

Agent 的 PASS-BDI 模型

樊 玮^{1,2}, 陈增强², 袁著祉²

(1. 中国民航学院软件基地, 天津 300300; 2. 南开大学信息技术科学学院, 天津 300071)

[摘要] Agent 的 BDI 模型主要基于 Bratman 提出的理性主体理论, 未能充分体现 Agent 主动性的认知过程, 使 Agent 的问题求解和心智状态之间的关系未得到充分表达, 且难于编程实现。将 Das 等人提出的计划—注意—同时性加工—继时性加工认知过程模型引入 Agent 的理论研究中, 建立了 Agent 的 PASS-BDI 模型, 利用多价 π -演算从心智状态, 认知过程, 整体行为等几个方面刻画了该模型的静态和动态特征, 并将研究结果应用于多 Agent 组织中 Agent 行为特征的刻画。PASS-BDI 模型加强了对 Agent 主动性认知过程的刻画, 且易于编程实现。

[关键词] 主体; π -演算; 认知过程; 心智状态

[中图分类号] TP18 **[文献标识码]** A **[文章编号]** 1009-1742(2004)06-0043-07

1 引言

目前对 Agent 理论模型的研究主要基于 Bratman 提出的理性主体理论^[1]。其研究核心是对 Agent 心智状态的描述, 人们侧重研究 Agent 的信念 (belief)、愿望 (desire)、意图 (intention) 的关系和形式化描述, 建立 Agent 的 BDI 模型。主要工作包括 Moore 的形式逻辑建模^[2], Cohen 和 Levesque 使用正规模态逻辑的可能世界模型对 Agent 的 BDI 特性进行描述^[3], Rao 和 Georgeff 提出了理性 Agent 的 BDI 模型, 并用非线性的分支时态逻辑建模^[4], 以及最近刘贵全等人基于情景演算和三值逻辑对多 Agent 进行建模^[5]等。但基于 BDI 对 Agent 的建模主要存在以下几个方面的问题: a. 所建立的 Agent 模型和具体的 Agent 结构之间缺乏明确的映射关系, 难于编程实现 Agent 的抽象模型; b. BDI 模型对 Agent 认知过程的描述未能充分体现其主动性; c. 由于标准的可能世界语义的不足, 即逻辑全知问题, 使得基于上述理论建立的 Agent 模型在逻辑上不承认无知^[6]。

J. P. Das 等人提出的 PASS (planning,

attention, simultaneous processing and successive processing) 认知模型^[7], 即计划—注意—同时性加工—继时性加工模型, 是认知心理学的代表。PASS 模型把智力研究的焦点从特质分析转移到内部过程的探索上, 致力于从信息加工的角度刻画智力操作的心理机制。PASS 模型基于 Luria 提出的认知加工的三层机能单元模型^[8], 便于对认知过程的分析与评估, 由于其结构清晰的神经生理学理论基础, 可以弥补 BDI 模型的不足, 特别是能够加强对 Agent 认知过程的描述, 因此, 结合 PASS 模型, 笔者研究了 Agent 的认知过程, 提出了一种新的 Agent 理论模型 PASS-BDI。

2 PASS 模型的理论框架

PASS 模型认为人类的认知加工包括 3 个相互协调的机能单元, 每个机能单元都有对应于人脑的特定功能区^[7]。第一机能单元是人类心理过程的基础, 它维持了一种合适的唤醒状态。只有达到合适的唤醒状态, 个体才能接受和加工信息。唤醒状态为有意的注意行为提供了可能。唤醒是一种皮层警觉状态, 而注意则是一种更为复杂的认知活动。

[收稿日期] 2003-06-18; **修回日期** 2003-09-10

[基金项目] 国家自然科学基金资助项目 (60174021)

[作者简介] 樊 玮 (1968-), 男, 陕西乾县人, 博士, 中国民航学院副教授

若皮层唤醒状态适宜,就可能产生2种常见类型的注意——选择性注意和分配性注意。选择性注意的任务要求主体关注有关的刺激或对有关的刺激做出反应,而忽视无关刺激;分配性注意的任务是测量在不降低效率的情况下同时对不同活动进行操作的程度。

第二机能单元与个体接受、加工、维持来自外部世界的信息有关。可以将其功能划分为同时性加工和继时性加工过程2种类性。同时性加工包括将刺激整合成集合,或是对有共同特性的许多刺激进行再认。继时性加工往往涉及将刺激整合成特定的系列,使各成分形成一种链状结构。

第三个机能单元要求个体形成行动的计划,执行并证明计划的有效性。计划的产生、选择和执行是计划过程的3个主要方面。个体在设想出一个方法之前必须认识到寻找计划的必要性。如果个体的知识基础范围内没有合适的计划,那么就可能不会使用计划,或者有可能通过指导或通过发明去寻找或获得一个计划。如果产生的计划不止一个,则关键是选择一个最好的计划,并对这些方案的有效性加以监控。

所有的认知过程都是在知识基础这一背景中运行,知识基础包括通过正式和非正式的方式积累起来的信息。正式的知识是指通过教学和阅读获得的信息,非正式的知识是通过个人的经验获得的。3个机能单元彼此之间有一种动态联系,在这种动态联系中,它们对个体的经验做出反应,服从于发展的变化,并形成相互联系的系统。

3 Agent 的 PASS-BDI 模型

参照 PASS 认知模型构造了 Agent 的 PASS-BDI 模型。该模型可以表示为一个 10 元组。

定义 1 Agent = (ID, S, K, I, Input, Output, Communicator, Attention, Coder, Planner)

元组中各元素的含义如下:

ID: 确认 Agent 身份的唯一标识号;

S: Agent 的内部状态,包括其心智状态及对其他 Agent 的信任态度等;

K: Agent 所拥有的知识,包括事实、信念、模型和规则等,知识包括知识基础中的知识和 Agent 在交互过程中获得的临时性知识;

I: Agent 的意图,指的是 Agent 的行为目的和所需完成的任务等;

Input: Agent 用于监听外部世界的感知器;

Output: Agent 用于产生事件,影响外部环境的效应器;

Communicator: Agent 之间交换信息的通信器,是一个虚拟组件,由 Input 和 Output 的部分功能交织组合;

Attention: Agent 的第一功能区,是 Agent 核心对外的直接感官器官。对应于 PASS 模型第一功能区,将其命名为警觉器;

Coder: Agent 的编码器,对应于 PASS 模型第二功能区,完成对信息的同时性和继时性加工;

Planner: Agent 的规划器,对应于 PASS 模型第三功能区。相当于 Agent 的中央处理器。

上述定义基于这样一个基本思想,即软件 Agent 是一个具有特定能力的计算主体,它的主要功能通过规划器中定义的操作方法体现,但为了更好地完成任务,它需要进行信息的搜集、处理和再生,因此,软件 Agent 与外界的交互主要是信息的交流。所定义的 Agent 以 2 种方式与外界进行信息交流:一方面,通过感知器与效应器及其虚拟组成的通信器对与自身意图相关的信息进行搜集和传播,为实现自己的意图由编码器结合 Agent 自身的基础知识对所获取的信息进行加工和处理;另一方面,通过虚拟通信器与其他 Agent 互通有无,了解环境状况,搜集有关的更广泛的信息,从而调整内部状态、知识结构和意图等。

基于上述对软件 Agent 的定义可以确定软件 Agent 的内部体系结构,常见的结构分为慎思结构、反应结构和混合结构,笔者给出的结构属于混合结构。公共基础数据库构建于多 Agent 环境的公共平台上,为所有的 Agent 提供数据和知识支持,各 Agent 也可对公共基础数据库中的数据提出修改意见或者直接修改之。公共基础数据库由面向 OLAP 和 OLTP 的基础数据、模型库、规则库等组成。如图 1 所示,软件 Agent 的结构中除了具有自己独立的知识库、规则库、意图等基本属性外,还由以下 7 个分工不同的对象聚合而成:

1) 感知器 (Input) 负责事件的监测,将外界环境中的刺激转交警觉器 Attention。

2) 效应器 (Output) 负责在 Agent 操作完成后,将操作结果以事件的形式发布给系统中的相关部分,并期望影响外部环境。

3) 虚拟通信器 (Communicator) 是感知器

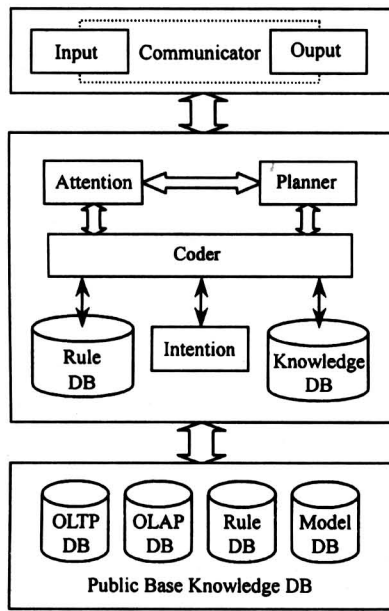


图 1 PASS-BDI Agent 内部结构图

Fig.1 The kernel structure of PASS-BDI Agent

和效应器的交叉组合, 仅负责 Agent 之间以消息协议的方式进行通信, 并将通信结果提交警觉器。

4) 警觉器 (Attention) 负责 Agent 对外界刺激的反应, 负责感知器、效应器和虚拟通信器的管理, 并完成对感知器提交信息的过滤和效应器反应形式的组织。警觉器同时提醒编码器对信息的进一步处理, 提醒规划器对 Agent 意图和目标的调整。

5) 编码器 (Coder) 完成 Agent 感知到的事件的处理, 按照意图和规划目标的要求进行同时性加工和继时性加工, 以辅助规划器调节 Agent 的目标和意图, 并完成规划器提交的数据和知识查询及计算任务。

6) 规划器 (Planner) 根据 Agent 的当前状态, 通过完成一定的操作来调整 Agent 的知识结构、意图和状态等, 完成 Agent 规划并指导警觉器和编码器完成相关任务的执行。

7) 意图管理器 (Intention) 负责管理和调整 Agent 的意图。

4 PASS-BDI 模型的形式化描述

4.1 PASS-BDI 模型心智状态

焦文品和史忠植等^[9, 10]将多价 π -演算^[11]引入到 Agent 心智状态与行为特征的描述中, 并给出了 Agent 的信念、目标和意图等的形式化描述, 可

以从静态特性和动态特性 2 个方面很好地刻画 Agent 的行为。采用 π -演算将 PASS-BDI Agent 的心智状态描述如下:

$$\text{AgentIntelligence} \stackrel{\text{def}}{=} \text{Belief} \mid \text{Goal} \mid \text{Intention} = \\ (\text{FactBaseBelief}(x) \mid \text{AttitudeBaseBelief}(\text{ID}, \tau, s)) \\ \mid \text{goal}_{\text{ID}}(x) \cdot \text{fact}_{\text{ID}}(x) \mid (\text{IntendID}(t, g) \cdot \overline{\text{time}}(t) \mid \\ \overline{\text{believe}}_{\text{ID}}(\text{ID}, t, g) \mid \text{goal}_{\text{ID}}(g)) \quad (1)$$

其中: $\text{FactBaseBelief}(x) \stackrel{\text{def}}{=} \text{fact}_{\text{ID}}(x) \cdot \text{Fact}_{\text{ID}}(x)$, 表示事实类信念;

$\text{AttitudeBaseBelief}(\text{nid}, \tau, s) \stackrel{\text{def}}{=} \text{time}(t) \cdot [t = \tau](\text{believes}_{\text{ID}}(\text{nid}, \tau t, s) \cdot \overline{\text{nid}}(t, s))$, 表示主观态度类信念。

4.2 PASS-BDI 模型认知过程的描述

4.2.1 第一机能单元——警觉器 警觉器在感知到外界的刺激或来自于内部规划器的监听要求时, 进入唤醒状态; 警觉器在一定的时间段内未监听到外界刺激时, 进入休眠状态, 以节省系统资源。

唤醒进程定义如下:

$$\text{StateArouse} \stackrel{\text{def}}{=} \text{input}_{\text{ID}}(t, x) \cdot ((v s) \\ (\overline{\text{state}}_{\text{ID}}(\tau t, g, s) \cdot \text{StateActive}(s))) \quad (2)$$

其含义为: Agent 在时刻 t 接收到输入器 input_{ID} 端口传送来的外界环境的刺激, 将根据自身的当前状态和目标决定一个新的状态 s , 并在端口 state_{ID} 上输出, 然后执行进程 StateActive , 改变或保持 Agent 当前状态。其中, 子进程 StateActive 定义为

$$\text{StateActive} \stackrel{\text{def}}{=} \text{perceive}_{\text{ID}}(s) \cdot ((v s) \cdot \\ ([s = \text{active}] \text{SetActive}_{\text{ID}} + [s \neq \text{active}] 0)) \quad (3)$$

其含义为: 进程在端口 $\text{perceive}_{\text{ID}}$ 接收到状态 s , 若为 active, 则执行进程 $\text{SetActive}_{\text{ID}}$ 唤醒进程; 否则, 什么也不执行, 相当于执行空进程 0。

休眠进程定义为

$$\text{StateSleeping} \stackrel{\text{def}}{=} ! \text{clock}(t) \cdot \\ ([t = \tau] \overline{\text{state}}_{\text{ID}}(\tau t, g, s) \cdot \text{StateSleep}(s)) \quad (4)$$

其含义为: 处于唤醒状态的 Agent 不断接受时钟端口的输出, 当在一段时间内 ($t = \tau$) 未接受到外界刺激或 Agent 内部的唤醒请求时, Agent 根据当前的目标 g 判定是否应该休眠, 并将判定结果 s 在端口 state_{ID} 输出, 然后执行子进程 StateSleep , 子进程 StateSleep 定义为

$$\text{StateSleep} \stackrel{\text{def}}{=} \text{perceive}_{\text{ID}}(s) \cdot ((v s) ([x = \\ \text{sleeping}] \text{SetSleeping}_{\text{ID}} + [x \neq \text{sleeping}] 0)) \quad (5)$$

其含义为: 进程在端口 $\text{perceive}_{\text{ID}}$ 接收到状态 s , 若为 sleeping, 则执行进程 SetSleeping 将进程置为休眠

状态；否则，什么也不执行，相当于执行空进程 0。

若 Agent 接收到来自规划器 (Planner) 的注意要求，则进入注意状态，注意状态又分为选择性注意和分配性注意。选择性注意进程定义为

$$\text{SelectiveAttention} \underline{\text{def}} \text{Planner}(t, x) \cdot ((\text{vs}) \overline{\text{stateID}}(\tau t, g, s) \cdot \text{StateSelectiveAttention}) \quad (6)$$

其含义为：进程在 t 时刻从端 Planner 接收到选择性注意请求 x ，则根据当前的目标 g 判定是否应该进入选择性注意状态，并将判定结果 s 在端口 stateID 输出，然后执行子进程 $\text{StateSelectiveAttention}$ ，该子进程定义为

$$\text{StateSelectiveAttention} \underline{\text{def}} (\text{vs}) \text{perceiveID}(s) \cdot ([s = \text{selectivestate}] \text{SetSelectiveState} + [s \neq \text{selectivestate}] 0) \quad (7)$$

其含义为：进程从端口 perceiveID 接收到其状态名字 s ，若 $s = \text{selectivestate}$ ，则将当前状态置为选择性注意状态；否则，结束进程 (0)。

分配性注意进程定义为

$$\text{DivededAttention} \underline{\text{def}} \text{Planner}(t, x) \cdot ((\text{vs}) \overline{\text{stateID}}(\tau t, g, s) \cdot \text{StateDivededAttention}) \quad (8)$$

其含义为：进程在 t 时刻从端口 Planner 接收到分配性注意请求 x ，则根据当前的目标 g 判定是否应该进入分配性注意状态，并将判定结果 s 在端口 stateID 输出，然后执行子进程 $\text{StateDivededAttention}$ ，该子进程定义为

$$\text{StateDivededAttention} \underline{\text{def}} (\text{vs}) \text{perceiveID}(s) \cdot ([s = \text{dividedstate}] \text{SetDivededState} + [s \neq \text{dividedstate}] 0) \quad (9)$$

其含义为：进程从端口 perceive 接收到其状态名字 s ，若 $s = \text{dividedstate}$ ，则将当前状态置为分配性注意状态；否则，结束进程 (0)。

警觉器进程由唤醒子进程、休眠子进程、选择性注意子进程、分配性注意子进程等组成，综合式 (2) 至式 (9)，可以给出警觉器进程为

$$\text{Attention} \underline{\text{def}} \text{StateArouse} \mid \text{StateSleeping} \mid \text{StateSelectiveAttention} \mid \text{StateDivededAttention} \quad (10)$$

4.2.2 第二机能单元——编码器 编码器的任务就是进行同时性加工和继时性加工，并将接受到的外界刺激转换为 Agent 内部的知识，也可建议修改公共基础数据库或直接修改公共基础数据库。编码器将对外界请求或自身目标完成的结果告知规划器，并同时经效应器输出。

同时性加工要求按照目标的要求，对有共同特性的许多目标进行再认，并对刺激过滤结果进行概括处理。同时性加工进程定义为

$$\text{SimuProc} \underline{\text{def}} (\text{vi}) \text{perceiveID}(t, g) \cdot (\text{InputID}(x) \mid \text{FindID}(g, i)) \cdot ([i \neq \text{process}] 0 + [i = \text{process}] (\text{Proc1ID}(x_1) \mid \text{Proc2ID}(x_2) \mid \dots \mid \text{ProcnID}(x_n)) \mid \overline{\text{GeneProcID}}) \quad (11)$$

其含义为：进程在时刻 t 从端口 perceiveID 接收到其目标 g ，则开始监听端口 InputID 的外界刺激 x ，并同时经子进程 $\text{FindID}(g, i)$ 查询 Agent 的基本知识与意图 i ，判定 Agent 是否可以完成该目标。紧接着执行下面的复合进程：若该 Agent 不能完成该目标 ($i \neq \text{process}$)，则结束进程 (执行进程 0)；否则，并行执行多个子进程 $\text{ProckID}(x_k)$ ， x_k 为各子进程按照自己的功能要求对刺激 x 进行过滤后的结果。各子进程 $\text{ProckID}(x_k)$ 的结果最后经子进程 GeneProcID 汇总输出。

继时性加工要求将刺激整合成特定序列，各序列子元之间的输入输出相互衔接。继时性加工进程定义为

$$\text{SuccProc} \underline{\text{def}} (\text{vi}) \text{perceiveID}(t, g) \cdot (\text{InputID}(x) \mid \text{FindID}(g, i)) \cdot ([i \neq \text{process}] 0 + [i = \text{Process}] (y(x) \cdot \text{Proc1ID} \cdot \bar{y}(x) \mid \dots \mid y(x_{n-1}) \cdot \text{ProcnID} \cdot \bar{y}(x_n))) \quad (12)$$

其含义为：进程在时刻 t 从端口 perceiveID 接收到目标 g ，则开始监听端口 InputID 的外界刺激 x ，并同时经子进程 $\text{FindID}(g, i)$ 查询 Agent 的基本知识与意图 i ，判定 Agent 是否可以完成该目标。紧接着执行下面的复合进程：若该 Agent 不能完成该目标 ($i \neq \text{Process}$)，则结束进程 (执行进程 0)；否则，串行执行多个子进程 $\text{ProckID}(x_k)$ ，每个子进程 $\text{ProckID}(x_k)$ 都首先在端口 y 上接收上一个子进程的输出 x_{k-1} ，然后进行处理，最后在端口 y 输出其处理结果 x_k ，作为下一个子进程 $\text{ProckID}(x_k)$ ($l = k + 1$) 的输入。

综合式 (11) 和式 (12) 给出编码器进程的定义为

$$\text{Coder} \underline{\text{def}} \text{SimuProc} \mid \text{SuccProc} \quad (13)$$

4.2.3 第三机能单元——规划器 规划器产生了一个新的目标之后，如果实现该目标的方法是明显的，那么，Agent 就会在它的知识基础中寻找一种方法。如果这种方法超出其知识基础，就需要产生行动规划并检验这个规划，若该规划可行，就执

行; 否则, 产生新的规划。规划完成后, 需判定该规划并修正基础知识。

规划产生进程定义为

$$\begin{aligned} \text{PlanProduce} \underline{\text{def}} & (vy) \text{goal}_{ID}(g) \cdot \\ & (\text{Find}_{ID}(\text{knowledge}, i) \cdot \bar{x}(y)) \cdot \\ & ([y = \text{finded}] \text{PlanExcute} + [y \neq \text{finded}] \text{NewPlan}) \end{aligned} \quad (14)$$

其含义为: 进程从端口 goal_{ID} 接收目标 g , 紧接着查找其知识基础 knowledge 并参照其意图 i 寻找实现目标的方法, 寻找结果在端口 x 输出。若找到了实现目标的方法 ($y = \text{finded}$), 则执行进程 PlanExcute 完成规划; 否则 ($y \neq \text{finded}$), 执行子进程 NewPlan 以产生规划。子进程 NewPlan 定义为

$$\begin{aligned} \text{NewPlan} \underline{\text{def}} & \overline{\text{Communicat}}(t, g) | \\ & (\text{Wait}_{ID}(\tau t, \text{plans}) \cdot \bar{x}(\text{plans}))! \text{PlanSelect} \end{aligned} \quad (15)$$

其含义为: 进程在时刻 t 自端口 Communicat_{ID} 输出其目标 g , 请求其他 Agent 的协作, 并启动等待进程 Wait_{ID} 接收其他 Agent 帮助产生的规划 plans , τt 表示一定的时延, 接收到的规划可能有 0 个或多个。等待进程在端口 x 输出其接收到的规划, 并执行规划选择进程 PlanSelect 。

规划选择进程定义为

$$\begin{aligned} \text{PlanSelect} \underline{\text{def}} & (vy) (\text{perceive}_{ID}(\text{plans}) \cdot \\ & (\text{Check}_{ID}(\text{knowledge}, i) \cdot \bar{x}(y)) \cdot \\ & ([y = \emptyset] \text{PlanProduce} + [y \neq \emptyset] (\text{Filter} | \text{PlanExcute})) \end{aligned} \quad (16)$$

其含义为: 进程在端口 perceive_{ID} 接收到规划 plans , 根据其基础知识 knowledge 和意图 i 用子进程 Check_{ID} 检查该规划, 并在端口 x 输出其检查结果。若检查结果为空 ($y = \emptyset$), 则表示该 Agent 对目标 g 无知, 启动进程 PlanProduce 重新产生规划, 否则 ($y \neq \emptyset$) 过滤 (Filter 子进程) 并执行特定规划 (PlanExcute)。

规划执行进程定义为

$$\begin{aligned} \text{PlanExcute} \underline{\text{def}} & (vy) (\text{perceive}_{ID}(\text{plans}) \cdot \\ & (\text{Excute}_{ID} \cdot \bar{x}(y)) \cdot ((\text{Evaluate}(\text{plan}, y) \cdot \\ & \bar{x}(z)) | \text{Modify}(\text{knowledge}, i) | ([y = \emptyset] | 0 + \\ & [y \neq \emptyset] \text{PlanProduce} \end{aligned} \quad (17)$$

其含义为: 进程在端口 perceive_{ID} 接收到规划 plans , 用子进程 Excute_{ID} 执行相关任务, 在端口 x 输出其执行结果, 并根据需要用子进程 Modify 修正其基础知识 knowledge 和意图 i , 同时判定, 若规划执

行完毕且无更好的规划 ($y = \emptyset$), 则结束进程 (0); 否则 ($y \neq \emptyset$), 执行进程 PlanProduce 产生新一轮的规划。

由式 (14) 至式 (17) 的定义可见, 规划器是一个递归进程。规划器进程定义为

$$\text{Planner} \underline{\text{def}} \text{PlanProduce} | \text{PlanSelect} | \text{PlanExcute} \quad (18)$$

综上所述, 可以将 PASS-BDI 模型的认知过程描述为

$$\text{AgentCognitive} \underline{\text{def}} \text{Attention} | \text{Coder} | \text{Planner} \quad (19)$$

即其认知过程由警觉器式 (10)、编码器式 (13)、规划器式 (18) 共同组成。

4.3 PASS-BDI 模型整体行为

Agent 的整体行为主要包括 Agent 的行为规范、Agent 的自适应能力、Agent 的接口等, 至于多 Agent 环境下, Agent 的联合意图及协作规划等, 将另文讨论。下面的描述省去了对有关表达式含义的详细描述。

Agent 行为规范 Agent 行为规范一方面用来规定当 Agent 要达到某种目标时应该或将要采取的行动, 相当于其规划, 另一方面用来约束当 Agent 处于某种状态 (或感知到某种刺激) 时的行为模式。当动作完成时, Agent 进入预期的状态。行为规范进程定义为

$$\begin{aligned} \text{ActRule} \underline{\text{def}} & \text{Planner}(g) | (\text{perceive}_{ID}(t, p) \cdot \\ & \overline{\text{Intend}_{ID}}(t, g) | \text{Planner}(g)) \end{aligned} \quad (20)$$

Agent 的自适应能力: Agent 的自适应能力主要表现为知识及能力的自我更新、目标的自我调整等。

知识更新进程包括新知识更新、外界刺激引起的知识更新和知识的反省等, 分别定义如下:

Agent 在接收到新的知识后, 会按照自己的知识和意图检查这些新知识, 若新知识符合自己的意图, 并和原有的知识一致, 则更新自己的知识库; 否则, 丢弃新知识。新知识更新定义为

$$\begin{aligned} \text{ReceiveKnowledge} \underline{\text{def}} & \text{receive}_{ID}(\text{nid}) \cdot \\ & \overline{\text{nid}} \cdot \text{Check}_{ID}(\text{knowledge}, \text{rule}, \text{Intention}) \cdot \\ & (v \text{result}) ([\text{result} = \text{consisting}] \text{UpdateKnowledge} + \\ & [\text{result} \neq \text{Consisting}] 0) \end{aligned} \quad (21)$$

Agent 在感知到外界的刺激后, 同样需要检查自己的知识、规则、意图等, 判定自己的感知历史中是否包含对类似刺激的反应能力, 若不包含, 则

需要更新自己的知识。对刺激的更新定义为

$$\begin{aligned} \text{InsKnowledge} \text{ def } & \text{perceive}_{\text{ID}}(x) \cdot \\ & \text{Check}_{\text{ID}}(\text{knowledge}, \text{rule}, \text{Intention}) \cdot \\ & (\nu \text{ result})([\text{result} = \text{consisting}] \text{UpdateKnowledge} + \\ & [\text{result} \neq \text{Consisting}] 0) \end{aligned} \quad (22)$$

Agent 在经历了一定的生存周期后, 其经验和知识都有了长足的进展, 此时, 需要对自己的知识重新进行归类和整理, 也可以称为历史经验学习, 这一过程可以定义为

$$\text{ExperienceLearn} \text{ def } [t = \tau] \cdot \text{Check}_{\text{ID}}(\text{knowledge}, \text{rule}, \text{Intention}) | \text{UpdateKnowledge} \quad (23)$$

综合式 (21)、式 (22) 和式 (23), 可将 Agent 的知识更新进程定义为

$$\text{Knowledge} \text{ def } \text{ReceiveKnowledge} | \text{InsKnowledge} | \text{ExperienceLearn} \quad (24)$$

Agent 可以通过 2 种途径来更新其能力: a. 直接向外界发出请求, 外界则返回特定的服务或知识。b. Agent 可通过自我反省, 进行能力的更新。

请求外界帮助的能力更新过程定义为

$$\text{RepCap} \text{ def } (\text{Req}(w, x) | \text{ReqKnowledge}(w, c)) \cdot \text{UpdateCap} \quad (25)$$

自我反省的能力更新过程定义为

$$\text{RenewCap} \text{ def } [x = \text{req}]((\overline{\text{Fact}}_{\text{ID}}(x) | \text{FactBaseBelief}(x)) \cdot \text{UpdateCap}(x)) \quad (26)$$

综合式 (25) 和式 (26), 可将 Agent 能力的更新进程定义为

$$\text{Capability} \text{ def } \text{RepCap} | \text{RenewCap} \quad (27)$$

目标的调整: Agent 既可能出于自身的需要, 也可能出于外界的需求, 要对自己的目标进行调整。当 Agent 感知到外界的某种需求时, 判断是否有能力满足该需求, 如果它所掌握的知识及能力足以实现该需求所蕴含的目标, 则将该目标作为自己的新目标之一。

$$\text{RenewGoal} \text{ def } [x = g]((\overline{\text{Fact}}_{\text{ID}}(x) | \text{FactBaseBelief}(x)) \cdot \text{UpdateGoal}(x)) \quad (28)$$

综合式 (24)、式 (27)、式 (28), 形成 Agent 的自适应能力如下:

$$\text{Adapt} \text{ def } \text{Knowledge} | \text{Capability} | \text{RenewGoal} \quad (29)$$

接口特性: Agent 的接口特性包括: a. Agent 向外界提供的服务; b. Agent 要实现自己的目标需要外界提供的服务; c. Agent 与外界进行通信的通道, 这些通道不仅是传输服务的媒介, 也是接收外界信息或

向外界发送信息的端口, 在 PASS-BDI 模型中对应于其感知器、效应器和虚拟通信器。

Agent 向外界提供的服务定义为

$$\text{AgentServers} \text{ def } ! (S_{\text{ID}}(\text{askAgent}_{\text{ID}}, y) \cdot ((\overline{x}(\text{sid}) \cdot \text{sid}) | \overline{\text{Fact}}_{\text{ID}}(\text{sid}))) \quad (30)$$

Agent 向外界请求的服务定义为

$$\text{Agent Req} \text{ def } \overline{R}(w, r) \cdot w(\text{nid}) \cdot \overline{\text{nid}} \cdot \text{ExcuteNext} \quad (31)$$

请求服务需申明哪儿请求和请求什么, 请求的结果同样是以函数指针形式返回的服务或者以指针形式返回的知识。

Agent 与外界进行的通信依赖 Agent 的警觉器完成, 关于警觉器的定义请参见式 (10)。

综合式 (10)、式 (30) 和式 (31), 可将 Agent 的接口特性定义为

$$\text{AgentInterface} \text{ def } \text{AgentServes} | \text{Agent Req} | \text{Attention} \quad (32)$$

综合式 (20)、式 (29) 和式 (32), 可将 PASS-BDI 模型的整体行为定义为

$$\text{AgentActionRule} \text{ def } \text{ActRule} | \text{Adapt} | \text{AgentInterface} \quad (33)$$

即 PASS-BDI 模型的整体行为由其行为规范、自适应能力、接口特性等共同决定。

5 PASS-BDI Agent 在 DAI 中的应用

DAI 系统中的分布式协作是 MAS 技术研究的热点和难点之一, 基于组织结构的 MAS 协作已成为这一研究领域的一个重要分支, 然而, 现有的研究大都集中于理论模型的探讨, 从实现技术上讲, 稍显欠缺。应用多价 π -演算的形式化方法, 可以对系统进行精确的且无二义性的描述, 能比较方便地转换为实际的程序, 并能在此基础上进行推理, 以验证其正确性和合理性。下面首先给出 MAS 中环境和组织的定义, 并在此基础上给出组织活动中单个 Agent 的行为过程。

定义 2 MAS 所处环境 $E = (A, C, G, R, K)$, 其中 A 为 Agent 的集合, $A = \{\text{Agent}_1, \text{Agent}_2, \dots, \text{Agent}_n\}$, n 为非负整数; C 为 A 中各个 Agent 能力的并集, $C = \bigcup_{i=1}^n \text{Agent}_{i,c}$; G 为 A 中各个 Agent 目标的并集, $G = \bigcup_{i=1}^n \text{Agent}_{i,g}$; R 为能力、目标关系, 表示能力对目标的承担关系, $R \subseteq 2^C G$, 2^C 表示 C 的幂集; K 为基础知识库。

定义3 MAS的组织定义为 $ORG = (E, M, O-G, K_0)$, 其中, E 为其所处的环境; $M \subseteq A$ 为组织成员的集合; $O-G \subseteq G$ 为组织的目标集合; $K_0 \subseteq K$ 为组织的初始知识集。

焦文品研究了 MAS 中组织结构的形成、角色与目标的关系以及组织目标的分解^[12], 在此基础上, 笔者给出 PASS-BDI 结构的 Agent 的行为过程如下:

$$\begin{aligned} & \text{AgentActive} \stackrel{\text{def}}{=} \overline{\text{Planner}_{ID} s} \cdot \text{StateArouse} \cdot \\ & (\text{SelectiveAttention} \mid \text{DividedAttention}) \cdot \text{Input}(x) \cdot \\ & (\text{PlanProduce}(c, g, x) \cdot \overline{\text{PlanSelect}(c, g, x)} r \cdot \\ & \quad ([r \neq \phi](\text{SimuProc} \mid \text{SuccProcc}) \mid \cdot \\ & \quad [r = \phi](\text{AgentInterface} \mid \text{Adapt}) \cdot \text{AgentActive})) \cdot \\ & \quad (\nu t) \overline{\text{Output}_{ID}}(t, b) \quad (34) \end{aligned}$$

组织中的个体为了完成某个目标, 执行如下的行为过程: a. 在其规划器 Planner_{ID} 端口输出期望的活动状态 s ; b. 由唤醒进程 StateArouse 将该 Agent 置于选择性注意 ($\text{SelectiveAttention}$) 或者分配性注意 (DividedAttention) 状态; c. 在感知器 Input 端口获得对外界环境 E 的认识 x , 启动规划器按照 Agent 的能力 c , 目标 g 和 x 产生规划 (PlanProduce); d. 选择规划并在其进程 PlanSelect 端口输出选择结果 r 。若能产生有效规划, 则启动编码器进行同时性加工 (SimuProc) 或者继时性加工 (SuccProcc); 否则, 依赖 Agent 的接口特性 (AgentInterface) 进行自适应调整 (Adapt), 修改环境和组织的能力、目标及其关系, 并递归执行行为进程 AgentActive ; e. 经过执行时延, 在端口 Output_{ID} 输出执行结果 b 。

与相关研究结果比较, 式 (34) 给出了组织中 Agent 行为的清晰结构, 并在其中强调了认知结构。基于这一结果, 一方面可以开展程序的性质验证工作, 另一方面也极大地方便了面向 Agent 的编程工作。

6 结论

PASS-BDI 模型以警觉器、编码器、规划器为核心, 在 Agent 模型理论和具体的 Agent 结构之间建立了明确的映射关系, 使 Agent 的建造者易于编程实现 Agent 的理论抽象模型。同时, 用多价 π -演算严格刻画了 PASS-BDI 模型的心智状态、

认知过程、整体行为等, 重点加强了对 Agent 认知过程的描述, 在认知过程中增加了对 Agent 无知 (式 (16)) 及其处理方法的描述, 扩展了 Agent 理论研究的内涵。最后, 用该模型研究了组织结构中 Agent 的行为。进一步的工作包括: 研究 PASS-BDI 模型下 Agent 的理性平衡问题; 研究多 Agent 的协作及 Agent 的组织结构与组织形成; 研究 PASS-BDI 模型下多 Agent 系统的体系结构问题等。

参考文献

- [1] Bratman M E. Intentions, Plans, and Practical Reason [M]. Cambridge MA. Harvard Univ Press, 1987
- [2] Moor R. A Formal Theory of Knowledge and Action [M]. Formalizing Common Sense, Ablex Publishing Corporation, 1990. 319~358
- [3] Cohen P R, Levesque H J. Intention is choice with commitment [J]. Artificial Intelligence, 1990, 42: 213~261
- [4] Rao A S, Georgeff M P. BDI agents: from theory to practice [A]. First International Conference on Multi-agent Systems (ICMAS-95) [C]. San Francisco, CA, 1995
- [5] 刘贵全, 陈小平, 范焱, 等. 多主体协作系统的一种形式模型 [J]. 计算机学报, 2001, 24(5): 529~535
- [6] 史忠植. 智能主体及其应用 [M]. 北京: 科学出版社, 2000
- [7] Das J P. Assessment of Cognitive Processes [M]. New York: Plenum, 1999
- [8] Luria A. Higher Cortical Functions in Man (2nd ed) [M]. New York: Basic Books, 1980
- [9] 焦文品, 史忠植. 构造 MAS 的动态体系结构的模型 [J]. 计算机学报. 2000, 23(7): 732~737
- [10] Jiao Wenpin, Shi Zhongzhi. Formalizing agent's attitudes with the polyadic π -calculus [A]. Proceeding of 4th Workshop on Practical Reason and Rationality [C]. Stockholm, Sweden, 1999, 21~27
- [11] Engelfriet J. Multiset and structural congruence of the pi-calculus with replication [J]. Theoretical Computer Science, 1999, 211: 311~337
- [12] 焦文品, 史忠植. 多主体间的协作过程研究 [J]. 计算机研究与发展. 2000, 37 (8): 904~911

(下转第 56 页)

- [7] Weimann A. Inverting a microwave backscattering model by the use of a neural network for the estimation of soil moisture [A]. Geoscience and Remote Sensing Symposium Proceedings, IGARSS '98 [C]. 1998 IEEE International, Jul 1998, 4 (6-10): 1837~1839
- [8] Wang Xin, Li Z Q. Numerical calculations of emission from rough soil surfaces and inversion of soil moisture by a neural network approach [A]. 2001 International Laster, lightwave and Microwave Conference Proceedings [C]. Shanghai China, 2001. 202~205
- [9] Liu Ning, Li Zongqian. Bi-spectrum scattering model for dielectric randomly rough surface [J]. Tsinghua Science and Technology, 2003, 8 (5): 46~52
- [10] 张德海, 张俊荣, 张升伟, 等. 双模态微波遥感器的研制 [J]. 遥感学报, 1999, 3 (2): 103~106
- [11] Bindlish R, Barros A P. Subpixel Variability of Remotely Sensed Soil Moisture: An Inter-Comparison Study of SAR and ESTAR [J]. IEEE Transactions on Geoscience and Remote Sensing, 2002, 40 (2): 1917~1920

Analysis of Data Sets With Different Microwave Remote Sensing Mode in Soil Moisture Retrieval

Yuan Wei, Li Zongqian, Liu Ning

(Department of Electronic Engineering, Tsinghua University, Beijing 100084, China)

[Abstract] The scattering coefficient and emissivity are both very sensitive to the moisture content of bare soil surface, which is the essence of measuring soil moisture by using microwave technology. The methods and characters of three microwave remote sensing modes for soil moisture retrieval are discussed in this paper. Firstly, scattering coefficient and emissivity for bare soil surface are calculated based on the theoretical backscattering model BSM. Then the paper analyzes the distribution of isomoisture regions of different data sets, and presents that soil moisture information can be retrieved by using ANN with appropriate input data set.

[Key words] microwave remote sensing; soil moisture retrieval; remote sensing modes; neural network

(cont. from p.49)

PASS - BDI Model for Software Agent

Fan Wei^{1, 2}, Chen Zengqiang², Yuan Zhuzhi²

(1. Software Base, Civil Aviation University Of China, Tianjin 300300, China;

2. College of Information Technology and Science, Nankai University, Tianjin 300071, China)

[Abstract] Recent research on software agent is mainly based on rational agent theories that have been presented by Bratman and its core is to build BDI models for agent. But the models can not present the active cognitive processes of agent, and it is hard to richly present the relations between agent problem solving and agent mental states. Because it is not easy to build the explicit corresponding relations between the theory model and the model structure, agent rational models are difficult to realize. This paper introduces a psychologically recognized model - PASS (planning, attention, simultaneous processing and successive processing) into the study about intelligent agent, builds a new agent model named as PASS - BDI, describes the mental states, cognitive processes and whole behaviors with pi-calculus at length and strengthens the active cognitive attributes of agent. Because having built the explicit corresponding relations between this theory model and the model structure, it is easy to program in AOP practice. An application of the model in MAS is presented at last.

[Key words] agent; pi-calculus; cognitive processes; mental state