

智能主体 人工智能 Agent 软件智能主体 (9)  
心态 行为

76-78

# 基于心态和行为规则的软件 agent

Software Agent Based Mental State and Behavior Rule

李保建<sup>1</sup> 曾广周<sup>2</sup> 林宗楷<sup>1</sup> TP18

(中国科学院计算技术研究所 CAD 开放实验室 北京100080)<sup>1</sup> (山东工业大学计算机系)<sup>2</sup>

**摘要** This paper presents a new software agent model. In this model, mental categories such as beliefs intentions and commitments are adopted, and the idea of behavior rules and meta behavior rules is introduced. Therefore agents constructed form this model have both intentional and simulation-response characteristics.

**关键词** software agent, mental state, behavior rule

## 1 引言

软件 agent 是 agent 研究的重要分支。尽管对什么是软件 agent, 不同的研究者会对此有不同的回答, 但它们至少是用某种计算机语言编制的软件。例如, Genesereth<sup>[2]</sup>等持如下观点: 一个实体当且仅当其能用诸如 ACL 之类的 agent 通讯语言进行正确地通讯才称为软件 agent。其研究动机源于如何解决丰富多样的软件产品之间的交互作用问题。

软件 agent 研究中面临的重要问题是 agent 概念的实用化问题。研究中存在两种倾向, 一种是理论上的有益的探索, 但其概念远离实现甚至无法实现; 另一种则是 agent 概念的滥用, 一个计算机程序可以随便地被贴上所谓 agent 的标签, 使其仅仅成为一个时髦的名词。

Yoav Shoham<sup>[3]</sup>在 agent 的研究中独树一帜。他明确地给出了一种 agent 概念, 使 agent 程序与通常传统的计算机程序有了鲜明的区别, 并首次提出了面向 agent 程序设计的思想。但是, 正如本文下面所指出的, 尽管他们根据其思想设计了一种面向 agent 的程设语言 AGENT0, 其工作仅仅是实验性的, 本文借鉴其思想提出了一种基于心态和行为规则的软件 agent 模型。最后给出我们的模型的实验技术。

## 2 Shoham 的 agent 理论

Shoham 提出了一种面向 agent 的程序设计范型(AOP)<sup>[3]</sup>。在这种范型中, 多 agent 相互作用从而促成了一种计算的社观。然而在文[3]中, 其主要精力放在了单个 agent 的设计。Shoham 给出定义: 如果一个实体可以被视为包含了诸如信念、能力、选择、承诺等心态, 这个实体便可以称为 agent。这些精

神状态与其通常概念大体一致, 并具有精确定义。采用这种定义, 什么是 agent 便成为哪些实体可以被看成具有心态的问题了。答案是任何东西都可以用心态来描述, 但是这种高层描述并不总是有益的, 因此应该有是否可以进行这种描述的规则。显然, 我们不能随意地给机器的这一部件贴上标签“信念”, 而给另一部件标以“承诺”。Shoham 提出了进行这种描述的三个条件:

①一个关于特定的心态范畴的精确理论; 这个理论应该有精确的语义, 并且每个范畴应与其通常概念大体一致。

②机器的各个部件应遵循这个理论。

③这种形式化理论在分析或者设计机器的时候应起到非平凡的作用。

基于这种考虑, Shoham 给出了关于各种心态范畴及其性质的定义。当然, 没有一个唯一正确的关于心态范畴的选择, 也没有关于它们的唯一正确理论, 因为不同的应用可能要求不同的心态范畴及性质。

支持 Shoham 关于其心态范畴选择的是如下对世界的非正式的观点: 在任何时间点, 未来取决于两个因素; 一个是过去的历史, 一个是 agent 的当前行动。而 agent 的行动取决于它的决定或者选择。决定则受到 agent 的信念的逻辑约束。此外, 信念涉及到世界过去、现在及未来的状态, 其它 agent 的心态, 这个及其它 agent 的能力。这种观点导致了以下两个基本心态范畴: 信念和决定(或者选择)的引入, 以及第三个从本质上来说不属于心态的范畴: 能力。也许是为了更具普遍性的缘故, Shoham 将决定这个范畴替换为承诺, 而决定则成了 agent 对自己的承诺。

Shoham 设计了用来描述这些范畴的简单时间点语言 (simple point-based temporal language)。在这个语言中, 时间对所有的心态范畴都是基本的。譬如用这种语言描述信念的一个例子:  $B_3^3 B_6^{10} \text{like}(a, b)^7$  的含义为在时间 3 agent a 相信, 在时间 10 agent b 会相信 a 在时间 7 喜欢 b。

基于这种心态范畴的选择以及描述这些心态范畴的简单时间点语言, Shoham 设计了一个通用 agent 解释器。被解释执行的 agent 程序的角色是控制 agent 心态的演化, 而行为则是 agent 承诺的某一时刻行动 (当其执行时间到以后) 的副作用。因此, 原则上来说, agent 的行为是相当简单的。每个 agent 以固定的时间间隔重复下面两个步骤:

1) 读入当前时刻的消息, 更新其心态, 包括其信念和承诺 (agent 程序负责这种更新)。

2) 执行当前时刻的承诺, 可能会引起进一步的信念变化 (这个阶段不依赖于 agent 程序)。

agent 可以承诺的行动包括基于言语行为的通讯行动诸如通知和请求等, 也包括任意的私有行动。agent 赖以作出承诺的原则称为承诺规则。在承诺规则中 agent 可以作出承诺的条件包括心态条件以及当前消息条件。而 agent 程序则简单地由其能力、初始信念及一系列的承诺规则组成。当然, 时间粒度也要由程序来决定。

Shoham 是最早提出面向 agent 程序设计概念的人, 而他的关于 agent 的理论也给人耳目一新的感觉。但我们认为其理论似有以下不足: 1) 信念、承诺、动作等之中的简单时间项, 给人一种不真实、不自然的感觉。程序设计人员或许觉得很习惯于这种方式的程序设计。2) 时间粒度的选择问题, 时间粒度显然不能选得过小, 不然有可能使 agent 在此时间间隔内无法完成上述的两个步骤, 这将导致 agent 的逻辑混乱; 时间粒度选得过大, 显然将会造成时间浪费, 同时也带来一些不便, agent 只能在这些大粒度的离散时间点上进行交互。

### 3 基于心态和行为规则的 agent

现在我们着手建造自己的 agent。我们首先给出 agent 的定义。

**定义 1** agent 是一个七元组: (mm, ms, br, mbr, bra, map, cap)。其中, mm 是消息监听器, ms 表示心态, br 表示行为规则, mbr 表示行为元规则, bra 是行为规则调整器, map 是消息映射器, cap 则表示能力。

**定义 2** agent 的心态由信念、意向和承诺组成。

信念、意向和承诺等心态范畴可以通过定义一种模态语言来精确地描述, 我们将另外撰文介绍。本文只是给出其非形式描述。

**定义 3** 信念由 agent 认为真的事实和规则组成。

我们所采用的信念与人们关于信念的通常概念大体一致, 与 Shoham 的信念概念不同之处有二: (1) 我们去掉了信念中的简单时间点。在约定 (default) 的情况下, 指的是 agent 的当前信念, 如果指过去或将来的信念, 应该用时态描述符明确指明。采有这种约定的价值在于当前信念对于 agent 采取什么行动起决定作用。(2) 我们在信念中除了保留最重要的事实性知识外, 增加了相关的规则性知识。

**定义 4** 意向是一个 agent 对其它某个或某些 agent 在某种心态条件下做某种或某些动作的意愿。

通常 agent 可以将其意向通知给其它 agent, 其它 agent 可以根据自己的当前心态来决定是否请求相应的动作。

**定义 5** 承诺是一个 agent 对其它某个或某些 agent 在一定心态条件下在将来某个时间做某个或某些动作的表示。一旦承诺的时间条件得到满足, agent 将检查那时它的心态条件以决定是否执行其承诺的动作。

我们的承诺概念和 Shoham 的承诺概念基本相同, 只是 Shoham 的承诺是基于简单时间点的, 而我们的承诺的时间条件用时态描述符表示, 更广泛一点, 譬如, 可以是“一旦有空闲”、“半个小时以后”等等。

**定义 6** 行为规则是三元组 (消息条件, 心态条件, 动作序列) 的集合。每一规则的语义为: 如果 agent 收到消息条件中规定的消息, 并且具有心态条件中规定的心态, 则 agent 将执行相应的动作序列。

值得注意的是, 这里的心态条件 (譬如某个特别的信念) 未必是 agent 的心态库中现成的, agent 可以利用其规则型信念进行简单的推理, 来判断某个特别的信念是否成立。

**定义 7** 行为元规则是 agent 依据某些行为规则的应用所产生的心态效应而对这些行为规则进行修正的规则。

**定义 8** 能力是 agent 可以进行的通讯和私有动作的集合。私有动作是个体 agent 所独有的, 而通讯动作对所有的 agent 都是相同的。通讯动作一般应基于言语行为理论分类为通知、请求、命令等。

对于 agent 各元素之间的关系, 我们可以用图 1

直观地表示。

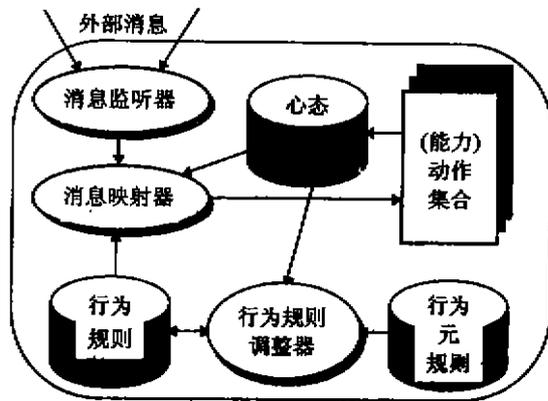


图1 agent基本模型

agent各个部分的作用及其agent的工作过程可以简述如下:消息监听器的作用在于时刻主动地监听发向agent的各种消息,包括来自其它agent的基于言语行为的消息以及来自用户事件和系统事件(如定时器事件)的消息,从而产生使agent发出某种行为的消息条件。而心态则产生使agent发出某种行为的心态条件。agent的心态是从其初始状态出发随着其持续不断的运行而不断地演化的。agent的行为规则是其根据其心态条件和消息条件发出某种行为的依据。一般来说,agent的心态是不稳定的、易变的,而其行为规则则是相对稳定的。但相对稳定并不意味着不可变,在一定的条件下,agent的行为规则会作出调整,而调整的依据则是其行为元规则。行为规则调整器依据行为元规则监视需要进行某条行为规则调整的心态条件,并进行相应的调整。消息映射器则是agent根据消息条件、心态条件和其行为规则产生相应的行动的转换机制。agent的能力是其可以进行的动作的集合,在未引进计算反射的条件下,agent的能力是不可变的。

#### 4 实现技术

这里我们讨论agent的基于Windows NT的Win32环境子系统或Windows 95及其所支持的网络的实现技术。我们在图1中给出的agent基本模型是一个面向实现的模型。

agent用消息监听器感知外界(包括用户、系统及其它agent)的消息,agent应能对其外界消息持续不断地接受,因此可以用一个持续运行的线程来实现。

行为规则调整器用于在一定条件下(譬如说agent相信自己受到了欺骗)调整其行为规则,什么时间进行调整完全是随机的。因此可以用另一个持续运行的线程来实现。

消息映射器和其激活的动作的执行则有不同的实现方案。一种方案是消息映射和其激活的动作顺序执行,并用一个线程实现:这样,当agent执行其动作的时候,消息映射功能只能暂停,一直等到动作执行完毕才能进行新的映射。另一种实现方案是消息映射器单独用一个线程实现,但它用创建新线程的方法来执行相应的动作,随着动作的执行完毕,其所创建的线程也随之而终止。第二种方案的优越之处在于agent可以同时不同的动作或服务。另外要注意的是,由于agent的行为规则中的心态条件也可以通过简单的推理得到的,消息映射器也应包含相应的推理机制。

**结束语** 本文借鉴Shoham关于agent的一些观点,提出了一种新的基于心态和行为规则的agent模型。由于在我们的agent模型中,既保留了心态的概念,又引进了行为规则和行为元规则,从而使我们的agent具有如下的特点:

- 行为规则的引入使其既具有意向型agent的特点,又具有一定的刺激反应特性。可以使agent之间更好地进行从低到高的各种层次的合作。

- 在心态中简单时间点的去除以及信念、意向和承诺等心态范畴的选择,是对人的心态的更好的模拟。

- 行为元规则的引入使得agent具有可以修正自己的行为规则以适应新情况的一定自适应特性。

- 本模型具有面向实现的特点,可以利用本模型建立实用的agent系统。

更进一步的工作应该对心态范畴及其表示方法予以更精细探讨,并研制agent开发环境。

#### 参考文献

- [1] K. Decker, Distributed problem-solving techniques: A survey, IEEE Trans. on Systems, Man and Cybernetics, SMC-17(5)1987
- [2] M. Genesereth, S. Ketchpel, Software agent, CACM, 37(7)1994
- [3] Yoav Shoham, Agent-oriented programming, AI, 60, 1993