomy of Agent, is presented in this paper. Since AASC provides the facilities for representing mental states of Agent, such as belief, goal, strategy and so on, reasoning about action and planning, it can serve as a uniform platform for interpreting the autonomy of Agent and constructing various types Agents.

Key words: Agent architecture; mental state; reasoning about action; situation calculus; BDI model

摘要: Agent结构的建立是Agent研究的重要内容.尝试着结合BDI结构和情境演算的优点,提出了一个能够刻画Agent的多种特征,尤其是自主性的智能体结构AASC(Agent architecture based on situation calculus).此结构既能表示Agent的信念、目标、策略等心智状态,又能进行行动推理和规划,为解释Agent的自主性、建构不同类型的Agent提供了统一的平台.

关键词: Agent结构;心智状态;行动推理;情境演算;BDI模型

中图法分类号: TP18 **文献标识码:** A

* Supported by the National Natural Science Foundation of China under Grant No.60273034 (国家自然科学基金); the National High-Tech Research and Development Plan of China under Grant Nos.2001AA113110, 2002AA116010 (国家高技术研究发展计划(863)); the National Grand Fundamental Research 973 Program of China under Grant No.2002CB312002 (国家重点基础研究发展规划(973)); the Natural Science Foundation and the High Technology Project of Jiangsu Province of China under Grant Nos.BG2001012, BK2002203, BK2002409 (江苏省自然科学基金和高技术资助项目); the Key Project of Natural Science Foundation of the Education Department of Jiangsu Province of China under Grant No.01KJB520010 (江苏省教育厅自然科学基金重点资助项目)

第一作者简介: 李斌(1965—),男,江苏扬州人,博士,副教授,主要研究领域为Agent技术,人工智能.

目前关于 Agent 还没有一个能被学术界普遍接受的定义,一般认为,如果一个系统具有如下特征:自主性(autonomy)、应激性(reactivity)、主动性(pro-activity)和社会能力(social ability),就可以称其为智能体(Agent),其中自主性是 Agent 的最显著的特征^[1].如何设计并建造能够刻画上述特征的 Agent,成为 Agent 理论研究的重要内容.

实现 Agent 自主性等特征的关键因素在于如何解释与模拟 Agent 智能行为的产生机制.目前,人工智能领域广泛接受的一种观点认为,Agent 智能行为取决于 Agent 的心智状态,即 Agent 的信念、期望、意向、能力等心智要素是控制 Agent 智能行为的关键因素.因此,设计并构造 Agent 的关键在于如何刻画 Agent 的各种心智成分以及这些心智成分如何影响 Agent 的行为.AI 在这两方面都进行了相关的研究,分别称为“意向理论”与“行动理论”.

关于意向理论,目前最流行的是所谓 BDI 模型^[2,3],主要用于刻画 Agent 的各种心智成分及其相互约束关系.BDI 模型具有良好的语形与语义,理论较为完善,但尚存在大量未解决的问题^[3].另外,BDI 模型主要侧重于刻画 Agent 的心智成分及其相互约束关系,很少涉及关于行动本身的推理及面向目标的规划,这是使得 BDI 模型与实际 Agent 实现之间脱节的原因之一.

由于 Agent 往往处于动态的环境之中,要体现 Agent 的根本特征——自主性,仅有上述研究是不够的,还必须刻画各种心智成分如何随环境的变化而变化,这就需要 Agent 具有对行为的推理能力.目前,情境演算(situation calculus)^[4]是广为采用的行动推理形式框架,在其中不但能讨论行动对系统状态的影响,而且可以研究规划问题.但现有的情境演算理论由于缺乏心智状态的表示,难以在其中探讨心智成分与 Agent 行为之间的相互关系.

本文结合 BDI 模型和情境演算的优点构造了一个智能体结构 AASC(Agent architecture based on situation calculus),使其既能表示 Agent 的心智状态,又能进行行动推理和规划,能够实现 Agent 不同程度的自主性以及刻画 Agent 多种特征,从而为实际应用中根据特定的应用背景建构 Agent 提供必要的理论基础.

1 AASC 的描述语言 L

本节介绍本文用于描述 Agent 结构的描述语言 L .我们使用带等词的二型一阶语言 L 来表示 Agent 的心智状态和行动. L 有两种型:行动型 *action*,其论域为行动;对象型 *object*,其论域为除行动以外的所有对象. L 包含以下几个部分:

- 个体,型为 *object*,其中,个体常元通常用 c, c_1, c_2, \dots 表示;个体变元通常用 x, x_1, x_2, \dots 表示;
- 谓词符号,型为 $object^n \rightarrow \{\text{true}, \text{false}\}, n \geq 0$,通常用 P^n, P_1^n, P_2^n, \dots 表示;
- 函数符号,型为 $object^n \rightarrow object, n \geq 0$,通常用 f^n, f_1^n, f_2^n, \dots 表示;
- 行动符号,型为 $object^n \rightarrow action, n \geq 0$,通常用 A^n, A_1^n, A_2^n, \dots 表示, ε 表示空;
- 逻辑符号包括“ \neg ”,“ \wedge ”,“ \exists ”,“ T ”,“ F ”,“ $=$ ”,其中 T, F 分别表示 true 与 false;
- 行动联结词包括“ $?$ ”,“ $;$ ”,“ \cup ”,“ $*$ ”,“ π ”,“ \parallel ”,“ s ”,具体含义见下文.

L 语言中项及合适公式的形成规则与通常规定类似,此处略.不含自由变元的公式称为语句,不含自由变元的项称为封闭项.我们将不含行动符号的合适公式称为经典公式,其集合记为 $Form(L)$.

行动表达式通常用 α, β 表示,由以下规则生成:

$$\alpha ::= a(x_1, \dots, x_n) \mid \varphi? \mid \alpha; \beta \mid \alpha \cup \beta \mid \alpha^* \mid \pi \alpha(x) \mid \alpha \parallel \beta \mid s(\alpha) \mid c(x_1, \dots, x_n),$$

其中 $a(x_1, \dots, x_n)$ 是原子行动, $\varphi?$ 表示判断 φ 是否为真的行动,“ $\alpha; \beta$ ”,“ $\alpha \cup \beta$ ”,“ $\pi \alpha(x)$ ”和“ α^* ”分别表示行动的序列执行、非确定性选择、非确定性变量选择以及非确定性迭代, $\alpha \parallel \beta$ 表示行动 α 和 β 并发执行, $s(\alpha)$ 表示搜索 α 的可行行动序列.例如:假设 α 形为 $a_1; a_2 \cup b_1; b_2 \cup c_1$, 其中 a_1, a_2, b_1, b_2, c_1 均为原子行动,在当前情境下,行动 a_1 不可执行,行动 c_1 可执行,行动 b_1 可执行且在模拟执行行动 b_1 之后的情境下,行动 b_2 可执行,则 $s(\alpha)$ 可能为 $b_1; b_2$,其结果依赖于系统具体的实现方法. c 是复杂行动名,表示策略规则(参见下文)中定义的一个复杂行动.

语言 L 具有很强的表达能力,它能表达一阶公式和复杂行动,包括并发动作和不确定行动.

2 AASC Agent 的心智状态及其表示

尽管以“意向姿态”研究 Agent 的思路得到许多学者的肯定,但对于哪些心智成分是刻画 Agent 的自主性是必需与基本的,尚无定论,各种以不同心智成分为要素的 Agent 模型也是层出不穷的. BDI 学派认为智能体的行为由其信念(belief)、期望(desire)及意向(intention)决定,因此信念、期望及意向成为智能体模型的主要成分. 这种模型能成功地解释 Agent 在静态环境中的行为,但不能有效地解释其在动态环境中的行为. 其主要原因在于,这种 Agent 模型不能表示 Agent 行动的效果及 Agent 执行行动的能力,因此不能预测其行为对环境的影响. 本文将考虑一些新的心智成分用于反映 Agent 对自身行为的知识以及对外界环境的作用效应、Agent 完成复杂行动的经验等. AASC Agent 的心智属性包括信念、策略、目标和事件.

2.1 信念

在本文中,信念是 Agent 关于环境状态及行动的效应、可行性等的知识,具体包含下述几类信念:

① 状态信念.

用于刻画 Agent 对环境及其他 Agent 的认识,它是 Agent 判断语句真假的基础. 本文用 L 中的一阶语句来表示.

② 关于基本能力的信念.

用于表示 Agent 能执行的原子行动,分为两类:

- 与领域有关的原子行动,由用户根据具体问题给出;
- 与通信有关的原子行动.

③ 效应信念.

用于刻画原子行动如何引起环境状态变化的知识,即行动的效果. 用两种关系流效应规则、一种函数流效应规则这 3 种规则来表示 Agent 的原子行动效果. 流是情境演算中的基本概念,简单地说,它刻画了取值依赖于情境的关系或函数,关于情境演算的详细介绍见文献[4],这里不再赘述.

3 种效应规则的语法形式分别为:

- 关系流正效应规则

$$(R(\bar{x}), F, A(\bar{y}), T, \Phi(\bar{x}, \bar{y})).$$

其含义是,对任意的 \bar{a}, \bar{c} , 如果当前 $R(\bar{a}/\bar{x})$ 为假,那么执行行动 $A(\bar{c})$ 以后 $R(\bar{a}/\bar{x})$ 为真的充要条件是 Agent 相信 $\Phi(\bar{a}/\bar{x}, \bar{c}/\bar{y})$ 为真. 其中 R 为关系流, A 为原子行动, $R(\bar{a}/\bar{x})$ 是指对 R 中所有 \bar{x} 的出现用 \bar{a} 替换.

- 关系流负效应规则

$$(R(\bar{x}), T, A(\bar{y}), F, \Phi(\bar{x}, \bar{y})).$$

其含义是,对任意的 \bar{a}, \bar{c} , 如果当前 $R(\bar{a}/\bar{x})$ 为真,那么执行行动 $A(\bar{c})$ 以后 $R(\bar{a}/\bar{x})$ 为假的充要条件是 Agent 相信 $\Phi(\bar{a}/\bar{x}, \bar{c}/\bar{y})$ 为真.

- 函数流效应规则

$$(F(\bar{x}), Z_1, A(\bar{y}), Z_2, \Phi(\bar{x}, \bar{y}, Z_1, Z_2)).$$

其含义是,对任意的 $\bar{a}, \bar{c}, d_1, d_2, (F(\bar{a}), d_1, A(\bar{c}), d_2, \Phi(\bar{a}, \bar{c}, d_1, d_2))$ 表示: 如果当前 $F(\bar{a})=d_1$, 那么执行行动 $A(\bar{c})$ 以后 $F(\bar{a})=d_2$ 的充要条件是,从当前的状态信念集中能得出 $\Phi(\bar{a}/\bar{x}, \bar{c}/\bar{y}, d_1, d_2)$ 为真. 其中 F 为函数流.

④ 关于可行性的信念.

用于表征原子行动能够执行的预决条件. 我们用具有如下形式的预决条件规则来表示:

$$(A(\bar{x}), \Phi(\bar{x})), \quad \text{其中 } A \text{ 为原子行动, } \Phi(\bar{x}) \text{ 为一阶公式.}$$

其含义是,对任意的 \bar{a} , 只要从当前的状态信念集中能够得出 $\Phi(\bar{a}/\bar{x})$ 为真, Agent 就认为行动 $A(\bar{a}/\bar{x})$ 是可行的.

2.2 策略

策略用于刻画 Agent 执行复杂行动的经验,它在 Agent 的自主决策方面起着十分重要的作用. 我们使用策略规则来表达 Agent 的经验.

策略规则的一般形式为 $(c(\bar{x}), \alpha(\bar{x}))$.其中, c 是一个策略名, $\alpha(\bar{x})$ 为一个行动表达式,即策略 $c(\bar{x})$ 的具体实施方案为 $\alpha(\bar{x})$.

2.3 目 标

目标表示 Agent 希望实现的目标,它的构成涉及如下 3 种成分:

- 行动:用封闭行动表达式表示,指出 Agent 为实现目标应采取的行动.
- 条件:用一阶语句表示,指明 Agent 达到目标后应满足的条件.
- 权值:用正有理数表示,指出该目标的优先级.

目标的一般形式为 $(\alpha(\bar{t}), g(\bar{t}), r)$.其含义是,如果目标 $g(\bar{t})$ 还没有实现(即 $g(\bar{t})$ 不能从 Agent 当前的状态信念集推出),权值 r 是所有可执行的目标和事件中权值最大的,那么 Agent 将执行行动 $\alpha(\bar{t})$.

2.4 事 件

事件表示 Agent 需作出及时反应的突发事件.这是因为 Agent 处于复杂的现实环境中,我们要求构造出的 Agent 不仅能主动地根据目标执行相应的行动,而且能对外部环境的突发事件作出及时反应.它的构成涉及如下 3 种成分:

- 行动:用封闭行动表达式表示,指出 Agent 响应事件应采取的行动.
- 前提条件:用一阶语句表示,指明 Agent 响应事件应满足的前提条件.
- 权值:用正有理数表示,指出该事件的优先级.

事件的一般形式为 $(\alpha(\bar{t}), p(\bar{t}), r)$.其含义是,如果响应事件的前提条件 $p(\bar{t})$ 满足(即 $p(\bar{t})$ 能从 Agent 当前的状态信念集推出),权值 r 是所有可执行的目标和事件中权值最大的,那么 Agent 将执行行动 $\alpha(\bar{t})$.

基于上述要素,我们定义 AASC Agent 的心智状态如下:

定义 1. AASC Agent 的心智状态是一个七元组 $M=(Bel, Act, Eff, Qual, Stra, Goals, Evt)$,其中:

- Bel 是状态信念集合,其元素为一阶语句;
- Act 是原子行动集;
- Eff 是效应集,其中每一原子行动、流对最多对应一条效应规则;
- $Qual$ 是预决条件集,其中每一原子行动最多对应一条预决条件规则;
- $Stra$ 是策略集;
- $Goals$ 是目标集;
- Evt 是事件集.

3 自主 Agent 的结构 AASC

上一节我们给出了 Agent 的心智成分及其表示方法,这为进一步讨论 Agent 的自主决策能力奠定了基础.现在我们对 Agent 的自主性作进一步的形式分析,并提出一个能够刻画自主 Agent 的结构 AASC.

在文献[5]中,Agent 的自主性定义为:可在没有人或其他 Agent 的直接干预下为实现目标持续运行,并能控制其内部的行动和状态转换.换言之,一个自主 Agent 的控制能力反映在如下两个侧面上:选择行动的能力和内部状态转换的能力.我们认为,在这种自主性定义下,自主 Agent 应具备下列特性:

- 接受外界信息以后,能判断是否接受,如果接受,能自动地更新或修正其内部状态;
- 基于自身的信念可以判断语句的真假;
- 能根据信念、目标和策略等状态自动地决定应采取的行动;
- 执行行动之后,能自动地更新或修正其内部状态.

因此,如下 3 种能力是自主 Agent 所必需的:

- 推理能力.

当 Agent 判断语句的真假、预测行动的可行性或决定执行条件时,推理能力是必需的.尽管用一阶语句表示 Agent 的信念,我们并不要求 Agent 具有完全的一阶逻辑推理能力.现在我们假定存在推理算子 Rea ,它可能是单

调的或非单调的,也可能是协调的或非协调的,通过推理算子 Rea , Agent 能够判断语句是真、假或者不确定. 特别地,我们用 $\varphi \in Rea(B)$ 表示 Agent 在当前环境信念集下相信 φ 为真.

- 心智状态的改变能力.

当 Agent 接受外界信息或执行行动以后,其心智状态必须随外界信息或行动发生变化,因此自主 Agent 应该知道如何改变它的心智状态. 我们用函数 Upd 刻画心智状态更新(update)过程,函数 Rev 表示心智状态修正(revision)过程.

- 决策能力.

体现 Agent 自我控制能力的一个重要方面是, Agent 在某一状态下,应该能够通过决策来决定其未来的行动. 针对上一节提出的 Agent 心智成分,我们用目标、事件选择函数 Sel_{ge} 和行动选择函数 Sel_a 表示 Agent 自我决定的过程.

基于情境演算的自主 Agent 结构 AASC 定义如下:

定义 2. 自主 Agent 是一个六元组 $(M, Rea, Upd, Rev, Sel_{ge}, Sel_a)$. 其中, M 表示 Agent 的非空心智状态集; Rea 表示推理算子; $Upd: M \times A \rightarrow M$, 称为心智状态更新函数, A 表示 Agent 的封闭原子行动项的集合; $Rev: M \times Form(L) \rightarrow M$, 称为心智状态修正函数; $Sel_{ge}: M \rightarrow Goals \cup Evt$, 称为目标与事件选择函数; $Sel_a: M \times (Goals \cup Evt) \rightarrow A$, 称为行动选择函数. 函数 Upd, Rev, Sel_{ge} 和 Sel_a 形式化地刻画 Agent 的自主性, 决定了 Agent 的运行.

基于 AASC 结构构造的自主 Agent 的执行过程如下:

- ① 接受外界信息, 根据函数 Rev 修正心智状态;
- ② 根据函数 Sel_{ge} 选择目标 g 或事件 e ;
- ③ 根据函数 Sel_a 选择行动 a ;
- ④ 执行行动 a ;
- ⑤ 根据函数 Upd 更新心智状态.

4 AASC Agent 解释器的设计与实现

由上述讨论可知, AASC Agent 的执行由函数 Upd, Rev, Sel_{ge} 和 Sel_a 决定, 因此, AASC 解释器的设计取决于上述函数的构造方法. 下面我们介绍 AASC 解释器的设计及其 Java 实现.

4.1 AASC Agent 解释器的设计

4.1.1 函数 Rea, Rev 和 Upd 的构造

我们确定 Rea 为经典逻辑推理算子.

心智状态的更新与心智状态的修正有着本质上的差别, 这种差别来自于两者不同的研究背景. 当 Agent 执行行动或接受外界信息以后, 心智状态的变化究竟是选择更新还是修正, 目前还没有很好的解决方法. 在本系统中, Agent 执行某种内部行动引起自身状态或运行环境的改变将导致 Agent 心智状态的更新; Agent 执行某种与外界的交互行动获得新的信息, 这些新的信息可能引起 Agent 的心智状态的变化, 此时用心智状态的修正算子来刻画. 信念修正算法可以根据系统实现的需要, 选用独立于语形的计算算法或基于语形的计算算法, 信念更新算子可以利用 Winslett 的可能模型方法(PMA), 详细内容可参见文献[6].

4.1.2 函数 Sel_{ge} 的构造

对于任意的 n 元组 $d=(d_1, \dots, d_n)$, 记 $d_{\cdot [i]}=d_i$, 其中 $1 \leq i \leq n$.

定义 3. 假设一个七元组 $M=(Bel, Act, Eff, Qual, Stra, Goals, Evt)$ 为 Agent 的心智状态, 如果存在 $i \in Goals \cup Evt$ 且满足下列条件, 则称 i 为 Agent 的候选目标事件, 由 i 组成的集合称为 Agent 的目标事件候选集, 用 I 表示.

- ① $i_{\cdot [1]} \neq \epsilon$;
- ② $i \in Goals \wedge \neg i_{\cdot [2]} \in Rea(Bel) \vee i \in Evt \wedge i_{\cdot [2]} \in Rea(Bel)$;
- ③ $\forall j \in I$, 且 j 满足条件①和条件②, 有 $i_{\cdot [3]} \geq j_{\cdot [3]}$.

上述定义的含义是: 从目标集 $Goals$ 和事件集 Evt 中选取权值最大, 目标未实现, 事件的前提条件满足, 且实

现此目标或事件的行动表达式不为空的目标或事件.

函数 Sel_{ge} 的构造如下:

定义 4. 假设 Agent 的心智状态为 $M=(Bel,Act,Eff,Qual,Str,Goals,Evt)$,则函数 Sel_{ge} 定义为:如果集合 I 不为空,则 $Sel_{ge}(M)$ 的结果为在 I 中随机挑选一个候选目标或事件;如果集合 I 为空,则 $Sel_{ge}(M)$ 的结果为空.

4.1.3 函数 Sel_a 的构造

定义 5. 假设 Agent 的心智状态为 $M=(Bel,Act,Eff,Qual,Str,Goals,Evt)$,函数 Sel_{ge} 选择的目标或事件为 i ,如果 $i_{[1]} \neq \varepsilon$,则行动 $i_{[1]}$ 为实现 i 的方案.

定义 6. 假定 δ 是一封闭行动表达式,则 δ 的单步可行分解(简记为 SFD)是一个二元组 (a, σ) ,其中 a 为 δ 要执行的第 1 个行动, σ 表示剩余的行动序列, δ 的单步可行分解可归纳定义如下:

① δ 为原子行动 a ,如果存在预决条件规则 $q \in Qual$,满足 $q_{[1]}(\bar{t})=a$ 且 $q_{[2]}(\bar{t}) \in Rea(Bel)$,则 δ 的 SFD 为 (a, ε) ;反之,置 δ 的 SFD 为 (Θ, a) ,其中 Θ 是特殊的原子行动,表示停止.

其含义为:如果原子行动 a 预决条件得到满足,则在当前环境下可执行行动 a ;反之则停止.

② δ 为测试行动 $\varphi?$,如果 $\varphi \in Rea(Bel)$,则 $(\varphi?, \varepsilon)$ 是 δ 的 SFD;反之, $(\Theta, \varphi?)$ 是 δ 的 SFD.

③ δ 为串联行动 $\delta_1; \delta_2$,如果 (a, δ'_1) 是 δ_1 的 SFD,则 $(a, \delta'_1; \delta_2)$ 是 δ 的 SFD.这里 $\varepsilon, \delta \neq \delta$.

④ δ 为不确定选择行动操作 $\delta_1 \cup \delta_2$,如果 $(a, \delta'_1), (a, \delta'_2)$ 分别是 δ_1 和 δ_2 的 SFD,则 (a, δ'_1) 和 (a, δ'_2) 均为 δ 的 SFD.

⑤ δ 为不确定选择参数操作 $\pi x \delta(x)$,如果对任意个体常元 $c, (a, \delta'(c))$ 是 $\delta(x)$ 的 SFD,则 $(a, \delta'(c))$ 是 δ 的 SFD.

⑥ δ 为不确定迭代操作 δ^* ,如果 (a, δ') 是 δ 的 SFD,则 $(\varepsilon, \varepsilon)$ 和 $(a, \delta'; \delta^*)$ 均为 δ 的 SFD.

⑦ δ 为并行行动 $\delta_1 \parallel \delta_2$,如果 $(a, \delta'_1), (a, \delta'_2)$ 分别是 δ_1 和 δ_2 的 SFD,则 $(a, \delta'_1 \parallel \delta_2)$ 和 $(a, \delta_1 \parallel \delta'_2)$ 均为 δ 的 SFD.这里 $\varepsilon \parallel \delta \neq \delta$.

⑧ δ 是复杂行动,如果存在策略规则 $s \in Stra$ 满足 $s_{[1]}(\bar{t}) = \delta$,只要 $(a(\bar{t}), \delta')$ 为 $s_{[2]}$ 的 SFD,则 $(a(\bar{t}), \delta')$ 为 δ 的 SFD.

⑨ δ 为搜索行动 $s(\delta_1)$,则 δ 的单步可行分解可由下述方法得到:

首先根据文献[4,7]定义宏 $Do(\delta_1, s', s'')$,其中: s', s'' 表示情境.宏 $Do(\delta_1, s', s'')$ 可扩展成二阶情境演算公式,表示在情境 s' 下,执行 δ_1 中的行动后的情境为 s'' ;

然后构造一个定理证明器来证明 $M_{sc} \models \exists s' Do(\delta_1, [], s')$,如果这是可证的,则得到一个原子行动序列 $a_{11}; a_{12}; \dots; a_{1m}$,记为 $\alpha_1 = a_{11}; \beta_1$;用定理证明器重复上述证明,将得到的原子行动序列分别记为 $\alpha_2 = a_{21}; \beta_2, \dots, \alpha_k = a_{k1}; \beta_k$;

最后, $s(\delta_1)$ 的 SFD 由以下步骤得到:在 $a_{11}, a_{21}, \dots, a_{k1}$ 中随机选择一个行动 a ,令 γ 为所有满足 $\exists j(1 \leq j \leq k)(a_{j1} = a$ 且 $\alpha_j = a_{j1}; \beta)$ 这一条件的 β 的非确定合成(即将这些 β 用不确定算子 \cup 联结),则 $(a, s(\gamma))$ 为 $s(\delta_1)$ 的 SFD.

上述方法的含义是,在当前情境下, δ_1 的执行路径可能有若干条,有些路径未必能完成, s 操作将在能完成的路径中选择一个可行行动 a 执行.如果第 1 个行动为 a 且能完成的路径有若干条,则单步执行行动 a 以后, δ_1 将转变为这些路径除去 a 以后剩余路径的并,记为 γ ,对 γ 将用上述方法同样处理.事实上,选择路径的 s 操作是通过一个规划器来实现的,它以离线的方式执行,因此,策略体 δ 的执行是在线执行与离线规划相结合的方式.

函数 Sel_a 的构造方式如下:

定义 7. 假设 Agent 的心智状态为 $M=(Bel,Act,Eff,Qual,Str,Goals,Evt)$,函数 Sel_{ge} 选择的目标或事件为 i, δ 为实现 i 的封闭行动表达式,则函数 Sel_a 定义如下:

$$Sel_a(M, i) = \begin{cases} a, & \text{如果 } \delta \neq \varepsilon \text{ 且 } (a, \delta') \text{ 是 } \delta \text{ 的 SFD 且 } a \neq \Theta \\ \Theta, & \text{否则} \end{cases}$$

4.2 AASC Agent 解释器的实现

在上一节讨论的基础上,我们用 Java 语言实现了一个 AASC Agent 解释器.此解释器主要包括以下 3 种数据类型:

- 基本数据类型.

它是构造其他类的基础,主要包括表达式类、条件类和表类。

- 心智成分类。

表示 Agent 的各心智成分,主要包括以下各类:

① 类 BeliefSet:表示 Agent 的状态信念,它封装了对状态信念的增加、删除、修改、匹配等操作。

② 类 Action、类 Context 和类 Effector:分别表示 Agent 能执行的行动、行动执行的前提条件以及行动执行后的效应。其中,行动分为心智成分操作、系统内置行动、与领域有关的原子行动等。

③ 类 StrategyConstruct:表示行动联结符。其中,行动联结符有顺序联结符、循环联结符、分支联结符、不确定选择行动联结符、非中断执行联结符以及搜索联结符等。

④ 类 StrategySet:表示 Agent 的策略集。

⑤ 类 Goal 和类 Event:分别表示 Agent 的目标类和事件类,它封装了用于判断目标是否已实现、事件的前提条件是否满足、计算目标和事件的权值、执行相应策略中的行动表达式等方法。

⑥ 类 IntentionStructure:表示 Agent 的意向集,它封装了增加意向、删除意向、执行意向的方法。

- 解释器类。

解释器类 Interpreter 封装了 AASC Agent 执行过程的方法,此方法的类 Java 描述如下:

```
public int run(){
    //循环直到所有目标全部实现
    while (true){
        //执行监控策略
        return Value=intentionStructure.executeStategy(observer);
        //如果目标已全部实现,结束
        if (intentionStructure.allGoalsDone()){
            return 0;
        }
        //产生完成目标或事件的可行策略集
        asl=new ASL(strategySet,beliefSet,intentionStructure);
        //选择权值最大的可行策略,表示有意向执行此策略
        if (asl.getsize()!=0){
            selectedElement=asl.getHighestRandomRatingValue();
            intentionStructure.intend(selectedElement);
        }
        //执行意向集中权值最大的意向
        intentionStructure.run();
    }
}
```

4.3 应用实例

本节通过具体实例来阐述如何利用 AASC 构造一个简单的用于电梯控制的 Agent。此 Agent 能够根据各层的呼叫请求来控制电梯执行上下移动及开关门动作,以响应各层的呼叫请求。基于上述 AASC,电梯控制 Agent 的心智状态构造如下:

- 状态信念集:

floor //电梯当前所在楼层,初始值为 1

lights //各层的呼叫请求,其数据结构为表.lights(N)为 1 表示第 N 层有呼叫请求,为 0 表示第 N 层无呼叫请求

opened //电梯门当前状态,为真表示开,为假表示关,初始值为假

• 原子行动:

down //电梯向下移动一层

up //电梯向上移动一层

off(N) //关第 N 层呼叫请求

open_doors(N) //开第 N 层电梯门

close_doors(N) //关第 N 层电梯门

• 预决条件规则:

① (down, $\neg(\text{floor}=1 \vee \text{opened})$);

② (up, $\neg(\text{floor}=6 \vee \text{opened})$) //假设楼层共 6 层;

③ (off(N), $N \geq 1 \wedge N \leq 6 \wedge \text{floor}=N \wedge \text{lights}(N)=1 \wedge \text{opened}$);

④ (open_doors(N), $N \geq 1 \wedge N \leq 6 \wedge \text{floor}=N \wedge \text{lights}(N)=1$);

⑤ (close_doors(N), $N \geq 1 \wedge N \leq 6 \wedge \text{floor}=N \wedge \text{opened}$).

• 效应规则:

① (floor, M , up, N , $N=M+1$)

//如果电梯当前所在楼层为 M ,则执行行动 up 以后,电梯所在楼层为 $M+1$

② (floor, M , down, N , $N=M-1$)

//如果电梯当前所在楼层为 M ,则执行行动 down 以后,电梯所在楼层为 $M-1$

③ (lights, $L1$, off(N), $L2$, repl($L1, L2, N, 0$))

//执行行动 off(N)后,第 N 层的呼叫请求由真变为假, repl($L1, L2, N, 0$)表示将 $L1(N)$ 置为 0,其他不变,其结果赋

予 $L2$

④ (opened, F , open_doors(N), T , true)

//如果电梯门状态为关,则执行行动 open_doors(N)以后的状态为开

⑤ (opened, T , close_doors(N), F , true)

//如果电梯门状态为开,则执行行动 close_doors(N)以后的状态为关

• 目标集:

(control, $\forall n(\text{lights}(n)=0), 1$).

• 策略规则:

① go_floor(N) //电梯运行到第 N 层

(go_floor(N), while $\neg(\text{floor}=N)$ do if floor < N then up else down endif endwhile)

② serve_floor(N) //响应第 N 层呼叫请求

(serve_floor(N), go_floor(N); open_doors(N); off(N); close_doors(N))

③ handle_reqs(Max) //响应距离当前电梯不超过 Max 的楼层呼叫请求

(handle_req(Max), $\pi m(\text{if}(\text{lights}(n)=1 \wedge \text{abs}(\text{floor}-n) \leq \text{Max}) \text{ then } \text{serve_floor}(n) \text{ endif})$)

④ minimize_motion(Max) //搜索距离当前电梯不超过 Max 的楼层呼叫请求

(minimize_motion(Max), handle_reqs(Max) $\cup \pi m(m=\text{Max}+1)?; \text{minimize_motion}(m)$)

⑤ control //电梯进入运行状态

(control, s(minimize_motion(0)))

其中:

if ϕ then δ endif =_{def} $\phi?; \delta$

if ϕ then δ_1 else δ_2 endif =_{def} [$\phi?; \delta_1$] \cup [$\neg\phi?; \delta_2$]

while ϕ do δ endwhile =_{def} [$\phi?; \delta^*$; $\neg\phi?$]

ϕ 为条件表达式, δ, δ_1 和 δ_2 为行动表达式.

假设当前情境下第 3 层和第 5 层发出呼叫请求,则 lights(3)和 lights(5)均为 1. Agent 解释器检查目标条件不

满足,以在线执行与离线规划相结合的方式执行 control 策略体,通过离线规划器搜索出距离当前电梯最近的楼层呼叫请求 3,响应后产生的原子行动序列为 up,up,open_doors(3),off(3),close_doors(3).Agent 解释器重复上述过程,直到目标条件满足为止.

5 相关工作

Agent 结构是建造 Agent 的一套特定方法,建造 Agent 的方法可分为 3 类^[8]:① 基于行为(behavior-based)的结构.典型的工作是 MIT 的 Brooks 提出的所谓包容结构(subsumption architecture)^[9].Brooks 认为,Agent 不需要知识,不需要表示,也不需要推理,Agent 可以像人类一样逐步进化,Agent 是由用于完成任务的行为构成的分层结构,这些行为相互竞争以获得对机器人的控制权.其优点是,Agent 能及时而快速地响应外来信息和环境的变化,甚至解决了传统符号 AI 很难解决的问题.其缺点是,其智能程度较低,缺乏足够的灵活性.② 层次结构(layered architecture).层次结构的系统通常被设计成至少包括如下两部分:高层是一个包含符号世界模型的认知层,它用传统符号 AI 的方式处理规划和进行决策;低层是一个能够快速响应和处理环境中突发事件的反应层,它不使用任何符号表示和推理系统,反应层通常被给予更高的优先级.典型的工作是 INTERRAP^[10].它从工程的角度提出了设计 Agent 的分层结构.Agent 包括知识库、世界界面和 3 个等级的控制层,定义了思维状态的种类和修正这些思维状态的基本功能及感知和行动的功能关系.其优点是,Agent 具有较高的智能、较强的灵活性以及快速响应性.其缺点是,各层需要不同的规范语言,如反应层语言、规划层语言以及协调层语言等;另外,与 BDI 结构相比,缺乏完善的理论基础.③ BDI 结构.典型的工作有 PRS,JAM,JACK 等^[11,12].BDI 结构一般能显式地表示 Agent 的信念、期望、意向和规划等部分,采用感知-思考-行动(observe-think-act)的执行方式.其优点是具有完善的理论基础、较强的灵活性以及快速响应性.其缺点是,缺乏行动本身的推理及面向目标的规划.我们提出的结构结合了 BDI 模型和情境演算的优点,具体反映在如下几个方面:区分目标和事件,并在结构中显式地加以表示,这是因为目标和事件分别刻画了 Agent 的两个不同特征,目标刻画 Agent 的主动性,而事件刻画 Agent 的应激性;显式地表示了 Agent 的原子行动、预决条件和效应,使得 Agent 具有对行为的推理能力;采用在线执行与离线规划相结合的执行方式选择 Agent 的行动,增强了 Agent 面向目标的规划能力.值得指出的是,我们提出的结构虽然含有 BDI 成分,但是我们着重于用心智状态来解释 Agent 的自主性、面向目标等特征,而不是刻画这些心智状态,这使得我们能够避免使用模态逻辑来表示 Agent 的心智成分.

其他相关的工作有多伦多大学认知机器人小组采用情境演算刻画机器人和 Agent,提出了 Agent 程序设计语言 Golog/ConGolog^[7],其语法是行动理论,语义是基于情境演算推理.虽然我们的基本思想和心智状态的表示部分地源于情境演算,但是我们提出的结构不能完全用情境演算语言来表达.另外,他们的工作只局限于行动理论在面向 Agent 程序设计语言上的应用,尚未将这一理论应用于 Agent 的结构建立中,特别是未能将行动理论用于解释 Agent 的自主行为.

6 结论

本文尝试着结合 BDI 结构和情境演算的优点,提出了一个能够刻画 Agent 的多种特征,尤其是自主性的智能体结构 AASC.此结构既能表示 Agent 的信念、目标、策略等心智成分,又能进行行动推理和规划,为解释 Agent 的自主性提供了统一的平台.基于上述 Agent 形式框架,我们开发了一个原型支撑系统 IASS.进一步的工作包括如何对已建立的 Agent 结构进行扩充,增加 Agent 学习功能,增强 Agent 对环境动态变化的适应性;与移动 Agent 合作 Agent 方面的研究成果相结合探讨如何建立移动的、合作的、智能体结构,进一步拓展 AASC 结构的适用范围.

References:

- [1] Jennings NR. An Agent-based approach for building complex software systems. Communications of the ACM, 2001,44(4):35-41.

- [2] Georgeff MP, Pell B, Pollack M, Tambe M, Wooldridge M. The belief-desire-intention model of agency. In: Muller JP, Singh M, Rao A, eds. Proceedings of the 5th International Workshop on Intelligent Agents V: Agent Theories, Architectures, and Languages (ATAL'98). Paris: Springer-Verlag, 1999. 1~10.
- [3] Rao AS, Georgeff MP. Decision procedures of BDI logics. Journal of Logic and Computation, 1998,8(3):293~344.
- [4] Reiter R. Knowledge in Action: Logical Foundations for Describing and Implementing Dynamical Systems. Cambridge, MA: MIT Press, 2001.
- [5] Castelfranchi C. Guarantees for autonomy in cognitive agent architecture. In: Woodridge M, Jennings NR, eds. Intelligence Agents: Theories, Architectures, and Languages. Amsterdam: Springer-Verlag, 1995. 56~70.
- [6] Katsuno H, Mendelzo AO. On the difference between updating a knowledge base and revising it. In: Fikes R, Sandewall E, eds. Proceedings of the Knowledge Representation and Reasoning. Cambridge, MA: Morgan Kaufmann Publishers, 1991. 387~395.
- [7] Giacomo GD, Lesperance Y, Levesque HJ. CONGOLOG, a concurrent programming language based on the situation calculus. Artificial Intelligence, 2000,121(1-2):109~169.
- [8] Wooldridge MJ, Jennings NR. Intelligent Agent: Theory and practice. Knowledge Engineering Review, 1995,10(2):115~152.
- [9] Brooks BA. A robust layered control system for a mobile robot. IEEE Journal of Robotics and Automation, 1986,2(1):14~23.
- [10] Muller JP. The Design of Intelligent Agents: A Layered Approach. Berlin: Springer-Verlag, 1996.
- [11] Huber MJ. Jam: A BDI-theoretic mobile agent architecture. In: Etzioni O, ed. Proceedings of the 3rd International Conference on Autonomous Agents. Seattle, WA: ACM Press, 1999. 236~243.
- [12] Busetta P, Rönquist R, Hodgson A, Lucas A. Jack intelligent Agents: Components for intelligent Agents in Java. Technical Report, Agent Oriented Software (AOS) Pty Ltd., 2001. <http://www.agent-software.co.uk/shared/home/index.html>.

全国第 7 届计算语言学联合学术会议(JSCL 2003)

征文通知

中国中文信息学会、中国计算机学会、中国人工智能学会和北京市语言学会于 2003 年 8 月 8 日~11 日在哈尔滨市与哈尔滨工业大学联合举办“全国第 7 届计算语言学联合学术会议(JSCL 2003)”。

一、征文范围

- (1) 计算语言学的理论基础: 知识表示、语义学、语用学、语料库语言学、记忆模型、机器学习、知识获取和推理技术;
- (2) 现代汉语的句法分析和语义分析: 汉语分析的策略、句法分析和语义分析中的计算问题、汉语分析的展望;
- (3) 汉语语料库技术及系统;
- (4) 汉语人机接口技术及系统;
- (5) 机器翻译技术、系统及评测方法;
- (6) 话语和篇章的分析与生成: 话语的心理学和语言学模型、篇章分析、话语生成;
- (7) 自然语言处理的应用系统: 汉语自动分词系统、智能检索系统、自动文摘系统、自动校对系统、文本自动分类系统、信息抽取、信息过滤、智能搜索引擎、文本挖掘、智能拼音汉字转换等;
- (8) 计算语言学的资源研究及建设: 树库、语法词典、词汇语义分类体系和语义词典、汉语分词词表、概念词典、知识库等;
- (9) 服务于计算语言学的支撑环境和软件技术。

二、来稿要求

全文不超过 8000 字, 每篇论文均应有中英文两种文字标题、作者、姓名、单位和不超过 200 字的摘要。来稿全文一式 3 份, 作者请自留底稿。会议概不退稿。大会录用的论文将收入有出版书号的会议论文集。

三、重要日期

- (1) 截稿日期: 2003 年 4 月 1 日(以邮戳为准) 注: 来稿请在首页上标明“JSCL 2003”。
- (2) 录用通知发出日期: 2003 年 5 月 1 日
- (3) 作者提交的论文激光印刷版日期: 2003 年 6 月 1 日(以到达日期为准)

四、联系方式

来稿邮寄地址: 100084 清华大学计算机科学与技术系 陈群秀 收
联系电话: 010-62781479