

密级: \_\_\_\_\_



**中国科学院大学**  
University of Chinese Academy of Sciences

# 博士学位论文

基于智能体的协同认知研究

作者姓名: \_\_\_\_\_ 张建华

指导教师: \_\_\_\_\_ 安波副研究员 史忠植研究员

\_\_\_\_\_ 中国科学院计算技术研究所

学位类别: \_\_\_\_\_ 工学博士

学科专业: \_\_\_\_\_ 计算机应用技术

研究所: \_\_\_\_\_ 中国科学院计算技术研究所

2014年6月



# Agent-Based Collaborative Cognition

By  
Zhang Jianhua

A Dissertation Submitted to  
University of Chinese Academy of Sciences  
In partial fulfillment of the requirement  
For the degree of  
Doctor of Philosophy

Institute of Computing Technology  
Chinese Academy of Sciences  
June, 2014



## 声 明

我声明本论文是我本人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢的地方外，本论文中不包含其他人已经发表或撰写过的研究成果。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

作者签名：

日期：

## 论文版权使用授权书

本人授权中国科学院计算技术研究所可以保留并向国家有关部门或机构送交本论文的复印件和电子文档，允许本论文被查阅和借阅，可以将本论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存、汇编本论文。

（保密论文在解密后适用本授权书。）

作者签名：

导师签名：

日期：



## 摘 要

随着科学的进步，各国纷纷制定了脑科学的长久发展计划，越来越多的人力资源和物质资源投入到了脑科学的研究中。经过了20多年的研究和发展，智能体和多智能体系统已经成人工智能甚至是计算机科学的研究热点，并在广泛的领域中得到了应用。随着计算机技术的不断进步、云计算的不断发展，这种动态不确定的计算环境为智能体技术提供了展现舞台。脑机接口研究的不断深入，使得大脑和计算机之间的信息交互更加的密切。为了更好的协同配合，我们可以将大脑的生物智能和计算机的计算智能相结合，成为混合智能。智能体中心智状态模型、内部运行推理机制成为研究智能体的核心问题，是智能体技术的关键部分。

本文在对已有智能体模型、智能体设计以及相关研究的基础之上，重点从动机、目标生成、目标动作序列规划对智能体进行研究。本文研究多智能体协同认知基本问题，建立动机模型，并在此基础上建立感知模型。单个智能体的能力是有限的，对于某些特定的问题，需要多个智能体协作完成，为此我们研究了多智能体的合作。当特定的目标建立之后，智能体需规划自己的动作。

本文的主要工作和创新点总结如下：

### 1. 提出了NGI动机模型。

动机是一种非常复杂的现象，根据我们承担的具体课题，我们提出了一个动机模型。N表示Need(需求)，G表示Goal(目标)，I表示Intensity(强度)。智能体不断的同外界进行交互，在交互的过程中会产生各种动机。为了对动机更好的进行建模，我们将动机分为感知动机、适应性动机和合作动机。动机的生成需要目标来解决，因此，我们的动机模型中确定了目标。不同动机具有不同的重要性，为此，需要对动机的强度进行标注。

### 2. 提出基于智能体的ABGP协同认知。

其中A表示Awareness（感知），B表示Belief（信念），G表示目标（Goal），P表示Plan（规划）。建立动机模型，并对生成的目标进行规划。智能体所处的环境进行着动态的变化，为此，我们需要对其所处的外部和内部环境进行感知，将信息及时传输到智能体。一个目标可能对应这多个动机，因此，我们需要按照动机的优先级别生成目标。目标生成后，选择优先级别最高的进行执行。对于已生成的目标，包括其初始状态和最终状态。智能体从其情景记忆中选择动作序列，进行动作的规划。

### 3. 提出基于信息共享的多智能体协同。

该方法通过采用信息桥接方式，打破了消息传输的二义性，方便了智能体之间信息的传输和检索。桥规则增强了知识库之间的互操作，使得信息可以按照我们预想的方式传输。在进行推理的时候，首先在智能体内部进行推理；在不能得出推理结果的时候，选择同时具有into和onto桥规的的候选智能体进行推理；如果不能得出结果，再选择三个智能体，其中一个只有into桥规则连接，一个只有onto桥规则连接，然后在选择一个智能体同时和这两个有智能体有连接，进行推理。

**关键词：**动机；规划；协同；智能体；动态描述逻辑；推理；知识；认知模型



## **Agent-Based Collaborative Cognition**

Zhang Jianhua (Computer Application Technology)

Directed by Professor An Bo and Professor Shi Zhongzhi

### **Abstract**

As the development of science and technology, many countries have formulated the long-term development plan of the brain science. More and more human and material resources had been sink into the research of brain science. After 20 years research and development, agents and multi-agent systems have become the artificial intelligence hot spot, and event in the computer science. What's more, they it has been applied in wide areas. With the progress of compute science, the development of cloud computing, the uncertain comparting environment provides the a state. As the research in brain computer interface, which make the information exchanging between brain and computer more closely. We can merge the biological intelligence and computer intelligence to collaborate. The model of agent mind, inner reasoning are become the core of the agent research, which are the key research point.

Based on the agent model,agent design and relevant research, we focus on the motivation, goal and goal plan research. The goal of this thesis is study the fundamental problems of cognition, construct the motivation model, and construct the cognition model. As the single agent has limited capacity, for the specific problem we need more agents to collaborate. When the specific goal established, agent need to plan their actions to reach it.

The main contributions of this thesis are summarized as follows:

- 1: Propose a NGI motivation model.

Motivation is a more complex phenomenon. According to the task we undertake, we propose the NGI motivation model. Where "N" means "need", "G" means "Goal", and "I" means "intensity". Agent communicates with the environment,and then generates the motivation. To model the motivation, there are three kinds of motivation: perception motivation, adaption motivation and cooperative motivation. The goal can solve the motivation, so each motivation has a goal. Each motivation has its attribute of weight.

- 2: Propose an agent ABGP cognitive model.

Where "A" means "Aware", "B" means "Belief", "G" means Goal, "p" means "Plan". We construct the motivation model, and then plan the goal. As the environment dynamic changing, agent needs to percept the inner and outer information. One goal may be consists lots of motivations, so we need construct the goal according the motivation weight. Once the goal established, we select the highest weight goal to executive. The goal consists the initial state and end state. Agent select the action from the episodic memory.

3: Propose an information sharing method based on bridge rule.

It is using bridge-rule to break the ambiguity of message transmission, which make our agents can get more information from other agents conveniently. The bridge enhance the knowledge base mutual operation, which make the information transmits as our expect. When we reasoning the knowledge, first we reason it int he agent's own knowledge; if failed,we select candidate agents which have both into bridge and onto bridge the continue reason; if also failed, we select three candidates, one has onto bridge,and the other has into bridge and the last one has both into bridge and onto bridge to reason.

**Keywords:** motivation, plan, collaboration, agent, Dynamic Description Logic, reasoning, knowledge, cognitive architecture

# 目 录

摘要	I
目录	V
图目录	IX
表目录	XI
第一章 绪论	1
1.1 研究背景与意义	1
1.2 脑机融合	3
1.3 研究现状及概述	6
1.3.1 动机相关进展	11
1.3.2 典型认知模型	14
1.4 本文的主要贡献	17
1.5 论文的组织	18
第二章 基于动机的协同认知	21
2.1 引言	21
2.2 动机模型	22
2.3 动机系统	24
2.4 动机执行流程	25
2.4.1 目标生成	27
2.5 本章小结	29
第三章 基于智能体ABGP协同认知	33
3.1 概述	33
3.2 心智模型CAM总体结构	33
3.3 动态描述逻辑基础	35

3.3.1	基本概念	36
3.3.2	动态描述逻辑语义	38
3.3.3	动态描述逻辑推理任务	39
3.4	智能体ABGP认知模型	41
3.4.1	环境感知	42
3.4.2	信念	43
3.4.3	目标	44
3.5	策略驱动的动作规划	44
3.5.1	动作策略	45
3.5.2	目标策略	46
3.5.3	效用函数策略	46
3.5.4	动作规划简介	46
3.5.5	基于动态描述逻辑的规划	48
3.6	联合意图	55
3.6.1	逻辑框架	56
3.6.2	联合意图理论框架	58
3.6.2.1	联合实现型意图	59
3.6.2.2	联合维护型意图	60
3.7	相关研究	62
3.8	小结	63
<b>第四章</b>	<b>基于智能体信息共享的协同认知</b>	<b>65</b>
4.1	引言	65
4.2	知识传播	66
4.3	CD3L知识库	67
4.3.1	CD3L语法	67
4.3.2	CD3L语义	69
4.4	支持链式桥规则的分布式动态描述逻辑推理	71
4.4.1	CD3L算法的一致性	71
4.4.2	一致性讨论	72
4.5	小结	73

第五章 智能体协同认知原型系统	77
5.1 引言	77
5.2 智能体体类型	78
5.3 多智能体Web服务组合	81
5.4 小结	85
第六章 结束语	87
6.1 本文主要贡献	87
6.2 下一步研究方向	88
参考文献	89
致谢	i
作者简介	iii



## 图 目 录

1.1	认知模型示意图 . . . . .	6
1.2	马斯洛需求 . . . . .	11
1.3	ERG理论模型 . . . . .	13
1.4	PSI 需求模型 . . . . .	14
1.5	Soar认知模型 . . . . .	15
1.6	本文的内容组织结构图 . . . . .	20
2.1	动机系统 . . . . .	25
2.2	动机执行流程 . . . . .	26
3.1	心智模型CAM总体结构 . . . . .	34
3.2	智能体ABGP认知模型 . . . . .	42
3.3	多智能体策略驱动的状态转移 . . . . .	46
3.4	树形目标存储结构 . . . . .	49
3.5	情景记忆片段 . . . . .	50
4.1	分布式知识传播 . . . . .	67
4.2	into桥规则 . . . . .	68
4.3	onto桥规则 . . . . .	68
5.1	Agent-0型智能体 . . . . .	78
5.2	Agent-1型智能体 . . . . .	79
5.3	Agent-2型智能体 . . . . .	79
5.4	Agent-3型智能体 . . . . .	80
5.5	Agent-4型智能体 . . . . .	80
5.6	多智能体信息共享 . . . . .	81
5.7	Web服务组合平台结构图 . . . . .	83





## 表 目 录

1.1 Soar认知记忆 . . . . .	16
------------------------	----



# 第一章 绪论

## 1.1 研究背景与意义

如何进一步的了解人类的大脑是21世纪科学界面临的课题之一。大脑是人类最为复杂的器官。为了探究大脑的奥秘，人类一直在进行着不懈的努力。随着信息技术的进步，人工智能得到了长足的发展。世界各国相继在脑科学方面投入了巨大的科研力量。各个国家分别积极的推出了自己的研究计划。从一个侧面显示了脑科学研究的重要和人们对此高度的重视。

2013年1月28日，欧盟委员会宣布，将石墨烯和人类大脑计划两大科技计划入选“未来新兴旗舰技术项目”，并分别设立专项研发计划，每项计划将在未来10年内分别获10亿欧元的研发经费。其中，人类大脑计划（Human Brain Project, HBP）由13个子项目组成[32] [34]。它们是：(1)老鼠大脑策略数据，该子项目产生战略性的数据，建立小鼠大脑数据，并和人脑进行比较；(2) 人类大脑策略数据，该子项目的目的是收集人类大脑的数据，以检测通过小鼠大脑数据来预测人脑行为的正确性；(3) 认知模型：该子项目选择定义良好的认知任务，成功的认知模型将应用到系统的认知平台；(4) 大脑研究的数学和理论基础：从数学角度对HBP给予理论上的支持，对HBP项目具有很大的帮助；(5) 神经信息学：帮助神经科学家组织和访问异构数据，提供多层次的脑图谱信息；(6) 大脑模拟：该子项目提供可以通过因特网访问的大脑模拟平台；(7) 高性能计算：满足HBP 的具体挑战性要求，该子项目将开展更多更具体的ICT技术研究、创新节能技术的开发；(8) 医学信息学：有助于为疾病治疗提供预测结果，为决策者真正将治疗方案应用于人体或者是动物之前提供宝贵意见；(9) 神经形态计算：HBP 设计、实施和应用一个神经形态计算平台，该平台可以为非专业的神经科学家和工程师提供可配置的神经形态计算系统，该系统可以方便科学家进行试验；(10) 神经机器人：该子项目将会为科学家和技术开发者提供软件和硬件的基础设施，模拟环境设计和运行在认知神经学的简单试验；(11) 应用：该子项目是实现HBP是实用化的第一步，主要分为三个研究方面：未来神经科学、未来医药和未来计算；(12) 伦理学和社会：HBP 将要推出一个重大的伦理和社会方案，该方案考虑了对社会、伦理和道德的影响，该计划利用基因组学、合成生物学、纳米技术、信息和通讯技术等验证开发方法；(13)HBP 项目管理：该子项目将支持HBP 决策、经营管理和研究方案，其主要职责包括协调科学发展蓝图、确保项目对投资者的利益[90]。

2013年4月2日，美国总统奥巴马宣布了一项重大的科研计划（Brain Research

through Advancing Innovative Neurotechnologies, BRAIN), 用于揭开人类大脑的未解之谜, 该计划是一个新的研究重点, 旨在彻底改变我们对人类大脑的理解[90]。计划得到了1亿美金的支持。科学家们将该计划称作“大脑图谱”工程, 官方正式命名为“运用先进创新型神经技术的大脑研究”研究计划。本计划由美国国家卫生研究院、国防高级研究计划局及国家科学基金会实施。鉴于该项目的跨学科性, 美国国立卫生研究院正在与其他政府机构, 包括国防部高级研究计划局(DARPA)和美国国家科学基金会(NSF)的密切合作。首期的投资将要侧重大脑研究的基础科研。科学家们为此将要开发新的工具, 以便能够剖析特定神经元的底层作用, 了解大脑的电路运行机理。通过加快技术创新, 研究人员可以展示单个细胞和复杂神经回路在时间和空间相互作用的革命性动态画面。科学家们将通过努力找到治疗大脑疾病的新方法, 这将填补大量研究的空白, 同时为我们探索大脑记录、处理、利用和检索信息提供前所未有的机遇。在2014财年, 主要的研究领域有: (1) 生成细胞图谱: 在这之前需要开发工具记录、标记和处理这些定义的神经元, 进而对其进行标注; (2) 生成大脑结构图谱: 使得我们可以发现大脑神经电路和大脑各个部分之间的联系, 懂得神经结构和功能之间的关系; (3) 开发新的大型网络记录功能: 从神经网络记录神经行为对我们来说是一个机遇, 我们更应该抓住; (4) 开发电路操作软件包: 通过直接激励和禁止神经元的方法, 科学家可以从观测中找到因果关系; (5) 将神经元的行为和活动连接起来: 巧妙的应用虚拟现实、机器学习等方法, 我们可以知道在认知和行为中的神经元活动; (6) 通过实验整合理论、建模、统计和计算之间的关系: 严格的理论、模型等有助于我们了解大脑的功能; (7) 找到人的大脑图像处理技术: 我们必须提高空间分辨率和/或人脑成像技术的时间采样; (8) 建立新的机制, 以利于人类大脑的数据; (9) 传播知识和培训: 通过将技能在社区中迅速传播, 大大加快知识的传播进程[91] [57]。

认知科学是对心智(mind)和智能(intelligence)的跨学科研究, 包括哲学、人工智能、神经科学、语言学 and 人类学等。认知科学的目标是揭开心智的秘密, 提高人类认知决策的能力。它的学术研究起源于20世纪50年代中期, 早期的科学家借助复杂的表征(representation)和计算程序来发展关于心智的理论。1979年, 美国认知科学学会(Cognitive Science Society)正式成立, 从此众多的学术机构建立了自己的科研队伍, 多种认知科学项目相聚展开, 甚至好多学校开设了与之先关的课程。例如, 哈佛、剑桥、麻省理工、斯坦福等。同时, 在此前后, 各种和认知相关的期刊杂志相继创刊, 如美国认知科学学会的《认知科学》杂志、《认知心理学》杂志、《认知》等等。认知科学成为一门学科的主要标志是: (1) 《认知科学》的创刊; (2) 斯隆(Sloan)论述了认知科学的技艺; (3) 认知学会的建立[106]。

我国的认知科学虽然起步较晚, 但是仍然积极有效的开展了前沿问题的探索。如

北京大学成立了以心理学系、生命科学院、信息科学中心等为骨干成员组成的脑科学与认知研究中心。北京师范大学成立了认知神经科学与学习国家重点实验室，该实验室在语言认知、数学认知、社会认知、心理发展和脑发育等研究领域进行了相关的科学研究。

本课题的研究目标是研究协同认知的基本问题，包括建立动机模型，建立认知模型，并在此基础上对认知领域的一些重要问题进行研究，包括基于认知模型的多智能体协作以及动作规划问题。

### 1.2 脑机融合

随着计算机的不断发展，人工智能科学家一直在努力寻找一种方式，使得计算机变得像人脑一样聪明，但目标的实现非常的不容易。科学家逐渐意识到大脑与机器融合并协同工作是一条非常有效的捷径。近年来，信息技术、脑科学、材料科学、通信技术的不断发展，使得大脑和计算机的结合越来越紧密。

脑机接口在大脑和计算机之间建立了信息传输的通道，从而使得计算机和大脑可以进行信息的交互。按照信息的传输方向，可以分为两类，即从大脑到计算机的传输和计算机到大脑的传输。对于由大脑到计算机的信息，我们可以记录大脑相关区域的神经信息，并转化为电信号，从而控制外部的设备。对于从计算机到大脑的信息，我们可以对大脑的相关区域进行电刺激，将计算机中的信息通过通信系统传输到动物的大脑中。目前研究人员对脑机接口中单向信息传输方式研究的是最多的，直到近期才开始涉猎双向信息传输的脑机接口。

我们在利用大脑的可塑性，结合机器学习的最新技术可以构建大脑和计算机相互适应的系统，从而可以提高脑机接口的性能。有的采用了自适应的算法改变了计算机适应大脑的能力，有的将强化学习应用脑机接口，利用奖惩机制对大脑活动进行自动调节。

我们可以看出，对于脑机接口的研究已经呈现了从脑机接口、脑机交互到脑机融合的发展趋势。但是如何发挥脑机融合系统中的生物智能和机器智的优势，我们需要解决很多的科学问题。如脑机融合过程中的协同感知、认知以及动作规划问题。为此，我们需要建立与之相对应的一整套新的计算理论和方法，为生物智能与机器智能互联互通打下基础。

大脑是一个异常复杂的系统，其具有很强的组织性和适应性，内部具有非常强大的计算处理功能。动物具有非常强大的感官系统，可以方便的对外部的信息进行接收。大脑将感知到的信息进行整合，形成对客观世界的认知，根据自身的计算，产生动机和目标，驱动动作的产生，并将动作应用与外部的环境。

感知是外部的客观世界直接作用于人的感觉器官而产生的。人具有听觉、视觉、嗅觉、味觉等不同的感知器官，这些感知器官可以实时的获得外部的信息，并将其不断的传入大脑中。在感知的多种研究中，其中视觉感知是人们研究较多的领域。

随着对计算机烟具的不断深入，计算机在众多的应用领域展现了超越人类智力的属性，主要得益于在信息存储、数值计算以及超大规模集成电路的不断发展。但是在模拟人类思维过程以及智能行为的方面而言，突破性的成果仍然非常的缺乏。现有的人工智能算法对于约束紧密、处理对象少的简单环境中的特定问题具有其特定的优势，但是对于复杂环境中的问题仍然缺乏行之有效的解决方法。

脑机融合被认为是影响21世纪重要的科技之一。对于脑机的研究，将会带动多个学科的发展，如信息科学、人工智能和认知科学。生物智能与机器智能各有优缺点，生物智能具有具有感知能力，而机器智能具有计算能力的优势，如果将两者互相结合，则会产生令现有的生物智能系统和机器智能系统望尘莫及的混合智能系统。通过自然科学手段对脑和认知进行研究，是人工智能、信息科学等学科进行研究的新的领域。对于大脑和认知科学的研究，已经成为自然科学研究的重要前沿，该领域受到了各国政府的高度重视，并提升到战略高度给予支持。与之相关的研究极大的促进了人工智能及信息科学发展，同时为实现更加智能的计算机系统通过了模拟的对象和生物模型。现阶段，研究人员通过大学科背景、多学科交叉、国际合作等方式进行对大脑和认知的研究。科学家高度重视将脑科学与信息科学进行深度的结合，从而形成新具有高度智能的系统。在《国家中长期科学和技术反发展纲要（2006-2020）》中，已经将脑科学和认知纳入国家重点支持科研领域。该研究具有学科交叉和前沿的双重特点，已经成为国家学术界所关心的问题。该项研究在激烈的国际竞争中具有居中轻重的地位，同时也是国家战略必争的重要领域，其研究将推动信息产业进行一个新的发展时期。

计算机先驱图灵将大脑作为计算机的原型，进而提出了图灵机原型，从而进一步奠定了现在计算机科学的基础。在认知科学中，人们将大脑看做是具有信息处理功能的系统，它是实现计算机系统的原型。两者相互依赖，促进了计算机科学和认知科学的发展。近年来，随着脑机接口技术的迅速发展，计算机与大脑之间的连接越来越紧密。借助脑机融合，可以建立大脑与外部设备之间信息的交互。

动物机器人是将微电极植入动物的大脑，通过电刺激的方式向动物的大脑下发指令，从而实现对动物行为的控制以及感知功能的增强，这在搜救方面具有极其重要的应用场景。

脑机融合研究的展开，具有重要的科学意义，主要包括：

- 揭示大脑处理信息的机制，推动人工智能的发展

随着研究的深入，将促进人们了解和认识智能的本质及其一般规律，为智能系统提供更加清晰的仿生模型和理论依据。有助于对智能科学体系的完善，加深人工智能研究的内涵，推动人工智能学科向前发展。

- **探索脑机融合的新理论，拓展智能系统研究的新领域**

研究脑机融合过程中的基本问题，如信息表示、知识表达以及动作决策。这将开辟智能系统研究的新领域，创立脑机融合的新理论，探索研究实现生物智能与人工智能的混合体系结构。在未来，相关的研究成果很有可能开辟人工智能研究的重要分支领域。

- **创建动物机器人新技术，提升战略性信息技术核心竞争力**

面对国家的需要，将大幅提高人造系统的智能水平，催生战略性的信息技术新产业。随着研究的深入，将会提升我国在相关领域研究的核心竞争力。

脑机系统的认知计算模型将为脑机融合提供脑机融合过程中的系统架构。脑机协同的认知过程主要包括环境感知、动机解析、目标生成以及动作规划等过程。在脑机认知模型中，记忆具有至关重要的作用。系统中主要包括感知记忆、情景记忆、语义记忆以及工作记忆。

脑机系统的环境感知主要是认知模型对智能体所处的外部环境进行信息的感知。智能体所处的外部环境不断的变化，为了更好的处理智能体所面对的复杂的事物，对外部环境的感知至关重要。其中的视觉感知，智能体可以其所处的外部环境的视觉信息进行感知。智能体对其所检测到的视觉信息进行语义分析，得出其所处外部环境中的真是状态，生成高层语义，有效的理解环境感知的内容，建立脑机协同的感知模型。智能体对外部环境的感知，不仅仅可以通过视觉对外部环境进行感知，还可以通过其它感知的方式，比如触觉、听觉等。如图1.1所示。

在所有的感知研究中，视觉研究在认知中一直具有重要的地位。在视觉研究的过程中，占主导地位的是以Marr为代表的特征分析理论，该理论认为视觉是由局部性质到整体性质主导的。而Gestalt理论则认为视觉过程是有大范围整体到局部的性质进行的。

对感知信息的理解是研究中的难点，其具有复杂性、抽象性等特点。在研究的过程中，需要利用高层语义对信息的抽象属性进行描述。对于不同层次的内容，可以采用不容的信息语义。

在认知中，记忆也非常的重要。情景记忆是个人的回忆。情景记忆有许多构成要素，同时还具有要素之间的关系。我们可以将构成要素分为两类，一类是可以观察的



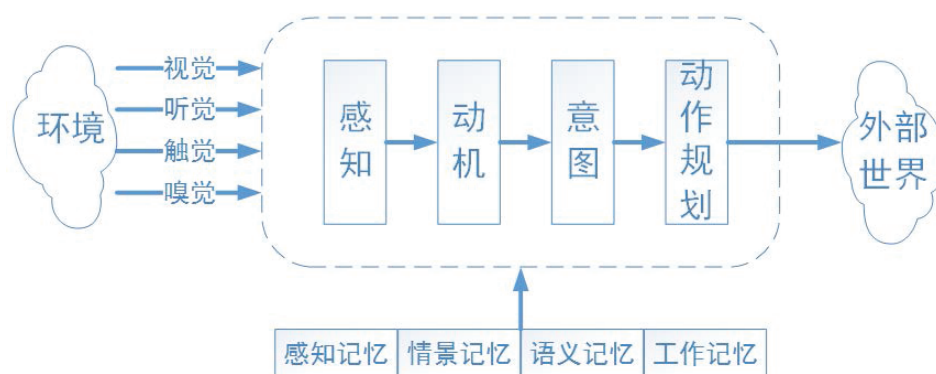


图 1.1: 认知模型示意图

事件，一类是假说构成的概念。片段记忆是长期的，具有语境环境的信息。片段记忆用来描述一个符号化形式的描述信息，其实抽象的。对于片段信息，我们需要对其进行存储。存储的主要信息有片段的序号、片段发生的时间、片段记忆的描述以及片段记忆中相互关联的感知信息。通过指针关联，认知系统可以通过将片段记忆中的信息与感知缓冲区中的信息进行关联。智能体可以通过多种方法进行自动推理，如案例推理、神经网络等方法。语义记忆的基本单元是概念，且每个概念都有自己的特征。语义模型在语义推理中具有至关重要的作用，它是语义推理的基础，决定了语义理解的深度、广度和易用度。

### 1.3 研究现状及概述

人工智能经过50多年的发展，但是还远远没有达到人类智能的水平。将计算机的快速信息高速处理能力同人类大脑高级智能相结合而形成的类脑计算机，可以实现高性能的信息处理。位于瑞士洛桑的联邦理工学院脑科学中心在2005年合作发起了一项名为“Blue Brain Project”的哺乳动物大脑逆向工程，该项目致力于利用超级计算机模拟哺乳类动物大脑[12] [108]。该计划的主要内容是在超级计算机上进行仿真。把所有已知的脑数据和知识整合在多层次、多尺度的脑模型中，使其在许多细节方面具有生物特性的脑模型[67] [39]。

人工智能长期的目标是实现一些系统，且这些系统可以在大多数的领域超过人类的智能水平[31]。伴随着时代的发展，心理学越来越受到重视。科学实验表明，情感是智能的一部分而不是与智能相分离的。情感能力对于计算机与人的自然交往至关重要。传统的人机交互，主要通过键盘、鼠标、屏幕等方式进行，只追求便利和准确，无法理解和适应人的情绪和心理。而如果缺乏这种情感理解和表达能力，就很难指望计算



机具有类似人一样的智能，也很难期望人机交互做到真正的和谐与自然。情感包括感知、动机、思考和动作等。

智能科学研究智能的基本理论和实现技术，是由脑科学、认知科学、人工智能等科学组成的交叉学科。脑科学从分子水平、细胞水平、行为水平研究人脑智能机理，建立模型，揭示人脑的本质。认知科学是研究人类的感知、学习、记忆、思维、意识等人脑心智活动过程的科学。人工智能是利用人工智能的方法和技术，模仿、延伸和扩展人的智能，实现机器智能。智能科学是实现人类水平的人工智能的重要途径。智能科学对类脑计算机的研发，起着推动指导的作用[97]。

认知是一种能力，它用来理解事物，不仅仅是现在的情况，还要预测事物将来的情况，并根据具体的情况产生相应的动作[111]。认知模型类似于计算机的体系结构。认知系统具有记忆、处理和控制单元，通过知识进行推理，完成系统的目标。系统中的知识可以通过学习进行更新。学习可以更新知识，但是学习不能够改变认知结构。认知模型的一个挑战是用来协调职能系统的各种能力，如感知、推理、规划、语言处理和学习。认知模型可以感知外部的环境，不仅可以了解环境当前的状态，对将来的环境状态进行预测，并且考虑需要采取的动作。将过去发生的事情存储在自己的记忆中，可以作为将来推理的知识。因此，记忆非常重要，过去的记忆可以用作将来的知识[21]。

认知模型明确提出了一个智能系统需要具备的基本组成部分。简单来说，主要有包含以下三个部分：

- 短期记忆和长期记忆，用来存储智能体的信念、目标和知识；
- 智能体内知识的表示，以及知识的组织；
- 运用知识进行推理，以及通过学习进行知识的更新。

对于特定的任务，人类可以不费力的采用自己的认知能力来解决，并且可以解决多种问题。当人类遇到不容易解决的难题时，我们也总会尽力应付过去。我们可以通过规划来对问题进行求解，当我们自身的资源或者能力有限的时候，我们还可以求助他人。例如，我们可以同自己的朋友借自己缺失的原材料，我们还可以让自己的朋友来帮助自己完成复杂的任务。我们可以在实践中学习，最终会成为一个领域中的专家[58]。因此，在一些特定的领域，认知建模具有无可替代的优越性。例如：

- **无人飞行器编队飞行控制**

随着科学技术的发展，各国都相继研发了各种型号的无人飞机。美国作为全球第一的军事强国，无人机的种类和数量都居第一。从而可以看出，美国无人机的发

展方向代表了世界无人机的发展趋势[5]。无人驾驶飞机系统在军事中发挥了越来越大的作用。随着科学技术的进步,无人驾驶飞机具备了越来越多的功能。现代的无人飞机不仅仅是执行情报、监视和侦察任务,现在还可以执行电子攻击、控制或者破坏放开的任务。由于单架无人飞机的能力有限,这就促使在战场上部署更多的数量,这样就可以执行更多更复杂的任务。实现协同打击的关键是规划出多无人机从各自起始点到目标点的最有协同攻击路径[6]。

### ● 计算机兵力生成

计算机生成兵力(Computer Generated Forces, CGF)是作战仿真中的自动化或者半自动化的作战实体。这些实体由计算生成并且控制,按照预先成的角色共同进行战术演练,协同完成总体任务。美国国防部将人类行为表示(HBR)列为该国防部建模仿真的六大目标之一。根据这一目标,专门成立了研究小组,对HBR进行广泛深入的研究。研究的重点主要在多Agent 认知体系结构、感知、协同等。在理论方面主要有Cohen&Levesque[28][105]等在1990年提出的联合意图理论。该理论关注团队的共同信念、联合目标、联合意图等等。为使所有的智能体共同实现一个目标,团队之间需要实现协作,首先需要建立联合的目标。且需要大家共同遵守,当目标已经实现或者目标不能够实现的时候,需要放弃目标,并通知所有的团体成员。Grosz&Kraus[45][46]提出了共享计划理论。该理论与联合意图不同,其并不是基于意图,而是通过实体常规意图采取行动。该理论致力于在协同的过程中描述整个团队的意图和信念。

### ● Web服务组合

随着Web服务技术的迅速发展,Internet上Web服务无论是类型还是数量都有了爆炸式增长。Web应用从局部发展到了全球,从集中式发展到分布式,变成了一种新的分布式服务计算模型[2]。然而单个Web服务的功能有限,如何协调和组织多个Web服务来构造新的Web 服务、提高服务组件的重用性和利用率,成为Web服务领域最具有挑战性的问题之一。Web服务(Web Service)是由IBM和Microsoft等共同提出,其目的是为Internet上跨越不同的区域、不同的行业的应用提供方便的操作。W3C 对Web服务做了如下定义: Web服务是有一个URI标识的软件一个软件应用,它的接口和绑定方式可以给予XML 来定义、描述和发现;基于当前的Internet协议, Web 服务支持以XML格式的消息交换的方式与其它的软件进行直接交互[16]。语义Web 服务组合中比较有影响力的语言有: OWL-S[70], WSML[29], SWSO[19]和WSDL-S[10]。

### ● 电子计算机游戏

随着计算机技术的不断发展，计算机计算能力得到稳步的提升，计算机游戏得到了快速的发展。在计算机游戏中，为了提高游戏的趣味性，游戏变得越来越复杂，最典型的是团队作战。在游戏CS中，我们可以选择两个队伍，在每个队伍中设置玩家的个数，同时还可以添加机器人，其中的机器人就具有人类的思维。机器人可以感知到同伴的状态，感知到对手的状态，还可以有诸多的参数设置，这极大的提高了游戏的可用性。在这放面，国产游戏《光荣使命》，也体现了团队作战的重要思想，为我们的认知提出了更高的要求。在学术界有robocup[9][77]，该项目的的主要目的是提供一个易于评价的比赛平台，促进人工智能的发展。

在随后的部分中，我们将进行相关工作的讨论。主要是对现有感知系统进行分析 and 考察。

认知计算是新近涌现出的智能计算方法和系统，它模仿了大脑的推理和感知机理，实现了自治体的感知技能[113] [114] [115] [121] [116] [117] [118] [112] [119] [120]。智能计算基于跨学科的研究。据我们所知，计算机已经达到了一个新的拐点，但是下一步如何发展，现在仍然是未知数。计算机自从1940年以来，得到了迅速的发展。

自从古代以来，为了更好的了解大脑，人们一直在进行着各种实验尝试。经过人们不懈的追求，最终促进了心理学的发展。随着心理学的不断发展，认知科学拓展了心理学研究领域，它更加关注知识在人类大脑中的存储和转换。作为认知科学研究的一个有用工具，人工智能作为计算机的一个分支，主要用来对人们的认知行为进行建模并进行模拟。在认知科学研究的过程中，人工智能不仅仅是作为一个工具，同时它还有科学的用途，即关注机器的认知行为[101]。现在认知科学研究的重点是大脑、动物和机械如何进行信息的感知、推理和与环境的交互。

为了彻底的理解认知科学，我们需要从多个角度对心智进行分析。换句话说，我们需要采用多种技术对心智进行评估和了解。人工智能是一个非常高效可行的方法，它使得我们可以对人类的行为进行计算建模。有许多的方法对心智进行建模，包括神经元的生成和监测，心智被表示为一系列高层次的规则，符号和规划。人工智能除了对人类的心智进行建模模拟，同时尝试将认知现象在机器中用程序进行实现。这些程序可以用来处理许多复杂的事物，且比人类自己做的更加出色。认知科学中的成就会帮助人工智能科学更好的对人类的心智进行模拟。反过来，人工智能可以提供更加精确的模型。虽然认知科学和人工智能的目标不尽相同，但是这两个领域的合作对各自的成功非常具有帮助。人工智能和认知科学相互进行渗透，最终达到各自的目标。

认知科学中最根本的目标是利用计算机的强大资源对人类的思维进行理解和补充。在人工智能中，一个智能体可以用计算机来进行计算模拟，这个智能体和环境进行着不断的交互并在这期间实现自己的目标。智能体首先对需要解决的问题进行分析，从

而找到解决方案。认知科学家的目的是找到一种理论，该理论可以用来解释大脑学习和构建逻辑规则的过程，在这个过程中智能体现在信息的选择性记忆和遗忘，和人类利用的资源。

除了对人类的心智进行透彻的了解，认知科学最终的目标是获得具有人类水平的机器智能。在这个水平上，智能体的智能和人的智能没有区别，甚至比人类更加的智能。由于智能体经常处于信息不够完全的环境，智能体只能在某种程度上对人类的智能进行模拟。换句话说，在真实的环境下，会有多种解决方案，这使得我们在设计智能体的时候，智能体同样应该计算这么多的解决方案。事实上，智能体必须具有根据现有的执行进行决策的能力，并根据过去的信息对现有的解决方案进行评估，选择出最优的解决方案。因此，对人类大脑的深层次理解，尤其是如何分析和解决问题的能力，对我们设计智能体具有举足轻重的作用。当前，多种研究项目，一部分是用来模拟人类之智能了解人类大脑，一部分用来解决复杂的问题。

外部环境的刺激，人类的大脑将接收到的信息进行处理。随着计算机的发明，计算机同样可以进行信息的处理。但是两者之间的区别是什么呢？历年来，人类认知的研究者采用计算机对图像和概念从侧面对人类的心智和概念进行分析[26] [69]。进来，认知科学从计算机科学中找到了灵感对人类大脑进行研究[65]。主要包括大脑及认知系统的联系和机器学习[66]。

我们对人类的意识的概念和经验都非常熟悉。意识和人类大脑的执行功能相关，它需要注意、信息集成和制定决策。我们可以说意识和高级认知功能相关，但是只有时间可以告诉我们动物和机器我们能否看做是具有意识。

脑机接口发展非常的迅速。脑机接口一般与脑电图相关。我们可以通过脑电图阅读大脑中的信息。如果我们对脑电图进行检测，并将信号转化为电信号，那么我们就可以将这些信息进行储存和分析。未来技术的发展暗示了脑机接口革命性的信息处理功能。在现阶段，脑机接口中我们对可用性和可访问性考虑的较少。

认知科学家和哲学家将心智看做是可计算的。一些神经科学家的研究表明心智也是可以进行计算的。计算的普遍性包括数字计算、模拟计算、神经计算、量子计算以及其他形式的计算[81]。当智能体系统初始化完毕之后，智能体会同外部的环境进行不断的交互。在智能体中，通常具有一个确定性的算法或者是程序用来进行决策，同时还有同外界环境进行交互的机制。在智能体中具有传感器、控制器和执行器，且智能体和环境实时的进行交互。

现在我们把计算机看做是信息处理的机器。同时我们所处的外部环境中，充满了各种各样的信息。

在智能科学中，信息处理是其具有的基本功能。智能体收集各种时空信息，计算



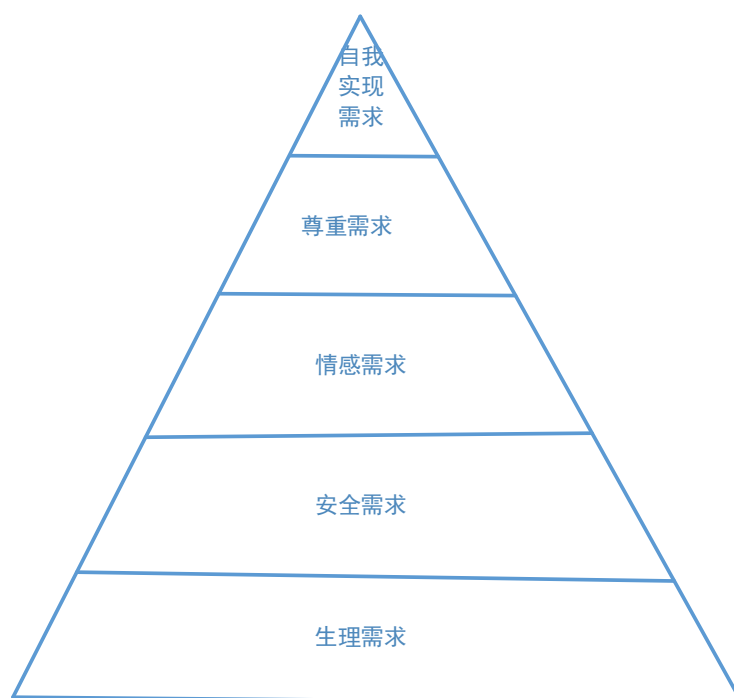


图 1.2: 马斯洛需求

机可以方便的对这些信息进行收集和处理。

### 1.3.1 动机相关进展

在智能科学和人工智能学科中，动机是一个非常重要的研究内容。动机的概念最早起源于心理学，Maslow曾经对人类行为进行了研究。最终，在其论文中指出，人类的需求由一个层次化的结构组成，这个层次化结构中主要包含五个部分[71]。自从该理论发布到了社会当中个，该理论就被人们广泛接受，且产生了非常重大的影响。该理论可以给人们带来许多的精神和动机，所以可以帮助人们很好的管理自己的生活。Maslow的需求层次理论与人类生活中的经验是非吻合的，所以确实得到了人们的认可。Masow的需求层次理论，将人们的基本需求分为了五个层次，如图1.2所示：其中第一层次的需求是生理上的需求；第二层次的需求是安全需求；第三是社交需求；第四是尊重需求；第五是自我实现需求。该理论是行为科学的理论之一。生理需求得不到满足，人类的生理机能就无法正常运转。换句话说，在这种情况下，人类的生命就会受到威胁。在某种意义上说，生理需求是人们行动最首要的动力。马斯洛认为，只有这些基本的需求得到满足和维持后，其它的需求才能才能成为新的激励因素。当基本需求满足后，满足的需要就不能够在成为新的激励因素了。当人们的生理需求满足之后，安全需求就提上了日程。整个有机体有一个追求安全的机制，安全需求包括人身安全、

生活稳定以及免遭痛苦疾病等需求。和生理需求一样，在安全需求得到满足之前，人们唯一关心的就是这种需求。社交需要时主要包含两部分的内容。意识有爱的需要，即人人都需要伙伴之间、同事之间的关系融洽和忠诚；人人都希望得到别人的关怀。二是归属的需要，即一个人归属于一个群体，希望成为群体中的一员。人人都希望有自己的社会地位，希望自己的能力和成就得到社会的认可。尊重主要分两类，一类是内部尊重，一类是外部尊重。内部尊重是指一个人希望在各种不同的情境中有实力，能够独立自主，它是人类的自尊。外部尊重是指一个人希望有地位、有威信、受到别人的尊重、和高度评价。当尊重得到满足之后，能够使人充满信心，对社会充满热情，体验到自己活着价值和用处。自我实现需求是最高层次的需求。该需求将个人理想、抱负、发挥到了个人能力的最大程度，完成了与自己能力相适应的一切事情需要。自我实现需求需要自己努力实现自己的潜力。五种需求从低到高呈阶梯状。一般来说，某一层次的需求相对满足了，就要向更高层次的需求发展，追求更高层次的需求成为驱使我们行为的动力。在五个层次中，生理需求、安全需求和社交需求是低层次需求，而尊重需求和自我实现需求是高级需求。马斯洛是一个人文心理学家，他的动机主要关注把人看做是一个初学者，并关注自身、情感、智能和审美需求等方面关系。

美国耶鲁大学的Clayton在马斯洛的层次需求理论基础之上，进行了更加实际的研究，提出了一种新的人本主义需求。Clayton指出，人们需要三种核心的需求，即生存需要，相互关系需要，成长需要[11]。因此，该理论又称为ERG理论。如图1.3所示：生存的需要与人们的基本物质生存有关，即生理和安全的需求，关系到人的存在或者生存，这实际上相当于马斯洛理论中的前两个需求。相互关系的需要，即人们对于保持重要关系的需求。这种社会 and 地位需要的满足建立在于其他需要相互作用中达成，与马斯洛的社会需要和自尊需要相对应。成长的需要表示个人谋求发展的内在愿望，它包括马斯洛自尊分类中的内在部分和自我实现层次中所包含的所有特征。除了用三种需求代替了马斯洛的五种需求之外，ERG与马斯洛的层次理论还是有区别的。ERG理论表明，人在同一时间可能不是一种需求在起作用，如果较高层次的需要得到抑制的话，那么人们会对低档次的需求渴望会变得强烈。马斯洛的需求层次是一种刚性的阶梯式上升结构，必须较低层次的需求必须在较高层次的需求满足之前得到充分的满足，二者关系不能打破。而相反，ERG理论不认为各类需求层次是刚性的，即这三种需求层次可以同时起作用。

Keller教授提出了一个激发与维持学生学习动机模型[53]。该模型关注如何通过教学来调动学生的动机。该模型中，认为影响学生的动机主要有：注意(attention)，贴切性(relevance)，自信心(confidence)和满足(satisfaction)。该模型的主旨是：为了激发学生的学习动机，首先要引起学生对一想学习任务或者是学习目的的兴趣；其次，要使

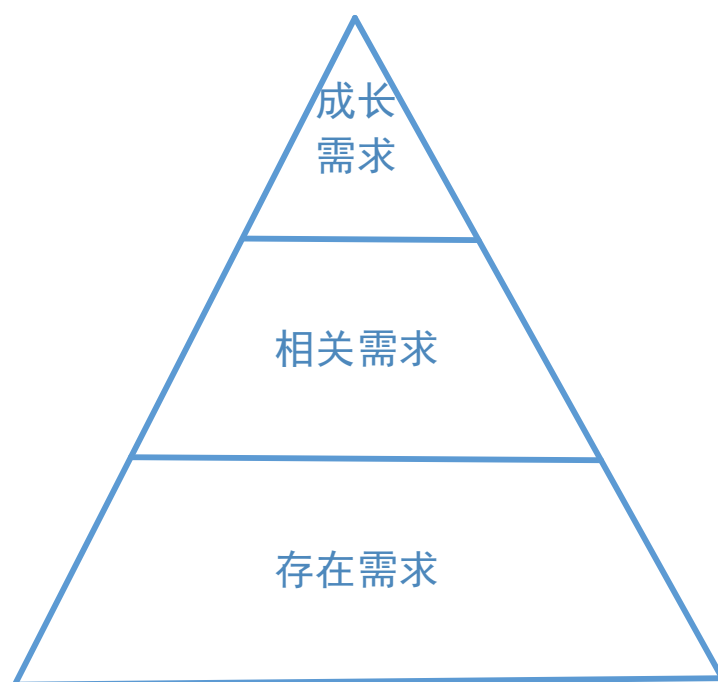


图 1.3: ERG理论模型

学生理解完成这项任务同自己相关；在次，需要学生相信自己可以完成任务。注意、贴切、自信和满足是一个统一的整体，没有主次之分，缺少任何一个元素，都可能会使学习者丧失学习动力和机会。因此，在教学是时候要考虑到这四个因素，忽视任何一个要素，到可能导致教学实施的无效或者是失败。ARCS动机模型是一个很具有操作性的动机体系，对于教学具有指导作用。

De Sevin等提出了一个虚拟角色的行为选择模型[30]。对于自治的虚拟人来说，动作选择架构需要自己决定、有动机性、反应性和主动性，从而成为一个高度自治的智能体。OMG等提出了一个商业动机模型[20]。该模型主要用是在变化的世界进行建模，帮助人们在商业决策中决策。对于该动机模型，商业公司可以应用该模型，输入自己公司的相关数据，计算机适合自己的决策。

Bach提出了一个框架，该框架基于心理学的研究[18]。目标通过与外部的环境进行交互，通过强化学习获得。Bach指出，在这个模型中，所有的行为都是为了实现智能体的目标，通过将动作作用于环境而实现自身的目标。Bach提出了一个三层需求模型。如图1.4所示：PSI中最底层的需求是生理需求，然后是认知需求，最后是社会需求。所有的PSI智能体都是由目标驱动的，都需要通过智能体对外部环境进行相应动作而实现。PSI的生理需求包括燃料和水，这两项作为自身的资源存在。因为智能体的每个动作都需要消耗自身资源。生理需求中还包括完整性，随着时间的推移，外部的环境

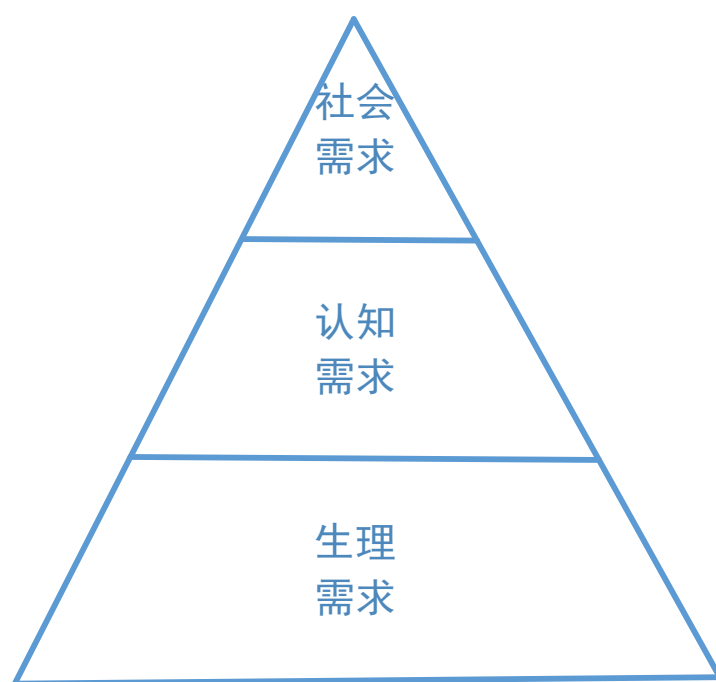


图 1.4: PSI 需求模型

会对智能体的造成损害，因此，要引起注意。当智能体处在未知的环境中时，需要对其周围的环境进行探索。存在于多智能体的环境中，智能体需要探测周围存在的智能体，从而使得自己有归属感觉。当智能体自身的能力不能完成复杂的问题的时候，还需要请求别的智能体来一起完成任务。

在多智能体环境下，多个智能体共同完成一个目标。为此，多个智能体也具有相同的动机，相同的需要。

### 1.3.2 典型认知模型

Soar[58][79]认知模型从20世纪80年代开始一直在维护，该认知模型由卡耐基梅隆大学的Allen Newell 和Paul Rosenbloom等共同开发。Soar是典型的符号系统[59][60][127]。Soar中程序性的长期记忆知识通过产生式的形式存储，然后以操作符的形式同问题的求解相联系。有些操作符的形式非常简单，这些原始的操作可以用来改变内部的状态也可以用来动作于外部的环境，另外的一些操作符就显得有些抽象。长期以来，Soar一直采用这种方式的知识管理方式，近期有所改变，添加了情景记忆和语义记忆。情景记忆用来存储历史的状态，而语义记忆用来存储原来知道的事实。如图1.5所示，该图是Soar最新的结构图。

Soar主要是基于目标、问题空间、状态和操作符理论对问题进行求解。在图1.5中，



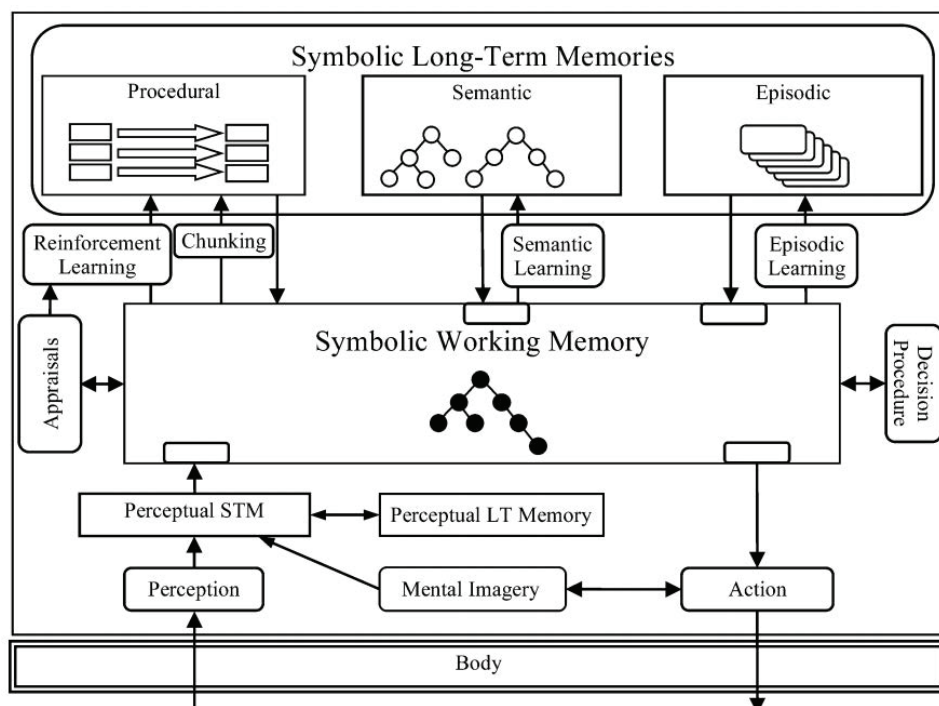


图 1.5: Soar认知模型

方角框表示模型的记忆模块，而圆角框表示系统的处理模块，箭头表示模块间的连接和信息的传输[58]。感知模块感知外部环境中的信息，并将这些信息存储在短期记忆存储中（Short Term Memory, STM）。符号工作记忆模块（Symbolic Working Memory）从短期记忆存储（STM）中抽取数据，并将其添加到Soar的工作记忆中。符号工作记忆模块作为模型的全局短期记忆模块从长期存储（Long Term Memory）中检索信息。知识存储在长期记忆模块中。长期记忆中的三个模块产生式模块（Procedural）、语义模块（Semantic）和情景模块（Episodic）是相互独立的，且每个模块具有自己独立的学习机制。

产生式模块内部含有规则，前一部分表示规则需要满足的条件，后一部分表示规则对应的动作。产生式规则可以修改工作记忆。系统根据喜好选择相应的产生式规则，一旦系统选定相应的规则，Soar系统就会根据规则产生相应的变化。当规则在工作记忆中执行的时候，其它的存储和处理程序在工作记忆中有自己的保留模块，可以监控规则的执行，且在执行的过程中可以对语义模块或者是情景模块进行初始化。在Soar模型中，同产生式模块相对的学习机制主要有，chunking和强化学习。Chunking学习机制可以产生新的产生式规则，而强化学习主要是可以更改规则的动作，生成选择的机制。语义存储记忆存储客观的事实，情景存储记忆是工作记忆的一个快照。产生式规则既可以更改外边的环境，也可以对产生式进行更改。

这些不同的存储记忆相互协同，共通过完成感知任务。在这个过程中，产生式规则提供了模型需要的控制信息，而工作记忆则成为了一个全局工作空间。Soar[58]最新的记忆系统如表1.1所示。

表 1.1: Soar认知记忆

记忆和学习系统	知识源	知识表现形式	检索知识
	子状态		
程序记忆 chunking	触发的 产生式 规则	产生式规则	精确匹配规则， 检索动作
程序记忆 强化学习	奖励	量化的选择指标	精确匹配规则， 检索动作
语义记忆	工作记忆	工作记忆中存 储结构的镜像	精确匹配线索， 检索对象
情景记忆	工作记忆	工作记忆 的快照	部分匹配线索， 检索对象
感知记忆	STM中的 感知信息	对视觉和 空间信息量化	根据符号 对应关系回忆

Soar中所有的规划都是为了达到系统的目标。系统通过自己内部的知识来规划和选择动作和操作符。Soar系统的处理过程为提出操作符，选择操作符，应用操作符，这个处理周期一直在重复，推动问题的求解。随着问题的求解，Soar会遇到僵局，即不知道下一步选择哪个操作符或者是系统选择了抽象操作符，但是不知道如何将抽象操作符具体化实现。为了解决系统产生的僵局，系统需要生成新的目标。即主要的目的是如何选择下一步需要执行的动作，或者是如何根据现有的知识将操作符具体化。当僵局问题解决后，系统继续问题的求解。

随着系统问题求解的进行，由于模型存在僵局，有时候僵局会迭代产生。因此，模型中的任务会进行层次化的分解。一个特定目标的状态包括其父目标，同时还包括认知结构，这个认知结构可以用来选择可以应用操作符。作为系统解决问题的一部分，子目标可以访问模型中的语义记忆和情景记忆，以加速问题的求解。最顶层的状态包括传感器从外部环境感知到的信息，对于所有这些信息，子目标都可以进行检索访问。在每一个认知周期，处在各个水平的目标都会进行动态的变化。有两种原因会导致系统的变化，一种是传感器感知到了外部环境的变化，另外一种是将操作应用到了

外部的环境。当系统的僵局解除，为了解除僵局而产生的目标消失，包含该目标所产生的子目标同时都会消失。

Soar对于不同的知识，具有不同的学习机制。chunking和强化学习可以获得程序性知识，情景和语义学习获得陈述性知识。当一个或者多个结果在子目标中生成的时候，chunking学习机制开始发生[61]。当这种情况发生的时候，Soar学习到了一条新的知识，然后将其作为条规则存储在系统当中。一个chunk的动作和结果相当，但是动作发生的条件和父目标相关，由父目标的条件来确定。当系统面对新的环境的时候，如果所处的状况和学习到的知识chunk中很类似，则学习到的知识会触发。强化学习可以对系统选择操作符具有非常重要的作用。

ACT-R (Adaptive Control of Thought-Rational) 是另外一个经典认知模型，它关注如何将认知进行模块化的分解并将这些分解的模块组合起来，以完成统一的认知[15][14]。该模型由卡内基梅隆大学的Robert Anderson 和Allen Newell领导开发。表面上看ACT-R像一种编程语言，实际上ACT-R的结构反应了人类对认知的假设。和Soar类似，ACT-R也是基于产生式规则的，但是ACT-R具有一个非常重要的特点，即ACT-R 允许实数的动作[13][63]。Soar是离散的二值动作。ACT-R更加的贴近生物特性，它的动作是和时间相关的，而不仅仅是程序步骤。

ACT-R是一个混合的认知系统，即他拥有符合系统和子符号系统。ACT模拟人的联想记忆，具有它的理论基础。ACT-R试图提供一种描述语言用来描述人的心智内容，这些具有层次化结构的内同组成了语义网络。ACT-R包含5个重要的组成部分，每个部分用来处理不同的知识。视觉模块用来确定物体的身份和位置；manual模块用来控制系统的动作；陈述模块用来检索长期存储记忆中的知识；目标模块用来在外问题求解的时候保持系统的内部状态；产生式系统用来协调其它模块的操作。

ACT-R系统具有2个瓶颈。系统中的每个缓冲中只允许存储一个陈述性知识。而像视觉信息等，同时会接受到多种信息，这样会造成系统的丢失。第二个瓶颈是在认知的过程中，系统每次只选择一条规则，与之相比，Soar 一次可以选择多条规则执行。ACT-R的视觉模块分为2个部分，每个部分都有自己的缓存。其中一个用来存储物体的位置，和背侧通路相连；另外一个缓存用来对物体进行识别，和腹侧通路连接。但是，由于背侧通路和腹侧通路的分离，现在受到了神经生理学家的质疑[89][88]。目标模块存储了系统当前环境下需要达到的目标，系统相关的动作都是为了实现该目标。事实上，它保证了系统中所用的动作的一致性，即都是为了完成特定的目标。

### 1.4 本文的主要贡献

本文的研究目标是研究协同认知的基本问题，建立认知的动机模型，并在此基础上

上对多智能体的协同认知进行研究。

本文的主要贡献包括如下几个方面：

### (1) 提出NGI动机模型

动机是一种非常复杂的现象，根据我们承担的具体课题，我们提出了一个动机模型。N表示Need(需求)，G表示Goal(目标)，I表示Intensity(强度)。智能体不断的同外界进行交互，在交互的过程中会产生各种动机。为了对动机更好的进行建模，我们将动机分为感知动机、适应性动机和合作动机。动机的生成需要目标来解决，因此，我们的动机模型中确定了目标。不同动机具有不同的重要性，为此，需要对动机的强度进行标注。

### (2) 基于智能体的ABGP协同认知

其中A表示Awareness（感知），B表示Belief（信念），G表示目标（Goal），P表示Plan（规划）。建立动机模型，并对生成的目标进行规划。智能体所处的环境进行着动态的变化，为此，我们需要对其所处的外部和内部环境进行感知，将信息及时传输到智能体。一个目标可能对应这多个动机，因此，我们需要按照动机的优先级别生成目标。目标生成后，选择优先级别最高的进行执行。对于已生成的目标，包括其初始状态和最终状态。智能体从其情景记忆中选择动作序列，进行动作的规划。

### (3) 提出基于智能体信息共享的协同认知

该方法通过采用信息桥接方式，打破了消息传输的二义性，方便了智能体之间信息的传输和检索。桥规则增强了知识库之间的互操作，使得信息可以按照我们预想的方式传输。在进行推理的时候，首先在智能体内部进行推理；在不能得出推理结果的时候，选择同时具有into和onto桥规的的候选智能体进行推理；如果不能得出结果，再选择三个智能体，其中一个只有into桥规则连接，一个只有onto桥规则连接，然后在选择一个智能体同时和这两个有智能体有连接，进行推理。

## 1.5 论文的组织

本文主要章节的内容安排如下：

第一章是引言，主要是介绍本课题的研究背景和意义，以及基于动机的认知模型的研究内容，阐述论文的主要贡献与章节的安排。

第二章介绍了动机模型。动机的概念来自于心理学。动机是使得智能体产生目标的内部原因，且动机的来源有内部和外部。在本文中，提出了NGI动机模型，N即需求，G是目标，I是意图。智能体中每个生成的动机都尤其自身的属性权重，一个或者多个动机对应一个目标。动机生成时候尤其原始权重，根据算法生成目标权重。

第三章介绍了基于智能体ABGP的协同认知。在多智能体环境中，每个智能体都在同其他的智能体进行协同交互。以往的文献中，主要关注智能体之间的协同，而对外部环境的关注不够。为此，我们提出了新的认知模型ABGP，A即环境感知，B是信念；G是目标；P是规划。在ABGP认知模型中，我们突出了环境对智能体的影响。接下来对目标进行分解，分解为一个个的小目标。最后，对每个小目标分别经进行规划，从而完成目标。

第四章介绍了基于智能体信息共享的协同认知。提出了CD3L算，该算法首先判断能否在一个DDL系统内部进行推理，然后在判断能否在两个D3L系统间进行推理，最后通过CD3L在多个D3L系统中进行推理。与D3L相比，新提出的CD3L弥补了传统桥规则描述逻辑系统的不足。

第五章介绍了协同认知原型系统。基于上述研究，设计了一个基于多智能体环境的MAGER的系统。为了使得系统更加的具有灵活性，将智能体分为多个类型，不同的类型具有不同的功能。并且，我们详述了系统实现的主要技术。

第六章对全文的主要内容进行了总结，同时对下一步需要进行的研究进行了展望。

所有章节的内容组织结构如图1.6所示。

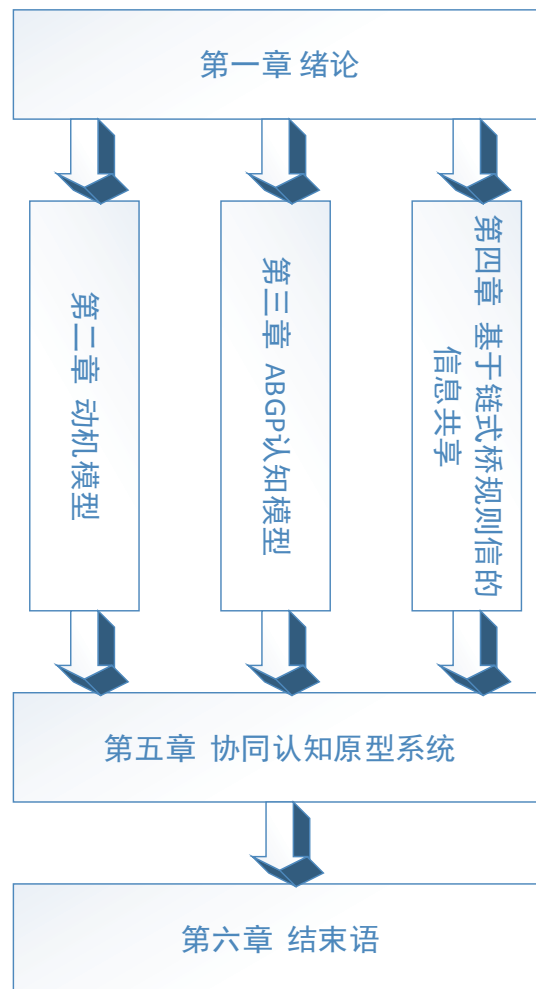


图 1.6: 本文的内容组织结构图

## 第二章 基于动机的协同认知

### 2.1 引言

对智能主体的研究，自二十世纪七十年代以来，一直是研究的热点，成为人工智能主要的研究方向之一。随着研究的不断深化，不同的研究人员对其有着不同的研究侧重点。在研究的早期，有的侧重从理论和逻辑的方面研究，有的侧重从应用、动作的角度研究。动态描述逻辑将静态和动态的知识表示与推理有机的结合在一起，形成了统一的形式化框架，极大的方便了我们构建智能主体。

智能主体在于外界环境交互的过程中，会接受到各种信息。按照信息不同的来源途径，主要有外部环境感知到的信息，智能体之间传输的信息和智能体内部随着状态的变化所产生的信息。智能体需要对这些信息进行分析，然后按照其重要程度，进行区别处理。处理不同的信息，需要对应不同的方式，这就涉及到了动机。动机作用于智能体或者是智能体内部，并指引其行为，维持智能体朝向某一目标。从某种意义上来说，所有的智能系统都是目标驱动的，即在这种系统中，所有有的服务都具有一定的目标性。智能体应该具备管理动机的功能。因此，本章我们对其进行建立模型并分析。

智能体的一个典型的行为是可以实现自己的目标。现在许多的理论、系统和科研项目专注于实现一个具体的目标，但是对目标的产生、选择和实现研究的还远远不够。

目标是为了解决智能体所产生的动机，为此，对动机的建模就非常重要。我们需要对动机进行形式化表示，从而，使得我们可以方便的用计算机进行处理。

通常我们认为，具有目标性的行为是自治的智能体显著特性之一[38]。这就意味着，当我们的智能体处在动态变化的环境中时，需要根据环境的变化采取适当的动作，这样才能更好的适应环境从而满足自己的需求。我们所说的环境，既包括智能体接触的外部自然环境，也包括多个智能体之间的交互信息，还包括智能体内部状态所产生的信息。现在，人们开发了多种系统用来处理这些事情。

现在人们所开发的系统中，对于单一的问题，系统可以高效的解决，但是系统的通用性差。为此，我们又有必要开发更加通用的系统，以更好的适应环境。因此，随之时间的推移，在人工智能系统中，产生了一个新的研究方向，设计更加通用的系统。但是，令我们感到震惊的是，对于系统来说，最初的动机如何生成，如何表示，我们给予的关注还是很不够的，有时候我们甚至假设这些动机是来自于其他的子系统。由于



我们现在设计的系统大多数是解决单一的功能，因此，人们需要开发可以同时支持多种功能的智能系统。在这样的背景下，动机的生成、管理、和选择就成为了一个非常重要的问题。我们最感兴趣的是研究智能体生成自己的动机的机制，从而满足智能体的需求。鉴于这个机制会包含许多种不同的状态和过程，我成对其称之为动机模型。

## 2.2 动机模型

一个动机可以表示为一个三元组 $\langle N, G, I \rangle$ ，其中N表示need，即需求；G表示goal，即目标；I表示intensity，强度。

对于一个动机的被选中，需要满足

$$R = (P, D, Strength(P|D))$$

，其中R表示实现动机的条件，D表示实现动机的动作集合，Strength(P|D)的取值为[0,1]。

在当前的情况下，我们的系统计划应用与老鼠机器人上，这个机器人是脑机结合的系统。在该系统中，我们机器人的生存环境是确定的。因此，该系统中的系统需求是确定的。根据脑机系统的自身特点和需求，我们提出了3中不同的需求：

- **感知需求：**智能体通过自身传感器进行环境的信息感知，这些传感器包括视觉、声觉、触觉、味觉和表情等。感知需求主要感知的是外界的自然环境，这些信息大多需要借助传感器的帮助达到智能体的内部。
- **适应性需求：**适应性需求主要是智能体为了适应外界自然条件和优化动作对环境的影响。简单来说，当智能体通过动作作用于外部的环境的时候，会使得外部的环境进行改变。因而，智能体对此需要进行慎重的考量，即通过改变外部的环境，使得改变后的环境更加的适应自己。
- **合作需求：**对于由大脑和机器组成的智能体来说，在大脑和机器之间需要有一个报酬函数。这个函数是的动作产生后，经过长时间来说，该报酬函数的值最大。

动机具有其自身的特性，在我们的计算机系统中，有的动机是短暂的、临时的，即动机生存的时间是相对少的时间。然而，在某些情况下，动机会一直存在。某种意义上来说，动机一旦生成，会一直在系统中存在，一直到系统宕机为止。

系统处于动态变化的环境之中，为了对需求进行相应，系统就产生了各种各样的动机。每个目标都有其自身的属性，有的目标较大，有的目标较小。较大的目标不容易实现，较小的目标则比较容易的实现。当目标产生后，还涉及到目标的选择问题。



不同的系统具有不同的选择方式，有的系统可以并行执行，而有的系统只能并发执行，甚至有的系统每次只能够选择一个动机进行执行。在时间 $t$ 时刻，每个目标均由多个子目标构成，即：

$$G_t = (G_1^t, G_2^t, \dots, G_n^t)$$

**定义 2.1 (层次目标)** 我们的目标由子目标构成，子目标是有层次性的，即不同的子目标之间执行就有先后顺序。子目标之间具有关联性，为此，我们的目标可以表示成为一个有向无环图(DAG)三元组，即：

$$DAG = (P, E, \prec)$$

在这里面有：

1.  $P$ ：表示所有的子目标节点，每个子子目标都表示为一个节点，所有的子目标在一起构成了我们的大目标；
2.  $E$ ：表示节点之间的联系，子目标和子目标之间的联系。不同的子目标之间的执行具有先后关系，为此，我们需要将其进有效的组合，以满足我们的需要；
3.  $\prec$ ：表示偏序结构。即我们的节点之间是具有先后关系的，所以，系统在执行的过程中要严格按照指定的顺序进行执行，否者系统不能够达到既定的状态。

对于目标，我们应该满足下面的条件约束：

- 每个目标都有一个特殊的TopGoal节点，这个节点是整个目标的根节点，且该 $TopGoal \in P, \forall p \in P, p \neq TopGoal$  and  $p \prec TopGoal$
- 如果 $\langle p_1, p_2 \rangle \in E$ ，则 $p_2 \neq p_1$ ；
- 如果 $\langle p_1, p_2 \rangle \in E$ ，则 $p_2 \prec p_1$ ；
- 如果 $p \in P$ ， $\exists p_1, p_2, \dots, p_n \in P$ 且 $\langle p_1, p_2 \rangle \in E, \langle p_2, p_3 \rangle \in E, \dots, \langle p_{n-1}, p_n \rangle \in E$ ，则有 $\langle p, p_2 \rangle \notin E, \langle p, p_3 \rangle \notin E, \dots, \langle p, p_n \rangle \notin E$

假设在 $t$ 时刻我们的目标有：

$$G_t = \{G_1^t, G_2^t, \dots, G_n^t\}$$

目标 $G_i^t = \{G_1^t, G_2^t, \dots, G_n^t\}$ ，给定 $G_i^t, 1 \leq i \leq n$ ，满足 $G_{i_r} \neq G_{i_s}$ ，即子目标中不存在重复的目标。

层次化目标具有其自身的有点，主要是表示的时候方便直观。但是层次化目标对于现实生活中某型目标的表示会失效，虽然这种情况下的目标依然可以进行分解。这种目标中的干扰因素就是时间。非常明确，目标由一个个的子目标构成，但是子目标中的完成是具有时序要求的，并不是具有组合的性质。这类似于我们的数字电路中的时序逻辑。而层次化目标表示类似于数字电路中的组合逻辑。

**定义 2.2 (目标序列)** 我们的目标由子目标构成，子目标是有层次性的，即不同的子目标之间执行就有先后顺序。子目标之间具有关联性，为此，我们的目标是是一个时序序列 $\{G_1, G_2, G_3, G_4\}$

目标是由一个个的小目标组成，且小目标之间具有时序结构。但是目标是没有层次化结构的。层次化结构智能解决某些特定情况下的类型，对于很多具体的目标表示失效。

对于目标序列这种表示形式，这就要求我们在确定目标之后，对目标进行分解。不但要求，我们将大目标分解为小目标，还要求我们在分解的时候保持其时序关系。在实际的目标表示中，对于同一个目标，其实现形式并不唯一。我们只要找到其中的一中解决方案即可。

## 2.3 动机系统

动机是一种非常复杂的现象。已经有多中理论试图从各种方面对动机进行解释和建模，从而方便计算机的处理。在这里，我们提出动机系统，该系统主要包含七部分。

1. **环境：**环境用来向智能体提供外部的环境信息。这些外部信息通过智能体自身携带的传感器进行感知，如声觉、听觉、视觉等，还有一类的信息在智能体之间进行传输。
2. **内部环境：**内部环境表示智能体内部状态的平衡信息，并且整体内部的环境随着智能体对外部的环境的改变而在进行着不断的变化。
3. **动机：**动机是一个抽象的概念，随着环境中信息的变化，智能体会根据内部的控制信息动态的维持平衡。智能体所产生的各种动机本质上是为了维持系统的一个平衡状态。
4. **动机库：**该库中存放许多不同形式的动机模式。当智能体感觉到外部情况变化的时候，智能体会成相应的动机。在动机生成的过程中，如果有动机库的帮助，则十分有利于动机的形成。

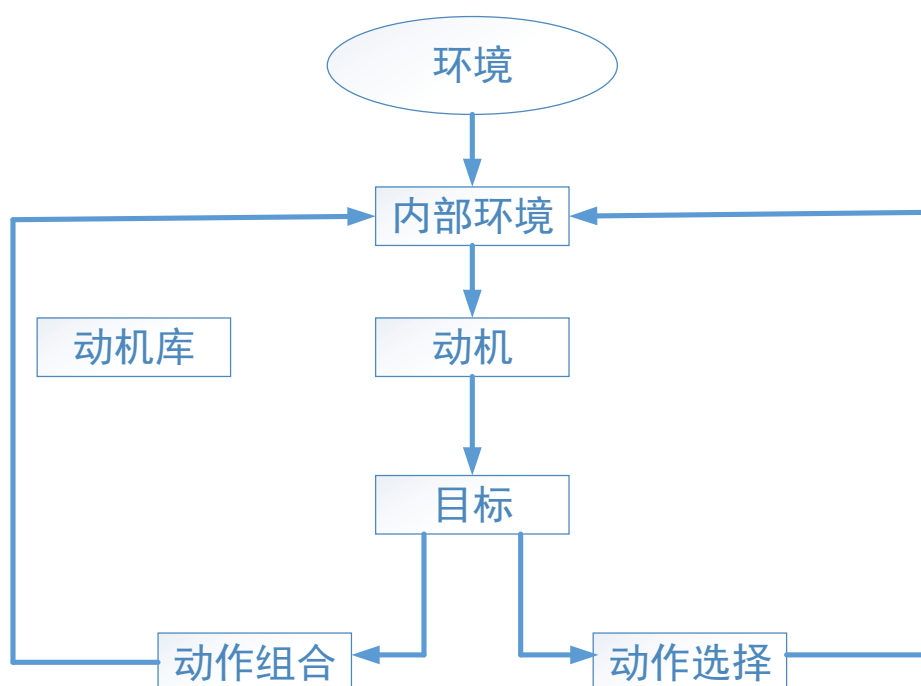


图 2.1: 动机系统

5. **目标：**目标是个人或者是组织所期望的结果。对于我们所期望的结果，我们需要一系列的动作来完成。这些动作是连续的，一个动作接着一个动作。
6. **动作选择：**动作选择用来从动作库中选择相应的动作来满足动作规划用。完成一个动机通常具很多种实现方案，具有相似功能的动作放在同一个动作集合中。当我们在进行动作规划的时候，从中选取相应的动作以进行组合。
7. **动作组合：**当我们的目标选定之后，随之需要考虑的就是完成目标。在这个过程中，最关键的一步是动作的组合。动作组合是我们这个认知过程中的重中之重。所有的一切，都要围绕这动作组合来完成。

动机系统，如图2.1所示：

## 2.4 动机执行流程

CAM心智模型中的短期记忆系统用来实现我们本章所述的动机系统。在短期记忆系统中，主要包含两个子模块，信念记忆模块和目标/意图记忆模块。短期记忆系统不但实现了BDI(belief desire intention)模型，并且实现了一个实际的推理程序[86] [76]。BDI模型受到逻辑和心理学的启发，其中最关键的技术就是使用逻辑符号构建智能体的信念、愿望和意图。

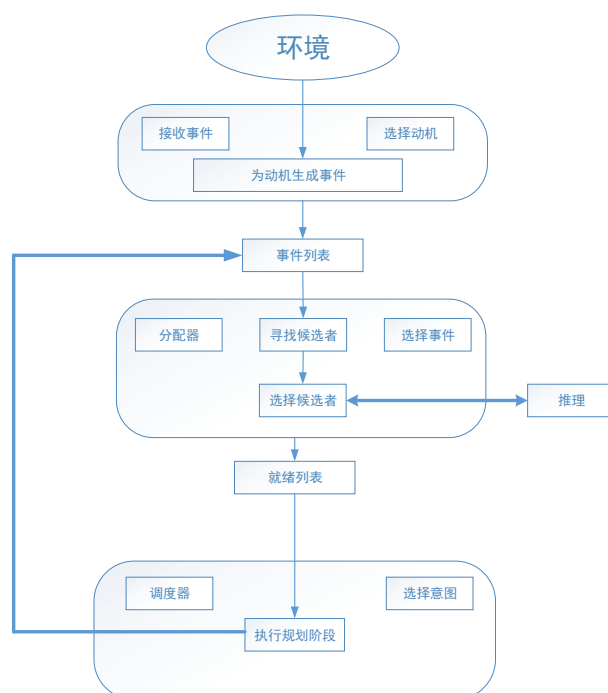


图 2.2: 动机执行流程

信念是智能体对客观世界的认识，即外界的客观世界在智能体内部的表示。智能体可以通过传感器或者是经验获得这些信息。在CAM系统中，信念记忆存储智能体当前的信念，包含了动机知识。

愿望是目标或者说是期望的最终状态。意图是智能体选择的需要现在执行的目标。意图是众多目标中的一个，且是认为当前情况下最应该需要执行的目标，通过意图的执行，可以极大的改善智能体的当前所处环境。目标/意图记忆模块存储当前的目标和意图信息。由于目标的多种多样，所以，它们都可以被选择执行，但是在同一时刻，只能有一个目标在执行。在CAM中，目标是由子目标组成的有向无环图，所以，在执行的时候，也是分步执行的。一个个子目标按照有向无环图图所表示的样子来完成，当所有的子目标都完成之后，总目标完成。

对于一个动机执行系统来说，最关键的就是智能体内部的规划部分。通过规划，每个子目标通过一些列的动作来完成，从而，最终实现我们所希望看到的是任务。规划对于已经选择的任务没有特殊的作用，规划主要是用来处理内部的信息和系统新产生的动机。为了收集这些新产生的动机并且将它们送到动机选择部件，我们需要有一个分发模块。同时，我们还需要一个模块，这个模块用来执行我们的规划。如图2.2所示：该图明确表示了动机的执行流程。事件的接收器从环境中收信息并将信息放置到

动机的列表中。分发模块不断的从事件列表中抽取事件，并为之提供相应的规划。当目标到达的时候，检测是现有的环境是否满足目标实现的条件。在这个过程中，需要对动作进行规划。每个动作都尤其成立的前提条件和结果条件，为此，我们都要对其进行检测。

接下来，分发器选择相应的规划用来完成选的目标，这个过程中需要推理机的配合。这就意味着，系统会找到一个或者多个我们在以前人为制定的方案。当我们在对现有的目标进行推理的时候，我们有可能找到的并不是唯一的解决方案。这时候，我们的推理机需要按照其内部的规则进行选择，选择的标准需要我们事先认为指定。不同的选择标准，会导致智能体在进行决策的时候，有不同的表现行为。选择好的规划方案后，系统会将需要事先的目标和方案联系起来。这将使得规划对目标有详细的了解，使得在目标规划的时候，有充足的信息可以利用。

调度程序在目标列表中选择需要执行的目标。在目标执行的过程中，每个目标是不可剥夺的，即当一个目标执行的时候，不能够剥夺其执行的权力而执行其它的目标。

考虑到动机的双重特性，即有的动机是显式的，有的动机是隐式的，这样，动机的进程就不是那么简答了。事实上，隐式的动机是更加重要的，因此，系统在执行的时候，应该首先考虑挑选隐式的动机执行。这里我们考虑的主要是显式的动机，并且假设显示的动机代表了智能体大多数的显式的目标。显式的动机提供了确定和实际的动作执行序列，显式的目标同时也允许更多宽松的行为和形式。

### 2.4.1 目标生成

在一个认知系统中，目标的实现蕴含了动机的实现。只要我们的目标可以实现智能体中所产生的动机，该目标就会进入我的事件队列。

智能体处在外部的自然环境中，如前面章节所述，其会接触到来在自然界的信息、来自其他智能体的信息和来自自身的信息。随着这些信息的产生，智能体内部中会产生各种动机。在智能体中，一个目标中往往包含有多个动机，为此，我们需要对目标的重要程度进行衡量。该评分标准中，动机具有非常重要的参与价值。

智能体中每时每刻都在产生着新的目标，但由于智能体自身能力的限制，智能选择少数的目标进行执行。因此，对目标的重要程度进行衡量就显得尤为重要。对于某些紧急情况下产生的动机而产生的目标，我们给给予最高级别的优先级别。目标计算完毕之后，直接将该目标放置到整个执行队列的适当位置。对于非常紧急的任务，甚至可以中断当前执行的目标，紧急执行新生成的目标。据此分析，我们在智能体ABGP模型中提出了算法2。该算法综合考虑动机的重要性，确定了目标的优先级。

当动机在智能体内部生成的时候，每个动机都有两个属性，该属性有智能体在设计的时候指定。动机的表示形式为 $(N, Goal, URG, \lambda)$ 。"URG"属性表示动机是否非常的紧急，只有当动机非常紧急的时候才将该位置1，否则，将该位置0。 $\lambda$ 属性用来表示动机的权值，该权值的初值由智能体设定。不同的 $\lambda$ 值对动机的执行的优先级别有重要的影响，因此，需要系统的设计者需要根据以往的历史数据进行实际的设计。 $\lambda$ 的取值范围为 $[0,1]$ 。只有根据历史数据，我们的设计才能更好的贴近我们真实的世界。智能体的动机产生后，会有响应的目标来完成动机。在这样的情况下，会有一个目标对应多个动机的情况。当系统要生成目标权重的时候，需要将具有相同目标的动机进行聚合。

算法1的作用是对动机集合中的数据进行预处理。算法中的 $Q_{order}$ 用来存储按照排序规则排序好的队列。在这里面，我们的排序规则分为两个方面。首先，在全局的动机进行扫描，查看其“URG”的情况。“URG”显示了该动机的紧急情况，当“URG”的值为1的时候，表示现在的动机非常紧急，需要智能体对当前的动机马上进行反应，否则，会对智能体造成比较大的损害。当含有“URG”为1的动机扫描结束后，将 $Q_{order}$ 的头队列的值取出来，对“URG”的值进行检测。当“URG”的值为1的时候，说明系统中存在非常紧急的动机，此时，算法终止，直接将 $Q_{order}$ 队列返回，无需要在对剩余动机进行排序。

如果检测到系统中没有含有“URG”为1的动机，则说明现有动机是一般优先级的，我们需要按照我们的算法对进行下一步的计算。如果经检测，所有动机的“URG”均为0，则按照 $\lambda$ 的大小进行降序排列。将排序的结果存储到 $Q_{order}$ 队列中。但是此时，我们的任务没有完成。在此，我们采用贪婪算法选择我们的动机。按照动机在队列 $Q_{order}$ 中的顺序，依次出队，从而我们每次选择的都是 $\lambda$ 值最大的动机。在动机选择完毕结束之后，我们对所选择的动机与现有的动机进行比较。对于第一个动机，我们不必进行判断，直接将其存储到 $Q_{valid}$ 队列中。从第二个动机开始，用来判断新的动机是否被现在动机蕴含或者是矛盾。该判断由“ $(\neg(\text{Conj}(M_i \rightarrow \alpha)) \text{unsatisfiable}) \text{ or } (\neg\alpha \in M_i)$ ”进行检测。目标在执行的时候，如果有蕴含的，表明系统中存在重复的动机，对们计算权重有影响。当动机中存在矛盾的时候，说明该动机是不能够被执行的，因此，要将此类动机剔除掉。我们采用的策略是剔除权重低的动机，这样，在 $Q_{valid}$ 中始终保存着高权重的动机。将所有符合要求的动机存储到 $Q_{valid}$ 中，最后将 $Q_{valid}$ 返回。 $Q_{valid}$ 中存储的是经过预处理的所有有效的动机。

经过数据的预处理， $Q_{valid}$ 中按顺序存储了不重复，不冲突的动机集合。接下来，我们开始进行目标权值的计算。首先，我们将 $Q_{valid}$ 对首元素出队，进接着判断其“URG”位的值。如果“URG”值为1，则此时，可以判定此动机是非常紧急的，我们直接返回1。如果检测到系统中不含有紧急的动机，则继续对目标的权值进行计算。因

为在系统中，权值高的动机具有更高的优先级。因此，我们首先确定 $Q_{valid}$ 队列中动机的个数，即 $|Q_{valid}|$ 。该值进行动态变化，是我们每个动机本身的一个权值。该权值表示该动机的排位靠前，目标的执行对其具有更大的影响。

当我们的目标执行的时候，不是动机中的所有的元素都能够满足，我们只能满足多数的动机。为此，我们对目标中可以满足的动机元素进行计数。在此，我们设置三个临时变量，以方便我们的计算。变量“eff”表示目标的执行，能有对系统的动机具有正面作用的计算结果。变量“total”表示智能体内所有的目标的加权值计算结果。变量“effcnt”是一个用来临时进行计算的变量，该变量的结果主要是用来计算目标执行后，对单个动机的影响进行计数。在对每个动机分析结束后，用“effcnt”除以“ $M_i$ ”，得出目标执行后可以满足动机中元素的概率。然后再乘以 $\lambda$ 和 $NUM$ 。 $\lambda$ 的值由动机在产生的最初智能体进行标识，而“NUM”是目标所支持的动机的个数。通过“NUM”我们将优先级进一步的加强。最后，将“eff”除以“total”计算出目标的权重。

智能体内部不断的对每个目标进行着计算。当目标的权值计算好后，将其放入目标执行队列。所有的目标按照优先级在对列中进行排序。算法中主要进行了两部分的工作，一部分是动机缓冲区的更新，一个是目标权重的计算。所以，这是一个复杂的问题。

## 2.5 本章小结

在本章中，我们主要介绍了智能体在复杂的环境中产生了各种动机，即目标，智能体的管理模型。动机是一个复杂的现象，并且在智能体中具有非常重要的作用。在动机模型中，我们主要是对动机的管理。包括目标的生成，目标的执行。智能体中目标一个个的执行，不断的将动作作用于外部的自然环境。

动机模型是我们智能体目标管理的基础，在智能体结构中具有举足轻重的地位。动机是智能体在产生目标之前自身的需求，我们需要对动机本身进行建模，进行形式化的表示。当动机的模型建立之后，处在自然环境中的智能体就会源源不断的在体内产生各种各样的动机。智能体中会有多种动机进行存在，但是当我们由动机生成目标的时候，一个目标可能对应着多个动机，因此，对于目标的生成，动机起着至关重要的作用。每个动机都尤其自身的权重和目标，为此，我们需要将所有的具有相同目标的动机进行统一规划，从而生成目标，生成后的目标进入存储队列。

我们的动机模型将应用于老鼠机器人中。在老鼠机器人中，老鼠处在外部的自然环境中，其大脑内部中的电极会将老鼠大脑的脑电波信号转化成计算机可以处理的电信号，从而计算机获得了老鼠处在自然环境中所产生的动机。

---

**Algorithm 1** Valid Motivation.

---

**Input:**

motivation seques:  $M = \{(N_1, Goal, URG, \lambda_1), (N_2, Goal, URG, \lambda_2), \dots, (N_n, Goal, URG, \lambda_n)\}$ ;

Goal:  $G$ ;

**Output:**

Valid motivation Queue:  $Q$ ;

- 1: Initial queue  $Q_{order}$  empty;
  - 2: Initial temporary motivation  $M_{tem}$  empty;
  - 3: Initial queue  $Q_{valid}$  empty;
  - 4: Initial add;
  - 5: Check the value of URG, if there is a motivation's UGR equals 1, then put it to the first of the queue  $Q_{order}$ ,
  - 6:  $M_{tem} := \text{Head}(Q_{order})$  ;
  - 7: **if** ( $M_{tem}.URG == 1$ ) **then**
  - 8:     return  $Q_{order}$ ;
  - 9: **end if**
  - 10: According the value of  $\lambda$  sort the motivations ordered by des ;
  - 11: //Dequeue the head of the queue  $Q_{order}$  and assign it to  $M_{tem}$
  - 12:  $M_{tem} := \text{Dequeue}(Q_{order})$  ;
  - 13:  $Q_{valid} \leftarrow M_{tem}$ ;
  - 14: **while** ( $Q_{valid}$  non-empty) **do**
  - 15:      $M_{tem} := \text{Dequeue}(Q_{order})$ ;
  - 16:     add:=true;
  - 17:     **for** ( $\alpha \in M_{tem}$ ) **do**
  - 18:         **if** ( $\neg(\text{Conj}(M_i \rightarrow \alpha))$  unsatisfiable) or ( $\neg\alpha \in M_i$ ) **then**
  - 19:             add := FALSE;
  - 20:             break;
  - 21:         **end if**
  - 22:     **end for**
  - 23:     **if** (add == TRUE) **then**
  - 24:         Inqueue  $M_{tem}$  to  $Q_{valid}$ ;
  - 25:     **end if**
  - 26: **end while**
  - 27: **return**  $Q_{valid}$ ;
-



---

**Algorithm 2** GoalWeight.

---

**Input:**

motivation seques:  $M = \{(N_1, Goal, URG, \lambda_1), (N_2, Goal, URG, \lambda_2), \dots, (N_n, Goal, URG, \lambda_n)\}$ ;

Goal:  $G$

**Output:**

Goal Weight:  $W$ ;

1:  $Q_{valid} := \text{Valid Motivation}(M, G)$ ;

2:  $M_{tem} := \text{Head}(Q_{valid})$ ;

3: **if** ( $M_{tem}.URG == 1$ ) **then**

4:   return 1;

5: **end if**

6:  $\text{Num} = |Q_{valid}|$ ;

7:  $\text{eff} := 0$ ;

8:  $\text{total} := 0$ ;

9: **for** ( $M_i \in Q_{valid}$ ) **do**

10:    $\text{effcnt} := 0$ ;

11:   **for** ( $\alpha \in M_i$ ) **do**

12:     **if**  $\neg(\text{Conj}(G) \rightarrow \alpha)$  unsatisfiable **then**

13:        $\text{effcnt} := \text{effcnt} + 1$ ;

14:     **end if**

15:   **end for**

16: **end for**

17:  $\text{eff} := \text{eff} + \frac{\text{effcnt}}{|M_i|} * \lambda_i * \text{NUM}$ ;

18:  $\text{total} := \text{total} + \lambda_i * \text{NUM}$ ;

19:  $\text{NUM} -$ ;

20: **return**  $\frac{\text{eff}}{\text{total}}$ ;

---



## 第三章 基于智能体ABGP协同认知

### 3.1 概述

上一章我们主要对智能体的动机部分进行感知。本章我们着重对认知模型进行描述。ABGP(Awareness-Belief-Goal-Plan)由中科院计算所史忠植研究员提出的认知模型[100]。由于单个的智能体自身能力有限，对于复杂的任务，单个的智能体自身能力并不能够胜任，需要多个智能体协同以共同完成任务。智能体在工作的过程总，需要同外部的环境进行交互。在智能体协同工作的时候，智能体不但要同别的智能体相进行信息的交换，同时要环境也会对智能体的动作和认知产生影响。基于ABGP的认知功能为实现基于ABGP的认知功能提供了必要的认知基础。因此，ABGP认知模型是一个非常重要的研究问题。

在多智能体环境中，对于复杂的任务，所有的智能体需要进行分工合作，从而，最终完成一个复杂的任务。每个智能体的目标确定之后，需要根据自身的信息进行规划，进行动作的组合。这些信息既包括内部的信息也包括外部的信息。在推理结束后，将这些动作序列作用于外部的环境。

### 3.2 心智模型CAM总体结构

认知研究已经成为当今世界大国重点关注的领域之一。一个普遍的看法是认知科学将在科学技术、医学发展、经济增长等多方面产生革命性的影响。在这个过程中，认知模型的建立对我们研究具有非常重要的意义。

CAM(Consciousness And Memory Model)是有中科院计算所史忠植研究员提出的心智模型[99]。图3.1为CAM认知模型的总体架构。CAM主要主要有意识、记忆和高级认知功能更三大部分组成。在智能体认知的过程中，每个部分都发挥了自己的作用，各个部分协同使得智能体完成认知的过程。记忆部分负责存储认知信息。这些认知信息有短期认知信息和长期认知信息。意识主要有两个作用，一是协调意识内部各个部分的相关工作，二是协调各个记忆组件。智能体的高级认知功能建立在意识和记忆的基础之上。高级认知包含好多例子，如图像轨迹检测、图像的语义理解，自然语言理解等。

记忆在人类的心智中具有具足轻重的地位。记忆是我们人类大脑中非常重要的组成部分，它存储这智能体所经历的历史信息，同时也存储智能体现阶段感知到的信息，这些信息可以在智能体的决策过程作为系统中的知识。随之时间的推移，智能

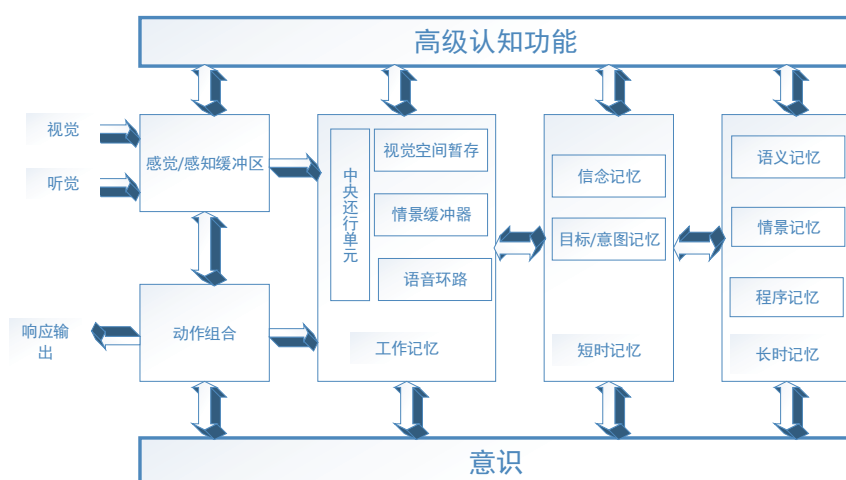


图 3.1: 心智模型CAM总体结构

体的知识会不断的进行积累。在这个个过程中，也会有新的知识不断的产生。下面，对CAM中的主要记忆部件进行简单的介绍：

- **长期记忆(Long-term memory):** 从图中我们可以清晰的看出，长时记忆主要包含了三个模块，语义记忆模块(Semantic Memory)、情景记忆模块(Episodic Memory)和程序记忆(Procedural Memory)。这三个个模块存储的知识都是具有较长时间的历史信息。其中，语义记忆存储的是过去的知识以及他们之间的联系；而情景记忆存储的是过去历史事件所发生的场景，语义记忆比情景记忆较少受到干扰，抽取信息也更迅速，往往不需要搜索。程序性记忆是记忆做事的方法。它被用于获得、保持和使用知觉的、认知的和运动的技能。程序性记忆理论大部分通常关注的是学习的时间进程。
- **短期记忆(Short-term memory):** 短期记忆存储主要包含了信念(Belief)、目标/意图记忆(Goal/Intention)。短期记忆用来暂时对信息进行存储。信念记忆存储的是智能体对世界的认知，包含描述环境特性的数据和描述自身功能的数据，是智能体进行思维活动的基础。目标/意图记忆存储智能体希望达到的状态或希望保持的状态。
- **工作记忆(working memory):** 工作记忆对于学习、推理、理解等高级的认知活动有着重要的作用。中央执行系统是工作记忆的核心，负责各子系统之间以及它们

与长时记忆的联系、资源的协调和策略的选择等。视觉空间画板主要负责储存和加工视觉空间信息，语音回路主要负责语音信息的存储于加工。情景缓冲器容量有限，它可以暂时存储多种信息的编码。该存储器在工作的时候，受到中央执行系统的控制。中央执行系统可以从中检索信息，当发现信息有错误的时候还可以对存储的信息进行修正。

CAM系统中的存储部件各自发挥自己的作用，共同实现模型的认知功能。我们的工作给予动态描述逻辑(DDL)。动态描述逻辑包含TBox, ABox和ActBox。ABox是关于个体的断言，主要用来描述论域中个体性知识；TBox是关于概念术语的断言，存储了外部世界的分类知识，ActBox存储了动作的相关知识，智能体在于外部交互的工程中，通过动作作用于外部环境。

1. 对于语义记忆，它是一种记忆形态。其典型的特征是：透过对于一般知识的事实与概念的了解，通过语言、文字、数学、算法等抽象的了解来形成记忆。它通常由情节记忆发展而来，是一种客观性的知识。
2. 情景记忆是指对智能体亲身经历的、发生在一定时间和地点的事件（情景的记忆。情景记忆涉及智能体生活中的特定事件，它与接收和保持的信息。
3. 程序性记忆是持久性记忆，关于如何做某事或关于刺激和反应之间联系的知识。程序性知识经常难以用语言来描述。因此，用TBox表示程序性记忆的存储关系。
4. 信念记忆存储了系统对于外部客观世界的认识，这同动态描述逻辑中的ABox类似，因此，采用Abox对其进行建模表示。在目标/意图记忆中，我们采用 $(S_i, S_g)$ 形式存储。其中 $S_i$ 表示目标/意图的原始状态， $S_g$ 表示目标的最终状态。
5. 传感感知缓冲区中存储的智能体感知到外部环境的变化。智能体感知到的外部视觉、听觉等信息暂时存储在该缓冲区中。因此可以由ABox断言的形式进行描述。
6. 动作缓冲区中，保存了智能体作用于外部的信息。ActBox中定义的动作保存在该缓冲区中。当系统需要执行的目标确定之后，从ActBox中选取相应的动作进行组合输出，从而完成对环境的刺激响应。

### 3.3 动态描述逻辑基础

动态描述逻辑(Dynamic Description Logic, DDL)是描述逻辑的拓展，其在描述逻辑的基础上引入了动作维[27] [64]。从而既可以刻画动作的内涵，也可以将动作本身作

为一种知识进行动态推理。在动态描述逻辑中，原子动作表示为前提公式集合和效果公式集合。外界客观的世界由前提公式和结果集合公式共同刻画。动态描述逻辑可以表示简单的动作，也可以表示复杂的动作。复杂的动作需要在简单的动作之上通过动作构造符进行构造，动作构造包括顺序、测试、选择和迭代。无论是原子动作，还是复杂动作，都可以通过DDL进行构造。DDL已经被证实是可靠的、完备的和可终止的。接下来，我们主要介绍DDL的相关概念和推理问题。

### 3.3.1 基本概念

**定义 3.1 (基本符号)** 在DDL语言中，包含以下符号：

- 概念名：  $C_1, C_2, \dots$ ;
- 关系名：  $R_1, R_2, \dots$ ;
- 个体常元：  $a, b, c, \dots$ ;
- 个体变元：  $x, y, z, \dots$ ;
- 概念运算：  $\neg, \sqcap, \sqcup$  以及量词：  $\exists, \forall$ ;
- 公式运算：  $\neg, \wedge, \rightarrow$  以及量词  $\forall$ ;
- 动作名称：  $A_1, A_2, \dots$ ;
- 动作构造：  $;, \cup, *, ?$ ;
- 动作变元：  $\alpha, \beta, \dots$ ;
- 公式变元：  $\varphi, \psi, \pi, \dots$ ;
- 状态变元：  $u, v, w, \dots$ 。

**定义 3.2 (概念定义)** DDL中的概念定义如下：

- 原子概念  $P$ ，全概念  $\top$ 、空概念  $\perp$  都是概念；
- 如果  $C$  和  $D$  是概念，则  $\neg C, C \sqcap D, C \sqcup D$  都是概念；
- 如果  $R$  为关系，  $C$  为概念，则  $\forall R.C, \exists R.C$  都是概念；
- 如果  $C$  是概念，  $\alpha$  是的总做，则  $[\alpha]C$  也是概念。

**定义 3.3 (公式定义)**  $C$ 为任意概念,  $R$ 为任意关系,  $a, b$ 为个体常元,  $x, y, z$ 为个体变元,  $\alpha$ 是动作,  $DDL$ 的定义如下:

- 形如  $C(a)$ ,  $R(a,b)$ 和  $[\alpha]C(a)$ 的表达式是断言公式, 它们是不带变元的;
- 形如  $C(x)$ ,  $R(x,y)$ 和  $[\alpha]C(x)$ 的表达式称为一般公式, 它们是带变元的;
- 断言公式和一般公式都是公式;
- 如果  $\varphi$ 和  $\psi$ 都是公式, 则  $\neg\varphi, \varphi \wedge \psi, \varphi \rightarrow \psi, \forall x\varphi$  都是公式;
- 如果  $\varphi$ 是公式, 则  $[\varphi]$ 也是公式。

**定义 3.4 (实例代换)** 形如  $\{a_1/x_1, \dots, a_n/x_n\}$ 的有穷集合称为实例代换。其中,  $a_1, \dots, a_n$ 为个体常元,  $x_1, \dots, x_n$ 为个体变元, 且满足  $x_i \neq x_j, i, j \in \{1, \dots, n\}, i \neq j$ 。

**定义 3.5 (实例公式)** 设  $\varphi$ 为一个公式,  $x_1, \dots, x_n$ 为公式中出现的个体变元, 令  $\varphi'$ 为  $\varphi$ 通过实例代换  $\{a_1/x_1, \dots, a_n/x_n\}$ 得到的公式, 则称  $\varphi'$ 为  $\varphi$ 的公式常例。

**定义 3.6 (原子动作)** 一个动态描述逻辑的原子动作是一个形如  $A(x_1, \dots, x_n) = (P_A, E_A)$ 的表达式, 其中:

- $A$ 为动作的名称, 它用来表示原子动作的操作符;
- $x_1, \dots, x_n$ 为个体变元, 制定动作的操作对象, 因此也常称之为操作变元;
- $P_A$ 为前提公式集, 其用来制定动作执行前需要满足的条件;
- $E_A$ 为结果的公式集合, 用来制定动作执行后的结果; 该结果集由一组表达式组成, 在不同的环境中, 表达式具有不同的执行结果。

在此, 我们需要特别说明:

1. 动作定义了状态之间的转化关系, 即动作  $A$ 将使得状态发生转变。如, 在世界的原始状态是  $u$ , 在动作  $A$ 的作用下, 系统会转移到状态  $v$ 。但是这种状态的转变是有条件的, 即现在的客观事件满足动作  $A$ 发生的条件。具体来说, 客观世界  $u$ 满足动作  $A$ 发生的条件, 同时在动作  $A$ 发生后, 客观世界  $v$ 也满足动作发生的条件。
2. 由于动作  $A$ 的发生会对外部环境进行改变。因此, 动作发生的前提公式和结果公式会有些不同。对于公式  $A$ , 动作发生的前提条件  $u$ 要求在动作发生前的环境中得到满足, 而  $v$ , 要求在动作发生后的环境中得到满足。否则, 动作  $A$ 执行的条件不能够正确的执行。

**定义 3.7 (原子动作实例化)** 设  $(x_1, \dots, x_n) \equiv (P_A, E_A)$ , 为一个动作描述,  $A(a_1, \dots, a_n)$  是在  $A(x_1, \dots, x_n)$  上经过实例代换  $(x_1/a_1, \dots, x_n/a_n)$  得到的, 则称  $A(a_1, \dots, a_n)$  为  $A(x_1, \dots, x_n)$  的动作实例。称  $A(a_1, \dots, a_n)$  为原子动作,  $P(a_1, \dots, a_n)$  称为动作  $A(a_1, \dots, a_n)$  的前提,  $E_A(a_1, \dots, a_n)$  为动作  $A(a_1, \dots, a_n)$  的结果集。

**定义 3.8 (复杂动作)** 复杂动作由简单动作进行通过产生:

$$\pi, \pi' \rightarrow \alpha | \varphi? | \pi \cup \pi' | \pi; \pi' | \pi^*$$

其中  $\alpha$  代表公式,  $\varphi?$  代表 DDL 中的公式。

$\varphi?$  称为测试公式, 他表示对  $\varphi$  进行测试, 以确定其在当前状态下的成立情况。  $\varphi$  成立, 则返回为真, 否则, 返回假;  $\pi \cup \pi'$  称为选择动作, 该动作执行后, 将在  $\pi$  和  $\pi'$  中挑选一个动作进行执行;  $\pi; \pi'$  称为顺序动作, 该动作的执行, 动作  $\pi$  和动作  $\pi'$  顺序执行;  $\pi^*$  称为迭代动作, 该动作的执行将重复执行  $\pi$  动作, 直至程序顺利完成。例如 *if  $\varphi$  then  $\pi$  else  $\pi'$*  可以记作  $(\varphi?; \pi) \cup ((\neg\varphi)?; \pi')$ , *while  $\varphi$  do  $\pi$  endwhile* 可以记作  $(\varphi?; \pi); (\neg\varphi)?$

随着动作不断的被执行, 外部的世界状态不断的被我们所改变。而这个变化的过程, 实际上论域中个体的属性以及个体之间的关系进行动态变化过程, 每个状态下个体的属性, 以及个体之间的关系, 组成了客观世界的状态描述。

通过对简单动作构造的复杂动作, 我们可以方便的表示计算机程序的控制流程。在计算机中, 典型的控制流程有判断, 循环, 分支, 通过我们现在的复杂动作构造, DDL 可以方便的对动作进行表示。

DDL 主要包含了三个部件,  $ABox$ ,  $TBox$  和  $ActBox$ 。其中  $ABox$  是实例断言的集合,  $TBox$  是术语公式的集合,  $ActBox$  是动作定义的集合。  $ActBox$  中既包含了原子动作, 也包含了复杂动作。

### 3.3.2 动态描述逻辑语义

本节中, 我们主要对动态描述逻辑的语义进行解释。我们以最简单的  $\mathcal{ALC}$  动态描述逻辑为例子。动态描述逻辑的解释可以看做是一个形如  $(W, \Delta, I)$  的三元组。其中  $W$  表示一个可能世界的集合,  $\Delta$  是一个论域,  $I$  是其中的一个解释函数。

对于每个  $w_i$  属于  $W$ , 解释函数  $I$  将  $w_i$  通过  $I(w_i)$  将  $w_i$  的每一个概念映射到  $\Delta$  的一个子集, 将每个个体名映射到  $\Delta$  的一个元素, 将每个角色名称映射到  $\Delta \times \Delta$  的一个子集。在此基础上, 我们给了了 DDL 的语义:

1.  $I(w) \models C(p)$  iff  $p^{I(w)} \in C^{I(w)}$ ;



2.  $(\neg C)^{I(w)} = \Delta \setminus C^{I(w)}$ ;
3.  $(C \sqcap D)^{I(w)} = C^{I(w)} \sqcap D^{I(w)}$ ;
4.  $I(w) \models R(p, q)$  iff  $(p^{I(w)}, q^{I(w)}) \in R^{I(w)}$
5.  $(\forall R.C)^{I(w)} = \{x | \text{对每个 } y \in \Delta, \text{若 } (x, y) \in R^{I(w)}, \text{则 } y \in C^{I(w)}\}$
6.  $I(w) \models \neg \phi$  iff  $I(w) \not\models \phi$ ;
7.  $I(w) \models \pi \vee \psi$  iff  $I(w) \models \pi$  or  $I(w) \models \psi$ ;
8.  $(\phi?)^I = \{(w, w) | I(w) \models \phi\}$ ;
9.  $(\pi_1 \cup \pi_2)^I = \pi_1^I \cup \pi_2^I$ ;
10.  $(\pi_1; \pi_2)^I = \{(w, w') | \text{存在某个 } w'' \in W \text{ 使得 } (w, w'') \in \pi_1^I \text{ 且 } (w'', w') \in \pi_2^I\}$
11.  $(\pi^8)^I = \pi_1^I$  的传递闭包;
12.  $I(w) \models \langle \pi \rangle$  iff 存在一个  $w' \in W$  使得  $(w, w') \in \pi^I$  且  $I(w') \models \phi$

### 3.3.3 动态描述逻辑推理任务

**定义 3.9 (冲突)** 在动态描述逻辑推理的过程中, 会遇到冲突的情况。当遇到此种情况, 推理进行回溯或者结束。 $C$ 是任意概念,  $R$ 为任意的关系,  $a, b$ 为任意的个体常元。

- $\{\perp(a)\}$ ;
- $\{C(a), \neg C(a)\}$ ;
- $\{R(a, b), \neg R(a, b)\}$

**定义 3.10 (公式集合一致)** 一个断言公式 $\mathcal{F}$ 是一致的, 当且仅当公式 $\mathcal{F}$ 中不含有冲突, 否则, 我们称 $\mathcal{F}$  是不对称的。

下面我们给出断言公式一致性的检测方法, 该算法主要是利用推理规则对所断言的公式进行扩充, 查看是否有冲突出现。在推导的过程中, 用到的主要方法如下:

- $\sqcap$ 规则: 若  $C_1 \sqcap C_2 \in \mathcal{F}$ , 且  $C_1(x) \notin \mathcal{F}$ ,  $C_2(x) \notin \mathcal{F}$ , 则将  $\{C_1(x), C_2(x)\}$  并入到  $\mathcal{F}$  中;

- $\sqcup$ 规则：若  $C_1 \sqcup C_2 \in \mathcal{F}$ ，且  $C_1(x) \notin \mathcal{F}$ ， $C_2(x) \notin \mathcal{F}$ ，则将  $\{D(x)\}$  并入到  $\mathcal{F}$  中，其中  $D = C_1$  或者  $D = C_2$ ；
- $\exists$ 规则：若  $\exists R.C(x) \in \mathcal{F}$ ，且没有  $y$  使得  $R(x, y) \in \mathcal{F}$  且  $C(y) \in \mathcal{F}$  则将  $\{C(y), R(x, y)\}$  并入到  $\mathcal{F}$  中；
- $\forall$ 规则：若  $\forall R.C(x) \in \mathcal{F}$ ， $R(x, y) \in \mathcal{F}$ ，且  $C(y) \notin \mathcal{F}$ ，则将  $C(y)$  并入  $\mathcal{F}$  中；
- 动作 $\alpha$ 规则：若  $[\alpha]C \in \mathcal{F}$ ，其中  $\alpha = (P_\alpha, E_\alpha)$ ，则将  $P_\alpha$  中所有的条件从  $\mathcal{F}$  中删除，将执行  $E_\alpha$  后得到的结果添加到  $\mathcal{F}$  中，将概念  $C$  并入  $\mathcal{F}$  中。

最后，检测  $\mathcal{F}$  中是否含有冲突，若没有冲突，则  $\mathcal{F}$  是一致的，否则  $\mathcal{F}$  是不一致的，整个算法推理结束。

但是推理的过程之中， $\sqcup$ 操作是比较特殊的。因为我们在推理选择的过程中，我们只是选择了其中的一个分支，这样我们可能碰到冲突。因此，在实际的图例过程中，如果一个分支出现了冲突，就放弃该分支，然后重新选择进行推理。直到所有的选择都结束的情况下才正式确定公式是不一致的，存在冲突。

上述的算法是对经典 *Tableaux* 算法的扩展，该算法主要是在传统 *Tableaux* 算法的基础之上只能加了对动作的描述和推理，确保了算法的正确性。

断言公式的一致性问题和在动态描述逻辑推理中居于举足轻重的地位。因为，其它所有的推理问题都可以转化为一致性推理问题或者是应用一致性推理结果。

**定义 3.11 (动作的一致性)** 一个形如  $A(x_1, \dots, x_n) \equiv (P_A, E_A)$  动作是一致的，当且仅当对于任意的个体常元  $a_1, \dots, a_n$ ，动作实例  $A(a_1, \dots, a_n)$  的前提集合  $P_A(a_1, \dots, a_n)$  和结果集  $E_A(a_1, \dots, a_n)$  是一致的。

在描述逻辑中，对动作的描述是一般性的描述。它是对具有该动作的所有实例的一种抽象。它用来刻画动作执行的基本模式，且动作的执行需要满足其执行条件和相应的执行结果。在动态描述逻辑中，动作的描述和概念的描述类似，是一种领域的公理。因此，在应用之前，需要对其合理性和一致性进行检测。在验证一致性的时候，我们需要构造出一个动作实例，然后检测其动作的前提和动作的结果的一致性。代入任意个体实例  $a_1, \dots, a_n$  之后，因为对于动作描述中的条件  $p=q$  和  $p \neq q$ ，可以看做是一般性的动态描述逻辑关系，而其它关系都是断言公式。这样，前提公式集合  $P_A(a_1, \dots, a_n)$  和结果公式集合  $E_A(a_1, \dots, a_n)$  都转化为断言公式集合，从而动作描述的一致性就转化为断言公式的一致性问题。

我们已知断言公式的一致性问题是可判定的，因而动作描述的一致性也是可判定的。动态描述逻辑本质上还是描述逻辑，只是增加了其对动态知识，即动作的表示。因此，与传统的动作推理还存在区别。

**定义 3.12 (动作的可执行性)** 原子动作  $(P, E)$  相对于  $TBox\ T$  和  $ActBox\ Act$  来说正  $ABox\ A$  上是可执行的，当且仅当  $Conj(A) \rightarrow Conj(P)$  相对于  $T$  和  $Act$  是可执行的。

$Con(.)$  表示公式的合取。该定理为我们提供了一个动作在给定  $ABox$  下的可执行方法。我们通过判定  $\neg(Conj(A) \rightarrow Conj(P))$  是不可满足的，推断出  $Conj(A) \rightarrow Conj(P)$  是有效的。从而，我们可以判定动作  $(P, E)$  在  $ABox\ A$  下是有效的。

**定义 3.13 (动作投影)** 动作  $\pi$  执行后相对于  $TBox\ T$  和  $ActBox\ Act$  来说使得公式  $\phi$  成立，当且仅当公式  $Conj(A) \rightarrow [\pi]\phi$  是有效的。

根据动作的投影定理，我们可以构建动作的规划推理。

**定义 3.14 (动作规划)** 给定  $ABox\ A_1, A_2$ ，以及动作序列  $Seq_a = (P_1, E_1), (P_2, E_2), \dots, (P_n, E_n)$ ，我们称  $Seq_a$  是  $A_1$  到  $A_2$  动作规划，当且仅当动作序列  $(P_1 \dot{\bar{g}} E_1); (P_2 \dot{\bar{g}} E_2), \dots, (P_n, E_n)$  在  $A_1$  上执行后，公式  $Conj(A_2)$  成立。

根据动作规划的定义，我们可以在动作的可执行性、动作投影基础之上，对动作序列进行构建。

### 3.4 智能体ABGP认知模型

由于单个智能体的能力有限，面对复杂的任务，需要多个智能体协作完成。各个智能体的协同合作，大大提高了整个系统的性能。其中，每个智能体需要不断的同其它智能体进行通信，且每个智能体还需要同外部的环境进行交互。

为此，我们需要构建认知模型。本节，我们主要讲述短期记忆部分，我们称之为ABGP(Aware-Belief-Goal-Plan)模型。智能体ABGP是心智模型CAM的一部分。如图3.2所示，ABGP短期记忆模块主要由四部分构成：环境感知部分，信念、愿望和规划。

环境感知是智能体在与环境交互的过程中所生成的知识。Endsley指出，环境感知具有四个主要的特点[33]：

- 环境感知是一种特殊的知识，这些知识智能体从外部的环境中获得；

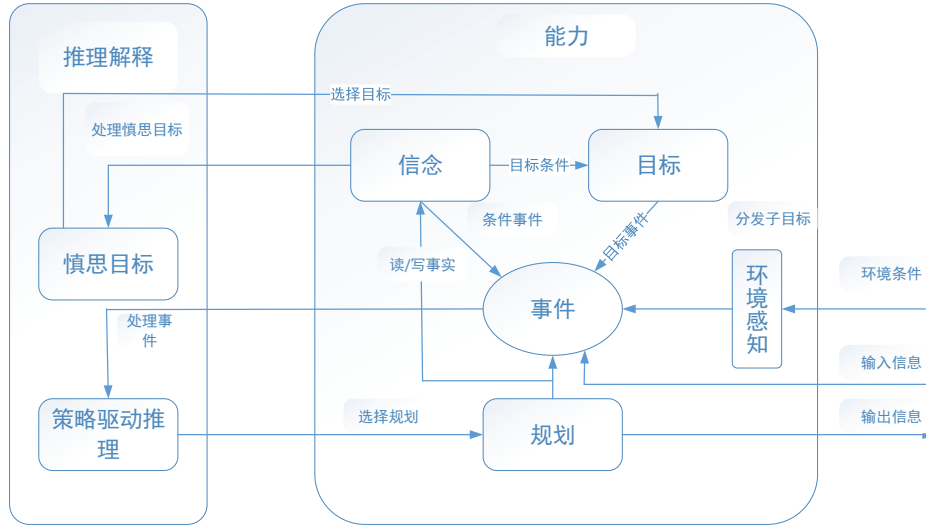


图 3.2: 智能体ABGP认知模型

- 外部的环境在动态的改变，所以智能体的环境感知也需要动态的变化；
- 智能体获得外部环境的信息，需要不断的同外部环境进行信息的交互；
- 环境感知通常是第二位的目标，也就是说，我们在达到我们目标的时候，需要考虑环境的因素。

Gutwin等提出了环境感知工作空间的概念框架，用来对一组智能体同外部环境交互进行建模[49]。作者在该文章中列举出了智能体环境感知概念框架中需要感知的元素。工作空间环境感知的信息，通过对这些元素的组合构成。

对于智能体内部的状态，我们可以同构建BDI模型来进行表示，该理论是一种可行的推理理论。BDI理论成功的将复杂的动机实例进行了模型化的表示，且该模型较简单。这就意味着，动作的产生和人们的意愿具有很大的联系，在建立模型的过程中忽略了认知中的其它的因素，如情感。BDI模型另外的一个有点是，它采用了心理学中流行的概念，且这些概念和人们的日常行为非常的贴切。

有感于此，我们在这里提出了我们的认知模型ABGP。以往的认知模型在多智能体协作的时候，主要考虑的是智能体之间的协同和信息交换，而对每个智能体外部环境的考虑欠佳，为此，我们将环境的感知添加到我们的认知模型中。

### 3.4.1 环境感知

多智能体感知应该考虑到基本因素和智能体之间的关系。多智能体的环境感知可以定义为一个二元组 $MA=\{\text{元素}, \text{关系}\}$ ，其中环境感知的元素主要包括：

- 身份(角色):谁是参加者;
- 地点: 他们的位置参数;
- 意图: 多智能体打算做什么, 它们的目标;
- 动作: 他们在做什么, 买对需要实现的目标, 智能体需要对外部产生自己的动作;
- 能力: 智能体可以做什么呢? 有时, 不同的智能体具有不同的功能, 因此, 需要对外公布出自己自身所具有的能力;
- 目标: 智能体需要达到什么样的目标;
- 时间点: 当智能体需要对外部的环境产生动作的时候, 需要有一个时间点, 在该时间点确定后, 智能体才能对外部的环境进行动作。

智能体之间的基本关系主要有:

- 任务关系: 任务关系涉及到任务的划分。有的时候, 单独的一个智能体受限于自身能力, 自己并不能够完全将任务完成。为此, 我们需要将一个大的任务划分为一个较小的任务, 这样通过分布式协作, 最终将大的任务完成。
- 角色关系: 当任务划分都具体的智能体的时候, 还要考虑到不同智能体自身的属性。由于智能体属性的不同, 这样在分工的时候, 我们需对其进行考虑, 尽量使得每个智能体做自己擅长的工作。角色关系定义了多智能体在合作的过程中的角色关系。
- 操作关系: 操作关系定义了多智能体在合作过程中操作的集合。
- 活动关系: 活动关系定义了多智能体在合作过中, 在相应时间智能体的角色。
- 合作关系: 合作关系定义了智能体相互之间的交互。

#### 3.4.2 信念

信念代表了一种信息, 一部分信息智能体可以从其所处的环境中获得, 一部分信息从自己自身的内部产生。这些信息构成了智能体自身的知识库。我盟通常把信念看做是智能体的一个知识库, 这个知识库中包含了丰富的内容, 主要有基本的公理、客观事实、客观的数据等等。

信念知识可以看做是一个三元组 $K = \langle T, S, B \rangle$ 。T描述了领域内基本的概念和它们的定义，以及从领域内的概念所产生的公理，即领域本体；S是领域中的事实和公理的对应关系，在这些限制中，存在着一定的因果关系。这些限制保证了知识库的一致性和完整性；B用来表示当前的信念状态集合，包含了事实和数据。随着时间的推荐，智能体外部的环境和智能体自身的状态的改变，B中的内容也在动态的改变。

### 3.4.3 目标

目标表示智能体的愿望，同时目标也驱动智能体产生动作作用于外部的环境。随着时间的推荐，智能体会有很多的愿望。但是这些愿望之间有的时候是一致的，有的时候是自相矛盾的。为此，在智能体有一个慎思处理，将所有的愿望统一为一致的愿望。对于，智能体内部中有冲突的愿望，按照系统中所拥有的算法，将其从愿望集合中剔除。在一个基于目标的设计中，目标具有不同的类型。有的目标是维持型的，即该目标主要是为了维护系统的某种特定的状态。有的目标是实现型的，该目标系统以前并没有达到，需要智能体根据现有的状态，向外部的环境输出自己的动作，从而达到自己所期望的目的。

目标的状态，允许我们在对智能体进行建模的时候，有多重处理方式。智能体不仅仅是被动的对外部的环境进行反映，即外部环境发生变化的时候，智能体为了维持自身的平衡，所采取一系列的动作。同时，智能体还可以不受外部环境的影响，主动的去完成一些目标。智能体中的目标需要生成，主要有3种目标生成方式：

1. 系统设计者在设计系统的时候确定或者是在系统初始化的时候选定；
2. 随着环境的动态变，为了响应环境的变化，而产生的目标；
3. 智能体内部产生目标。

### 3.5 策略驱动的动作规划

当一个目标被选择后，智能体必须选择一个有效的途径到达该目标，甚至有的时候需要修改现在正在运行的目标，这个推理的过程我们称为规划。为了完成这个推理过程，智能体可以采用两种方式：其中的一种方式是采用智能体内部已有的规划库，这个库中包含有许多已经定制好的规则，这样的库我们称为静态库；另外一种方式是进行及时的规划，不采用静态数据库中的知识，我们称之为动态规划。

静态规划，用来处理某些特定的目标。预先设置的目标可以通过查询静态库，从而从库中该得到相应的动作支撑，达到目标。静态库中的规则我们已经给定，因此，对于我们需要达到的目标，此时的问题转化为搜索问题。

对于特定的目标，动态规划需要找到一条有效的途径来实现，这是需要考虑当前的信念，但是并不应用系统内部的静态知识库。在动态规划的过程中，自然环境一直正进行着动态的变化，因而，信念状态也伴随着动态的变化。即使对于相同的目标，由于所处环境状态的不同，规划和实现的方式也会有差别。因此，动态规划对于智能体来说是十分必要的，尤其对于多智能体协作的复杂环境下。在动态规划方式下，我们采用有顶到下的方式。在这种方式下，一个目标包含一系列的子目标。因此，我们实现一个目标，首先要实现一系列的子目标。只有当所有的子目标都实现的情况下，目标才算最终的实现。

关于策略有许多种不同的定义方式，不同的领域有不同的标准。例如，IETF/DMTF定义策略为一系列的管理规则。有许多人把策略简单的等于一些规则，这种看法是狭隘的。在这里，我们采用相对宽松的定义：策略是用来指导系统行为的方法。在多智能体环境系下，策略用来指导多智能体的合作过程，在这期间，每个智能体都遵循相同的策略。策略告诉智能体应该做些什么(目标)，怎样做(动作)，如何量化参数(效用参数)从而指导智能体的行为。

Kephart等定义了一个统一的框架，用来进行自动的计算，计算的时候考虑的主要是系统所处状态和当前所采取的动作[55]。通常情况下，我们需要对智能体所处的状态进行标注，对于智能体中某些重要的部分，我们同样需要进行标注。且进行标注的时候，还有时间点的考虑。时间点标注的时候，具有非常重要的作用。通常情况下，我们在表示状态S的时候，状态S会有一个属性向量。在该向量中的参数，或者是通过传感器直接获得，或者是通过智能体内部的推理部件进行推理而得出。一个策略会直接或间接的导致系统中动作的发生。当动作发生的时候，会导致智能体或者智能体内部的某些部件发生预订的变化。这一种，对于维持的动作，会保持现有的状态不变；而如果是实现型的目标，则会发生相应的变化，且该变化达最终目标的一个中间的状态。这个统一的框架，同样的适用于多智能体系统。

一个多智能体系统，在时刻 $t$ ，系统所处的状态为 $s_0$ ，在 $s_0$ 状态，系统中有一系列的属性技巧数值来对其进行表示。动作策略会直接或者间接的导致动作的发生。在这样的情况下，系统会从一个状态转移到另外的一个状态。我们可以把策略看做是一个转移函数，多智能体可以通过策略进行状态的转移。如图3.3所示 根据采取的策略不同，智能体会采取不同的状态，从而，系统也会达到不同的状态。

### 3.5.1 动作策略

动作策略用来描述在给定状态下多智能体所采取的动作策略。一般的动作策略表示为 $P=\{S_t, A, \_, \_ \}$ ，其中“ $\_$ ”表示空。动作采用产生式规则的形式，

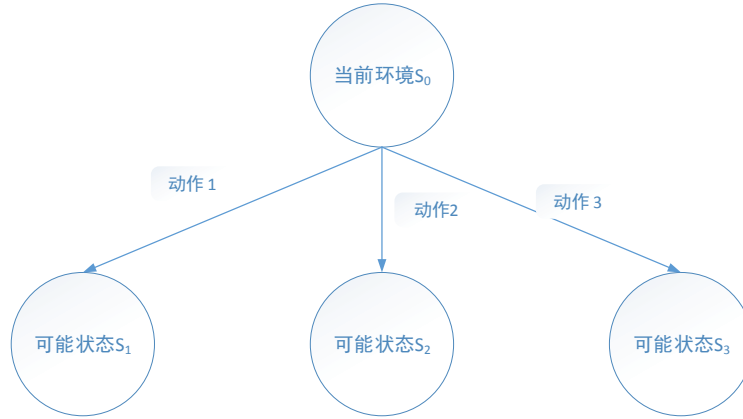


图 3.3: 多智能体策略驱动的状态转移

即IF(Condition) THEN(Action)。Condition代表动作产生钱智能体需要满足的前提条件，它或者是一个状态或者是一系列的状态满足当前动作执行的条件。Action代表需要执行的动作。

### 3.5.2 目标策略

目标策略不是给定一个状态，要求系统需要采取某些动作达到该状态，而是用来描述系统需要达到的状态。目标策略在给出的时候，有的时候给出的是系统需要达到的一个状态，有的时候给出的是系统需要达到的几个状态，系统在执行的时候可以根据自身的情况选择一个状态进行实现。在动作策略中，系统会采取一个非常明确的策略而达到系统需要抵达的状态。在目标策略中，系统给定了目标状态，那么在实现的过程中，需要一个动作的序列来到达给定目标。这个动作序列有可能很长，有可能很短。智能体每产生一个动作，系统的状态都会改变。因而，系统也是有一系列的状态转换而形成的。是一个离散的事件序列。通常目标策略表示为 $P=\{S_t, \_, S_g, \_ \}$ ，其中“\_”表示空。

### 3.5.3 效用函数策略

效用函数策略是一个目标函数，该函数指出了系统达到可能状态的概率。效用函数相对于动作策略和目标侧率，提供了一个更加宽松和详细的机制。但是效用函数的制定者，需要对系统本身具有非常详实的了解，充分的了解模型本身以及认知模型所采用的算法、优化等。通常目标策略表示为 $P=\{S_t, \_, \_, U\}$ ，其中“\_”表示空。

### 3.5.4 动作规划简介

动作规划是智能体为了完成自身的认知任务而构建的动作序列。在ABGP认知模



型中，负责动作规划的主要有两个部件，即目标构建器、目标分解器和目标动作规划器。目标构建器负责生成系统中需要解决的目标。目标分解器负责从记忆中提取需要进行规划的目标的相关信息，将目标分解为可实现的小目标。由于每个智能体的能力有限，对于某些大目标超出了自己自身的能力，需要将其分解，从而是得智得以完成目标。目标动作规划器的主要任务是对每个子目标进行规划，生成达到目标的动作序列。将生成的动作序列存储到动作缓冲器中，智能体按照动作缓冲器中动作的序列依次进行执行。下面我们组要对目标如何进行分解，如何进行规划进行讨论。

我们观察，当我们在做自己的工作或者是按照自己的兴趣做某件事情的时候，都存在着动机。但是这些动机之间也有区别，有的是利于自己的，有的是利于集体的，但是无论如何，整个过程中都有动机的存在。比如我们吃饭，在吃饭之前是因为我们感觉到饥饿，体内的能力下降，需要补充我们体内的能量。这就是我们的一个需求，一个动机从而产生。

智能体将动作作用于其所处的外部环境，改变自身的状态。我们任然需要对智能体对外部环境动作后的状态进行进行评价，对于执行的效果，我们主要主要是根据动作执行后的受益来进行计算[92]。Sun等人提出了他们的看法，认为动机可以为动作的受益评估提供一种力量环境[104]。在这种方式下，动机本身对评估具有很大影响。即使在相同的环境下，我们采去不同的动作，我们的收益也会不同，这很符合我们的实际情况。最典型的例子，当我们饥饿的时候，我们既可以吃米饭，也可以吃牛肉，这两种方式都暂时的解决了我们的饥饿问题。但是他们的时效性是不同的，从我们现实生活中的经验来看，吃牛肉坚持的坚持的时间要大于吃米饭的时间。从而，可以看出，动作本做为我们的一个评价指标还是有其合理性的。

### 例 3.1 (认知情景) 假设存在两个关于“王杰”的不容认知场景

- 情景1: 为了在奥林匹克数学竞赛中取得优异的成绩，王杰一直在认真的钻研数学题目。现在，这时候王杰的主要动机就是在比赛中尽可能的取得好成绩，只有复习的越充分，取得好成绩的机会才会越大。所以，王杰一直很努力。
- 情景2: 为了在奥林匹克数学竞赛中取得优异的成绩，王杰一直在认真的钻研数学题目。但是这时候王杰感觉到了饥饿，因此他需要暂时打断学习的过程，去吃些东西，以缓解暂时的饥饿。

情景1和情景2是王杰处在这两个场景。在情景1中，王杰的主要任务是抓紧时间复习，以在奥林匹克比赛中取得优异的成绩。在此种环境下，王杰主要采取的动作如“复习以前错误的题目”、“研究新的题目”，“思考题目新的解法”等动作行为具有高的

收益。在情景2中, 王杰感觉到了饥饿。在这种情况下, 饥饿是复习课程具有更高的优先级别, 因此, 王杰选择了补充体能, 吃些零食。在这个动作规划的过程中, 我们可以看到, 王杰首先考虑了“饥饿”这一动机, 在“饥饿”的语境下, 王杰选择的动作是“吃零食”, 从而缓解饥饿的程度。整个选择的过程中, “吃零食”的受益要大于“复习以前错误的题目”、“研究新的题目”, “思考题目新的解法”等动作的收益。因此, 王杰选择了“吃零食”这一最大收益的动作。

通过这里例子我们可以明显的看出, 动机是有优先级别的。智能体会根据自身的情况, 确定不同动机的优先级别。在不同的时间, 不同的地点, 不同的环境下, 动机具有不同的强度。如前文章所述, ABGP认知模中引入了NGI动机模型。

### 3.5.5 基于动态描述逻辑的规划

目标规划是目标队列中按照智能体内部的优先级算法, 选则优先级算法最高目标进行规划。在规划完毕之后, 系统将所规划的队列存储动作的缓冲区中顺序执行。面对现有的目标, 智能体必须通过自己内部的推理系统规划处所有的动作序列。在规划的过程中, 应该考虑多种信息, 以产生符合当前情况的动作序列。

系统在规划的过程中, 有两种不同的规划方案。一种是静态的, 一种是动态的。静态的方案中, 智能体在最开始的时候设计者会将规划库中存储一定的方案。这些方式是历史累计的知识, 我们将其编写为案例库。当智能体在执行的过程中, 如果遇到类似的情况, 则可不必要进行复杂的推理, 直接从案例库中查找已有的解决方案, 然后将可行的的动序列输出。所以在ABGP模型中, 我们引入了静态库。在这里, 我们的知识表现形式为 $(S_i, S_g)$ , 其中 $S_i$ 表示目标的起始阶段, 而 $S_g$ 表示目标的最终状态。该种方案最大的优点是速度快, 但是由于其是历史积累的信息, 所以, 其最有一定的局限性。因为智能体处在动态变化的自然环境中, 其积累的知识毕竟是很少的一部分。当遇到新的问题后, 如果知识库中不存在历史的记录, 则智能体需要动态进行推理。

当智能体执行完毕一个目标或者有更紧急的目标需要执行的时候, 智能体会选择新的目标进行执行。通常情况下, 目标非常的大, 为此, 我们需要采用分而治之的思想, 将大目标分解为一个个的小目标。这样, 当每个小目标完成后, 大的目标也就完成了。

当智能体的目标确定之后, 需要对智能体的中的目标进行分解, 将其转变为一个一个小目标。

对于大目标的分解, 我们主要有两种方式。一种是采用树形结构进行存储, 所有的叶子节点就是我们所需要实现的最小目标。一种是采用序列存储, 通过一系列的动

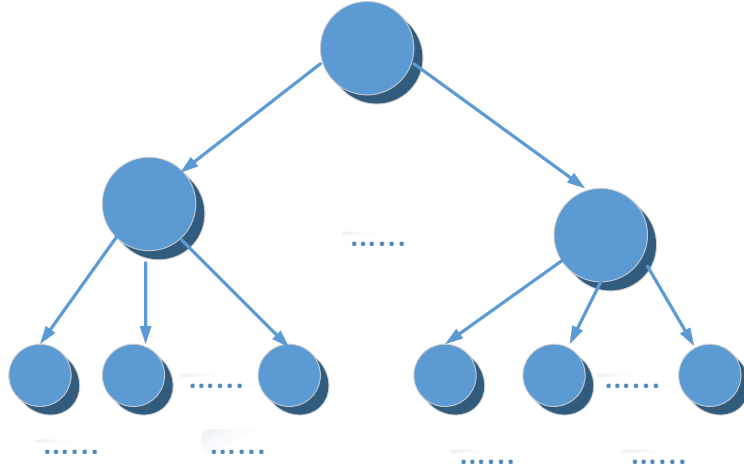


图 3.4: 树形目标存储结构

作进行进行状态的转变，最终到达我们的理想目标状态。对于采用树形结构存储的形式，该方式采用层次化存储。如图3.4所示 在树形的结构中，顶层存储的是我们系统需要实现的最终目标。最终目标分解为一个或者多个子目标，如此进行迭代。最终分解到不能分解为止。系统需要执行的时候，只要将所有的子目标一次实现即可。采用树形结构有尤其自身的有点，实现简单。每个目标单独实现即可。但是，树形结构尤其局限性。现实生活中的好多目标，虽然也需要分解为一个个的小目标，但是小目标之间具有时序顺序。因此，树形存储结构在此环境中就会失效。如例3.2所示：

**例 3.2 (树形存储失效)** 我们学习的目的是为了更好的为祖国服务。在这里，工作是我们的最终目标。但是在这个目标实现之前，我们需要成功的实现许多的子目标。最典型的有：幼儿园，小学，初级中学，高级中学，大学。当我们在每个阶段都完成了自己的学习任务后，我们就可以毕业为国家服务了。现在来看，这里边有严格的时序问题。即幼儿园一定是最开始所需要就经历的过程，大学毕业是我们最终的过程。我们需要实现这个过程中所有的学习任务，但是不能颠倒。我们不可能先进行大学课程的学习，然后在进行幼儿园课程的学习。

鉴于树形层次目标表示的局限性，我们采用动作序列表示。每个动作发生都将对我们现有的状态进行修改。为此，我们将历史的信息存储到情景记忆存储器中，对于我们需要实现的目标，我们首先查询情景情景记忆存储器。

在ABGP中，的每个目标表示为 $G=(G_i, ..., G_e)$ 。其中 $G_i$ 表示目标的起始状态， $G_e$ 表示目标的最终状态。

**定义 3.15 (目标实现序列)** 目标的实现序列 $G_{seq}$ 可以表示 $G_{seq}=\{(G_{init}, a), (G_{temp1}, a), ...,$

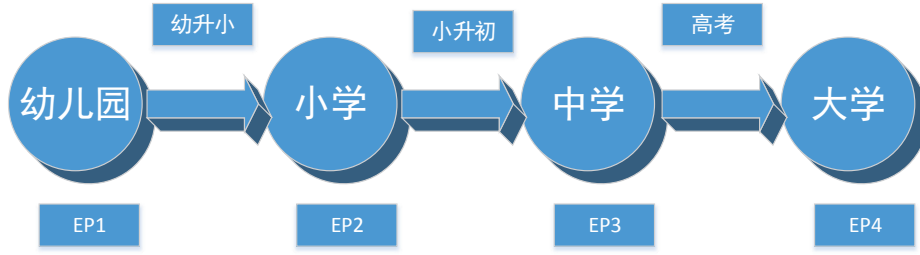


图 3.5: 情景记忆片段

$(G_{tempn}, a), G_e\}$ , 其中  $G_{temp}$  表示为中间的目标。我们的任务是通过检索在我们的情景记忆中将需要的动作检索出来。

**定义 3.16 (情景记忆存储)** 一个情景记忆存储表示为  $EP_{seq} = ((EP_1, a_1), (EP_2, a_2), \dots, (EP_n, a_n))$ 。其中,  $EP_i$  表示记忆的情景记忆,  $a_i$  每个情景动作的转换。每个情景记忆是一个描述逻辑公式  $Set(EP)$  的集合, 每个动作也是一个描述逻辑公式的集合。动作  $a(EP_i) = P(EP_i), E(EP_i), (1 \leq i \leq n)$ ,

对于存储的情景基于, 需要满足如情记忆的转换条件, 即:

- $Set(EP_i) \models Tran(P(EP_i));$
- $Set(EP_{i+1}) = (Set(EP_i) \setminus \{Tran(\nu) \mid \neg \nu \in E(SET_i)\}) \cup Tran(E(EP_i))$ 。

在定义3.16中,  $Tran$ 表示将动作应用与现在的情景, 而是得系统进入新的情景。 $Tran(\nu)$ 表动作替换后当前系统中的公式。在定义中, 我们主要说明了动作可以作用于当前场景需要的两个条件。条件一说明, 动作的发生需要满足的发生的条件, 即动作发生的前提需要得到保证。条二说明, 每当动动作作用于当前的场景, 系统的状态都会进行相应的改变。这两个条件同时满足才保证了我们系统中的状从最最开始的状态转换到我们的目标状态。

从定义3.16中我们可以清晰的看出, 动作的改变是是具有时序关系的。动作按照情景记忆中的顺序不断的执行, 情景状态不断改变, 知道最好达我们的目标状态。

如图3.5所示: 为一个人人生学习的轨迹。、在这个过程中, 我们需要经过幼儿园阶段的学习, 在学习的过程中满足考试的要求, 且经过幼升小的考试, 则进入到小学阶段的学习。当进入小学的阶段后, 经过长时间的学习, 达到小升初的要求, 则进行考试, 如果考试的达到要求, 则进入中学阶段进行学习。经过多年中学的学习, 达到高考的报名要求, 参加高考。然后通过高考的选拔, 进入大学是生活。在这里面, 幼儿园,

小学，中学，大学都是一个个的状态。其中幼儿园我们可以看做是起始的状态，而大学可以看做是我们最终的状态。

在我们的模型中，情景存储着历史的目标信息。该知识是我们进行推理的基础。通过对知识的选择匹配，我们以最快的速度得出我们所需要的信息。

为了加快对情景记忆中知识的检索，知识的组织我们采用倒排索引结构。即具有相同初始目标的记忆存储在一个链表中。

**定义 3.17 (动作蕴含)** 在这里我们给定动作  $\alpha = (P_\alpha, E_\alpha), \beta = (P_\beta, E_\beta)$ 。我们说动作  $\alpha$  蕴含动作  $\beta$ ，则当且仅当  $P_\alpha \models P_\beta, E_\alpha \models E_\beta$ 。

算法3中，我们首先按照目标的初始值对情景记忆中的知识进行检索。为此，我们采用了两个链表集合。 $Q_1$ 用来存储初始目标和最终目标都相同情景记忆知识。 $Q_2$ 用来存储初始目标相同但是最终目标不相同的记忆知识。之所以采用两个链表集合进行存储，主要是为了加快搜索的速度。 $Q_1$ 相当于我们的一个快速缓存，因为开始和结果都相同，而 $Q_2$ 则相当于一个普通存储，其开始目标相同，但是最终的目标不同。因为存在这样的情况，我们的情景记忆中的动作序列长于查询目标，即查询目标是情景记忆中的一个连续子集的情况。

对情景记忆中的候选集合进行倒排所以排列，加快了查找速度。同时按照情景记忆中数据目标状态的同，将搜索空间分为两部分，一部分我们可以进快速的检测，相当于缓存解集，令一部分则相当于慢速查找。在算法中，我们首先对当前的目标状态进行对比，如果目标状态满足情况，则我们考虑执行了的动作后的状态。将情景记忆中的状态和目标记忆中的状态进行对比。其次，还有比较比较动作是否可行。

该算法能否有效的找到我们需要的动作序列，主要受到目标规划的影响。由于我们在比较的过程中是一比一的按照次序顺序比较，因而，生成目标的序列状对系统具有至关重要的影响。这就需要我们对情景记忆中知识进行有效的总结和归纳，采用大量是数据分析典型的结果进行存储。只有这样才能增强动作规划的成功率。

算法3，主要考虑的是一对比对形式。但是我们客观世界中目标状态对应一个状态的情况。在这种形式下，情景记忆中的一连串目标可以看做是我们所需要实现目标的多个子目标。

对于智能体来说，现实生活中还存在更加复杂的情况。在我们所规划的动作中，一个简单的动作其实可以有几个动作来完成，但是我们前面所采用的短发对于这种情况是失效的。存在一种情况，我们静态库中的多个动作对应这我们规划动作中的一个动作，对于这种情况，我们的算法不能进行正确的处理，从而导致算法失败。所以，

---

**Algorithm 3** Episode Memory Match.

---

**Input:**

Desired Goal:Goal;

**Output:**

Finding Episode;

```

1:  $Q_1$ =The Episodes which the initial and end Goal as the same as  $G_i$ ;
2:  $Q_2$ =The Episodes which the the initial goal equal and the end goal does not equal;
3: while ( $Q \in Q_1$ ) do
4:   if ( $|G|==0$ ) then
5:     return Q;
6:   end if
7:   if ( $|Q| \neq |G|$ ) then
8:     continue;
9:   end if
10:  if (Match(Q,G)) then
11:    return Q;
12:  end if
13: end while
14: while ( $Q \in Q_2$ ) do
15:  if ( $|Q| < |G|$ ) then
16:    continue;
17:  end if
18:  while ( $|Q| \geq |G|$ ) do
19:    if (Match(Q,G)) then
20:      return Q;
21:    else
22:      DequeueHead( $Q_2$ );
23:    end if
24:  end while
25: end while
26: exit;

```

---

我们可以采用试探性的方式，对算法进行改进。为此，我们可以需要设置一个阈值，该阈值用来表示在几个动作之内对算法进行试探。如果在静态库中，在阈值规定的范

---

**Algorithm 4** Match.

---

**Input:**

Episode Memory element:  $Q$ ;

Desired Goal:  $G$

**Output:**

Boolean;

```

1:  $Q_{com} = \text{DequeueHead}(Q)$ ;
2:  $G_{com} = \text{DequeueHead}(G)$ ;
3: if ( $\text{MatchEpisode}(Q_{com}.EP, G_{com}.Goal) == \text{TRUE}$ ) then
4:   if ( $\text{getHead}(G_{com}) == \text{null}$ ) then
5:     return TRUE;
6:   else
7:     while ( $\text{next}(G_{com}) \neq \text{null}$ ) do
8:        $\text{actQ} = Q_{com}.a$ ;
9:        $\text{actG} = \text{action}(G_{com}.Goal)$ ;
10:      if ( $\text{MactchAction}(\text{actQ}, \text{actG})$ ) then
11:        if ( $\text{Match}(Q, G)$ ) then
12:          return TRUE;
13:        end if
14:      else
15:        return FALSE;
16:      end if
17:    end while
18:  end if
19: else
20:   return FALSE;
21: end if
22: return FALSE;
```

---

围之内试探有结果，则继续下一步探测，否则返回匹配失败。

在检测的过程中，还存在一种情况，即需要匹配的现存目标中，多个动作对应于静态库中的一个动作。对于这种情况，在我们的算法中也没有记性考虑。对于，这种情况，我们打算在下一步研究中，给予讨论。

---

**Algorithm 5** MatchAction.

---

**Input:**

Action  $\alpha$ ;

Action  $\beta$ ;

**Output:**

Boolean;

```

1: if ( $\alpha \& \beta == \text{null}$ ) then
2:   return FALSE;
3: else
4:    $res_1 = \text{Conj}(P(\alpha)) \rightarrow \text{Conj}(P(\beta))$ ;
5:    $res_2 = \text{Conj}(P(\alpha)) \rightarrow \text{Conj}(P(\beta))$ ;
6:   if ( $\neg res_1 \& \neg res_2$  unsatisfiable) then
7:     return TRUE;
8:   else
9:     return FALSE;
10:  end if
11: end if
12: return  $\frac{eff}{total}$ ;

```

---



---

**Algorithm 6** MatchEpisode.

---

**Input:**

Episode Memory element's temple state:  $EP_i$ ;

Goal's temple state:  $G_i$ ;

**Output:**

Boolean;

```

1: if ( $\neg(\text{Conj}(EP_i) \rightarrow \text{Conj}(G_i))$ ) then
2:   return TRUE;
3: else
4:   return FALSE;
5: end if
6: return FALSE;

```

---

当我们对动作的规划结束后，如果情景记忆知识库中的知识不能满足我们所需要的目标。我们可以进行试探性的解析。我们的目标中一个个的子目标，我们智能



体中的知识库可以按照此种情况进行组织。知识库中具有相似知识的动作我们组织在一起。然后还可以按照其实现的代价分别组成集合。可以按照目标的偏好，进行动作的选择和解析。但是，此种情况仍然存在不能将目标所需要的动作进行正确解析的情况。此种情况的优势也非常明显，智能体可以按照自己的倾向进行动作的选择。

### 3.6 联合意图

在多智能体系统中，由于单个的能力有限，求解的问题复杂，超出了单个智能体自身的能力。因此，对于复杂的问题，需要多个智能体进行联合完成对复杂问题的求解。多智能体之间的联合工作显示了多智能体的社会性行为。为了更加方便的揭示多智能体之间合作的这种关系，需要提供有效的机制对其进行分析。对于智能体来说，我们需要对智能体之间的联合性社会行为进行分析。例如，智能体之间的协作机制、单个智能体动作的选择机制和智能体内部状态变化的机制等等。

在人工智能领域，智能体由各种认知部件组成。其中最具有代表性的就是BDI体系结构。单个的智能体有其自身的特点，但是多智能体之间也有其自身的特点，且该特点不是单智能体特点的简单累加。多智能体之间有其独有的特性行为，为此需要提供抽象的概念模型。联合意图抽象概念模型可以有效的对多智能体之间的行为进行支持。

多智能体之间进行协作，主要分为两种情况。一种是所有的智能体共同实现一个任务，一种是所有的智能体共同维护一个智能体现有的状态。我们称共同实现一个任务的多智能体为实现型多智能体，共同维护一个状态的多智能体为维护型多智能体。对于实现型多智能体如，大型的计算任务，每个智能体承担一部分的计算量，最终完成整个计算任务。对于持续型多智能体如，多个智能体共同维持一个现有的状态，如多个智能体共同移动物体，且在移动的过程中物体保持水平状态不变。显然，此种约束对多智能体产生影响，他们广泛分布于多智能体合作中。为此，我们将按照其特点，分别进行讨论。多智能体之间的实现性联合意图指多个智能体联合实现某任务。多智能体之间的维持性联合意图指多个智能体联合维持某个任务的状态，使其横成立。

在多智能体中，除了单个智能体的信念外，外部环境还对其产生了显著的影响，包括实现型多智能体和维护型多智能体。当智能体在执行任务的过程之中，自己除了对任务本省了解，还需要时刻关注外部环境对其影响。

联合意图提下了多个智能体的联合行为选择，因为选择性和联合型是联合意图的本质属性[4]。联合意图具有如下重要性质：

- 选择性

联合意图是智能体对外来行为的合理选择，它对智能体的具有重要的影响，对其具有约束作用。意图是所有多智能体联合协作的根本性因素，对多智能体具有重要影响。

- **联合性**

联合意图是多智能体之间联合协作的体现。多智能体之间并不是简简单单的进行组团。每个智能体都知道其需要实现的目标，并且清楚的了解到现在的目标对于自己的能力来说过于复杂，需要同其它的智能体进行协作。

- **可满足性**

拥有联合意图的智能体，其意图是可以满足的，或者说是可以实现的。即虽然在当前情况先，意图不能够得到满足，但是在不久的将来，意图会得到满足。

- **持续性**

持续性在联合意图中具有重要的作用。持续性是指智能体在执行的过程中，不会轻而易举的放弃自己的目标。持续性表明了智能体对于其所需要完成的任务具有某种承诺，该承诺智能体会尽其最大的能力完成任务。

- **一致性**

智能体在执行以意图的时候，其智能体之间的意图和智能体内内部的意图是一致的，否则，智能体将不知道如何进行动作的执行。

- **非冲突**

智能体的多个联合意图的执行，意图之间不会产生冲突。即某个联合意图的执行不会阻碍其他联合意图的执行。

- **与信念一致性**

智能体的意图与智能体的信念是一致的。如果智能体与具有某种联合意图，则智能体的信念中认为该意图是可以实现的。

### 3.6.1 逻辑框架

在这里，我们将基于联合意图的逻辑框架对多智能体系统进行分析。主要包括语法、模型和语义。

在这里我们采用形式化语义 $L$ ，该语言是分支时序逻辑的扩充。该语言主要由 $L_t$ 和 $L_s$ 组成。其中 $L_t$ 表示状态公式集合， $L_s$ 表示路径公式集合。 $\Phi$ 表示命题符号的集

合,  $Set_{ag}$ 表示智能体符号的集合。其中, 我们对其进行规定, 小写字母a, b, c, ...表示原子命题;  $\alpha, \beta, \dots$ 表示公式; x, y, ...表示智能体。

**定义 3.18 (语言L语法)** 形式化语言L由以下规则定义的最小封闭集合:

- 如果  $p \in \Phi$ , 则  $p \in L_t$ ;
- 如果  $\varphi, \psi \in L_t$ , 且  $x, y \in Set_{ag}$ , 则  $\neg\varphi, \varphi \wedge \psi, Bel(x, \varphi), MB(x, y, \varphi), AI(x, y, \varphi), MI(x, y, \varphi), MAI(x, y, \varphi), MAB(x, y, \varphi), WAC(x, y, \varphi), MAC(x, y, \varphi), JAI(x, y, \varphi), MMI(x, y, \varphi), MMB(x, y, \varphi), WMC(x, y, \psi), MMC(x, y, \psi), JMI(x, y, \psi), JI(x, y, \psi, \varphi) \in L_t$ 。
- $L_t \in L_s$ ;
- 如果  $\varphi, \psi \in L_s, x \in Set_{ag}$ , 则  $\neg\varphi, \varphi \wedge \psi, \psi Until \varphi, \varphi Until \forall \psi \in L_s$ ;
- 如果  $\varphi \in L_s$ , 则  $A_\varphi \in L_t$ 。

语言L的一个模型是  $\{T, \prec, U_{ag}, \pi, [], B, C\}$  组成。其中T表示多种智能体在每一时刻所对应的状态;  $\prec$ 描述了时间上的先后顺序;  $U_{ag}$ 表示智能体的集合;  $\pi$ 表示使得公式成立的时刻集合;  $[]$ 表示对智能体进行赋值; B表示信念; C表示智能体所选择的路径。

**定义 3.19 (路径)** 时刻t的一条路径集合  $S \subseteq T$ , 且满足:

1.  $t \in S$ ;
2.  $\forall t_1, t_2 \in S : (t_1 < t_2) \vee (t_1 = t_2) \vee (t_1 > t_2)$ ;
3.  $\forall t_1, t_2 \in S, t_3 \in T, (t_1 < t_2 < t_3) \Rightarrow t_3 \in S$ ;
4.  $\forall t_1 \in S, t_2 \in T, (t_1 < t_2) \Rightarrow (\exists t_3 \in S : (t_1 < t_3) \wedge \neg(t_3 < t_2))$ ;
5.  $t_1 \in S : (t = t_1) \vee (t < t_1)$ 。

**定义 3.20 (语言L的语义)** 1.  $M \models_t \text{ iff } t \in \pi(p)$ ;

2.  $M \models_t \varphi \wedge \psi \text{ iff } M \models_t \varphi \text{ 且 } M \models_t \psi$ ;
3.  $M \models_t \neg\varphi \text{ iff } M \not\models \varphi$ ;
4.  $M \models_t A_\varphi \text{ iff } \forall S : S \in S_t \Rightarrow M \models_{s,t} \varphi$ ;

5.  $M \models_t Bel(x, \varphi)$  iff  $\forall t' : (t, t') \in B([x]) \Rightarrow M \models_{t'} \varphi$ ;
6.  $M \models_{s,t} \varphi \wedge \psi$  iff  $M \models_{s,t} \varphi$  且  $M \models_{s,t} \psi$ ;
7.  $M \models_{s,t} \neg\varphi$  iff  $M \not\models_{s,t} \varphi$ ;
8.  $M \models_{s,t} \psi Until_{\varphi}$  iff  $\exists t' \in S : (t \leq t') : \text{且 } M \models_{s,t'} \varphi \text{ 且 } (\forall t'' : t \leq t'' \leq t' \Rightarrow M \models_{s,t''} \psi)$ ;
9.  $M \models_{s,t} \psi Until_{\varphi}$  iff  $\forall t' \in S : (t \leq t') : (\forall t'' : t \leq t'' \leq t' \Rightarrow M \models_{s,t''} \neg\varphi) \Rightarrow M \models_{s,t'} \psi$ ;
10.  $M \models_{s,t} \varphi$  iff  $M \models_t \varphi$ , 其中  $\varphi \in L_t$ ;
11.  $M \models_t MB(x, y, \varphi)$  iff  $M \models_t Bel(x, \varphi)$  且  $M \models_t Bel(y, \varphi)$ ;

其中, Until是“until”算子; F是存在时序量词, G为全称时序量词; A是全称路径算子, E是存在路径算子。

### 3.6.2 联合意图理论框架

我们对[4]中的基本框架进行扩展, 对多智能体环境中的智能体之间的信念和意图进行说明。

智能体的意图对应着智能体实现某个命题, 对应着智能体需要实现的目标或者是任务。在实现型联合意图中, 智能体通过对路径的选择, 最终达到目标。

在智能体合作中, 外部环境具有非常重要的影响。我们在这里我们将联合意图中智能体对环境的感知分别应用与实现型和维持型场景。在实现型中, 只要意图还没有最后实现, 那么智能体就会一直进行努力, 且在这个过程中智能体中的环境感存在智能体执行的条件。在维持型中, 智能体的每次动作都需要对当前的意图进行保持, 同时外部的环境感知满足智能体执行的条件。

$Aware(x, \varphi)$ 表示智能体x所感知到环境信息, 在当前情况下, 智能体可以继续执行意图 $\varphi$ 。

**定义 3.21 (实现型意图语义)**  $M \models_t AI(x, \varphi)$  iff  $M \models_t Bel(x, \neg\varphi) \wedge Aware(x, \varphi)$  且  $(\forall S : S \in C(x, t) \Rightarrow \models_{s,t} FBel(x, \varphi) \wedge FAware(x, \varphi))$ 。

上述的语义揭示了实现型语义的本质属性。我们将智能体意图视为智能体对世界轨迹的选择。在这个过程中, 智能体在时刻t确定意图 $\varphi$ 没有实现且当前的外部环

境 $Aware(x, \varphi)$ 满足智能体执行条件。最终智能体实现了意图 $\varphi$ ，在这个执行过程中，外部的环境应该一直都可以满足智能体执行条件，否则，即使智能体即使知道目标 $\varphi$ 是大家的联合意图，由于外部环境的改变，仍然不能够完成任务。

**定义 3.22 (维持型意图语义)**  $M \models_t MI(x, \varphi)$  iff  $(\forall S : S \in C(x, t) \Rightarrow M \models_{s,t} GBel(x, \varphi) \wedge FAware(x, \varphi))$

上述语义揭示了维持性语义的本质属性。智能体在维护意图的时候，首相在最开始的状态智能体认为目标已经实现，且当前外部环境满足维持目标的条件。随之时间的推移，智能体的每次状态的改变都有两个约束条件，即当前状态维持不变和当前的外部环境支持智能体自身条件，使得智能体可以完成维持目标的任务。

### 3.6.2.1 联合实现型意图

**定义 3.23 (共同实现型选择)**  $MAI(x, y, \varphi) = AI(x, \varphi) \wedge AI(y, \varphi)$

2个智能体具有共同的选择来实现 $\varphi$ ，但是这并不等同于他们具有联合实现型的意图 $\varphi$ 。为了形成联合意图，需要智能体之间意图互知和合作。

**定义 3.24 (互知实现型意图)**  $MAB(x, y, \varphi) = MB(x, y, AI(x, \varphi), AI(y, \varphi))$ 。

上述概念表明，智能体之间的意图互知，知道双方都有实现性意图 $\varphi$ 。

**定义 3.25 (弱实现型合作)**  $WAC = (Bel(x, \varphi) \wedge \neg Bel(x, Bel(y, \varphi)) \rightarrow AI(x, MB(x, y, \varphi))) \wedge (Bel(x, AG\neg\varphi) \wedge \neg Bel(x, Bel(y, AG\neg\varphi)) \rightarrow AI(x, MB(x, y, AG\neg\varphi)))$

弱实现型合作是指，智能体x和智能体y，如果x知道 $\varphi$ 并且x不知道y知道 $\varphi$ 成立，则x让双方都知道 $\varphi$ 成立；如果x知道 $\varphi$ 永远不可能成立，并且不知道y知道 $\varphi$ 永远不可能成立，则让双方都知道 $\varphi$ 永远不可能成立。

**定义 3.26 (实现型合作)**  $M \models_t MAC(x, y, \varphi)$  iff  $\forall S : S \in C(x, t) \Rightarrow M \models_{s,t} (MB(x, y, WAC(x, y, \varphi) \wedge WAC(y, x, \varphi)))Until\neg AI(x, \varphi) \text{ 且 } \forall S : S \in C(y, t) \Rightarrow M \models_{s,t} (MB(x, y, WAC(x, y, \varphi) \wedge WAC(y, x, \varphi)))Until\neg AI(y, \varphi)$

智能体x和智能体y具有实现型意图合作是指，智能体双方都知道自己会同对方合作，也知道对方会和自己合作，知道自己放弃目标 $\varphi$ 。

**定义 3.27 (联合实现型意图)**  $JAI(x, y, \varphi) = MAI(x, y, \varphi) \wedge MAB(x, y, \varphi) \wedge MAC(x, y, \varphi)$

上述定义非常清晰的表示了联合实现型意图的本质。首先需要智能体双方都具有相同的选择 $\varphi$ ；其次知道双方都有这样的选择 $\varphi$ ；最后知道双方非常有必要进行合作。

基于上述语义定义，我们得到了如下重要的属性。

**定理 3.1**  $\models JAI(x, y, \varphi) \rightarrow MB(x, y, \neg\varphi)$

这一定理解释了在联合意图实施的最开始，意图 $\varphi$ 在该时刻不成立。

**定理 3.2 (实现型联合意图一致性)**  $\models \neg(JAI(x, y, \varphi)) \wedge JAI(x, y, \varphi)$

即在实现型意图实施的过程中，不存在冲突。所有的智能体既一起实现意图 $\varphi$ ，也一起不实现 $\varphi$ ，这样的情况是不存在的。

**定理 3.3 (与智能体意图一致)**  $\models \neg(JAI(x, y, \varphi) \wedge (AI(x, \neg\varphi) \vee AI(y, \neg\varphi)))$

即智能体的联合意图同智能体内在的意图是一致的。

**定理 3.4 (联合意图可满足性)**  $\models JAI(x, y, \varphi) \rightarrow MB(x, y, EF\varphi)$

对于联合意图，智能体相信意图最终存在时间和路径实现最终的结果。

**定理 3.5 (与信念一致性)**  $\models \neg(JAI(x, y, \varphi) \wedge (Bel(x, \neg EF\varphi) \vee (Bel(y, \neg EF\varphi))))$

上述定理指出智能体中联合意图与智能体的意图是一致的。

**定理 3.6 (非冲突性)**  $\models JAI(x, y, \varphi) \wedge JAI(x, y, \psi) \rightarrow MB(x, y, E(F\varphi \vee F\psi))$

上述定理说明意图 $\varphi$ 同意图 $\psi$ 会以某时序得以实现。

### 3.6.2.2 联合维护型意图

**定义 3.28 (共同维护型选择)**  $MMI(x, y, \varphi) = MI(x, \varphi) \wedge MI(y, \varphi)$

共同维护型选择，首先需要智能体有共同的选择维持意图 $\varphi$ 。

**定义 3.29 (互知维护型意图)**  $MMB(x, y, \varphi) = MB(x, y, MI(x, \varphi), MI(y, \varphi))$ 。

上述概念表明，智能体之间的意图互知，知道双方都有维护性意图 $\varphi$ 。

**定义 3.30 (弱维护型合作)**  $WMC = (Bel(x, \neg\varphi) \wedge \neg Bel(x, Bel(y, \neg\varphi)) \rightarrow AI(x, MB(x, y, \neg\varphi)))$

弱合作的本质意义是使得合作的双方都知道合作的信息。

**定义 3.31 (维护型合作)**  $M \models_t MMC(x, y, \varphi)$  iff  $\forall S : S \in C(x, t) \Rightarrow M \models_{s,t} (MB(x, y, WMC(x, y, \varphi) \wedge WMC(y, x, \varphi))) \text{Until} \neg MI(x, \varphi)$  且  $\forall S : S \in C(y, t) \Rightarrow M \models_{s,t} (MB(x, y, WMC(x, y, \varphi) \wedge WMC(y, x, \varphi))) \text{Until} \neg MI(y, \varphi)$

智能体 $x$ 和智能体 $y$ 具有维护型意图合作是指，智能体双方都知道自己会同对方合作来维持意图，也知道对方会和自己合作，知道自己放弃目标 $\varphi$ 。

**定义 3.32 (联合维护型意图)**  $JMI(x, y, \varphi) = MMI(x, y, \varphi) \wedge MMB(x, y, \varphi) \wedge MMC(x, y, \varphi)$

上述定义非常清晰的表示了联合维护型意图的本质。首先需要智能体双方都具有相同的选择 $\varphi$ 意图进行维护；其次知道双方都有这样的选择 $\varphi$ 来进行维护；最后知道双方非常有必要进行合作。

**定理 3.7**  $\models JMI(x, y, \varphi) \rightarrow MB(x, y, \varphi)$

这一定理揭示了维护型联合意图一个非常重要的条件，即都认为 $\varphi$ 已经成立。

**定理 3.8 (维护型联合意图一致性)**  $\models \neg(JMI(x, y, \varphi)) \wedge JMI(x, y, \varphi)$

即在维护型意图实施的过程中，不存在冲突。所有的智能体既一起维护意图 $\varphi$ ，也一起不维护 $\varphi$ ，这样的情况是不存在的。

**定理 3.9 (与智能体意图一致)**  $\models \neg(JMI(x, y, \varphi) \wedge (MI(x, \neg\varphi) \vee MI(y, \neg\varphi)))$

即智能体的联合意图同智能体内在的意图是一致的。

**定理 3.10 (维护型联合意图可满足性)**  $\models JMI(x, y, \varphi) \rightarrow MB(x, y, EG\varphi)$

对于联合意图，智能体相信意图最终存在时间和路径实现最终的结果。

**定理 3.11 (与信念一致性)**  $\models \neg(JMI(x, y, \varphi) \wedge (Bel(x, \neg EG\varphi) \vee (Bel(y, \neg EG\varphi))))$

上述定理指出智能体中联合意图与智能体的意图是一致的。

**定理 3.12 (非冲突性)**  $\models JMI(x, y, \varphi) \wedge JMI(x, y, \psi) \rightarrow EG(\varphi \wedge \psi)$

上述定理说明意图 $\varphi$ 同意图 $\psi$ 存在一个轨迹，使得其恒成立。

### 3.7 相关研究

在多智能体的环境下,为了共同完成一项复杂的任务,智能体之间经常进行正式或者非正式的沟通。在多智能体协同之前,会进行早期的协调和合作,协同是一个长期的过程。Gray将协同描述为:“参与者看到了问题的不同的方面,并且可以建设性的利用这些不同点,跨越自身能力的限制来完成这个问题”[43]。

作为活跃的智能体,在与外部环境进行交互的同时,也动态的做出相应的决策。智能体可以通过外部环境的信息和智能体内部的信息来做出决策,并同时改变着自身所处的状态。在多智能体环境下,智能体协同完成一个或者是多个任务。换句话说,智能体内部的信息也是非常重要的。在多智能体合作的环境下,这些智能体协同的做出决策,以完成集体的目标。因此,当智能体在进行决策判断的时候,首先是考虑的全局的利益,从而对外部环境信息利用的不够。但是外部的环境信息在智能体有其自身的影响,因此,为了使得模型更加的实用,需要考虑外部环境的影响。

在多智能体的环境下,一个单独的智能体没有足够的知识和能力用以完成当前的任务。所有的智能体一起协作完成任务,但是这都超出了一个智能体所能承担的责任。每个智能体通过自己是传感器从外部的环境中获得相应的信息,感知到自己需要完成的任务,并通过自己内部的组合,对外部输出动作序列。Cohen提供了对Bratman提出模型中意图的三个有特点的角色,并且指出可以避免一些负面的效果[28]。Ferguson描述了协同的问题解决并对最初的混合计划。但是这些模型都关注在智能体信念的状态和推理,而忽略了外部环境对智能体的影响。COLLAGEN[87]和RavenClaw[22]主要关注任务的执行,但是在模型中缺乏明确的规划和信息交互。

作为智能体的一个模型,BDI认知模型得到了哲学领域和人工智能领域的认可。Brathman的理论被Cohen和Lecques进行了形式化。在他们的形式化理论中,意图根据他们暂时信念和目标生成。Rao和Georgeff提出了BDI模型中的可能世界模型[36]。他们提出的抽象模型主要包含三个动态变化的部分,即信念,愿望和意图,另外还包括一系列的事件。BDI模型这个结构,允许我们在这个三个结构上进行查询和排序工作。BDI模型可以识别的事件,包括内部的事件也包括外部的事件[47][83]。

Wooldridge和Lomuscio提出了一个多智能体的VSK逻辑,这个逻辑允许我们对环境表示。智能体在环境汇中可以知道什么是正确的、在环境中什么是可视的,别的智能体是否也感知到了环境和智能体到底从环境中得到了什么样的知识[126]。在多智能体环境下,群组感知就是对组内其它的智能体有自己的了解,并为自己的行动提供一个环境。群组感知可以简单的分为在多智能体环境中的其它智能体,其它智能体正在做些什么,其它智能体工作的地点[48]。当智能体得到这些信息后,智能体可以将它们的通信简化、更好的组织动作和预测别的智能体的动作,从而可以更加方便的和别的



智能体协同工作。

基于策略的管理是管理中的一种模式，根据系统的不同的功能分别进行管理。这种方式可以减少系统中的信息的保持代价和通信代价，并改善系统的灵活性和适应性。一个理性的智能体可以随着环境的变化而变化，根据其传感器感知到的信息和智能体系统内部存储的知识，智能体会向外部的环境给予相应的动作。Kephart等指出，一个具有反应能力的智能体，严格的按照产生是规则在所处的环境中动作。在这种模式下，理性智能体的动作是设计者事先设计好的，或者是智能体自己计算的。一个基于目标的智能体，当选定目标之后，规划系统内部的动作序列，当所有的动作序列计算完毕后，作用于外部的环境。一个基于效用函数的智能体，所有的动作是基于一个效用函数的，每次选择效用函数值最大的动作作为我们的选项。

#### 3.8 小结

本章提出了智能体ABGP协同模型。在这里面我们主要突出了环境感知在智能体感知世界中的重要作用。由于单个智能体的能力有限，对于复杂的任务，需要多个智能体协同来完成。在以往的多智能体模型中，主要考虑的是多个智能体之间如何协作，而忽略了环境对智能体的影响。为此，在智能体ABGP协同认知模型中，我们添加了环境感知。这样，智能体在进行决策的时候，不仅仅要考虑来自于智能体之间和智能体内部的信息，还要考虑到智能体自身所处在外部的环境信息。外部环境信息，通过智能体自身的传感器不断的感知数据信息，这些信息传达到智能体内部，同其它的信息一起，支撑智能体的动作决策过程。



## 第四章 基于智能体信息共享的协同认知

### 4.1 引言

描述逻辑 (Description Logic, DL) 是基于对象的知识表示形式化工具, 它吸收了KL-ONE的主要思想, 是一阶逻辑的可判定子集[25][94]。描述逻辑只能处理静态领域, 动态描述逻辑克服了描述逻辑只能在静态领域应用的局限, 极大的扩展了描述逻辑的应用范围。应用动态描述逻辑, 一方面可以对静态领域的知识进行刻画推理; 另外一方面, 可以将静态知识作为背景, 在其基础上进行动作的描述和推理。动态描述逻辑能力强于命题语言的逻辑系统。描述逻辑系统主要有四部分组成: 概念和关系的构造集, TBox断言, ABox断言, TBox和ABox上的推理机制。描述逻辑的推理能力, 主要取决于这几个要素的选择和假设[98]。

由于语义网是一个分布式系统, 动态描述逻辑只适合处理同构信息, 不能为语义网提供合理的逻辑基础, 为此提出了分布式动态描述逻辑 (Distributed Dynamic Description Logic, D3L)。D3L是在动态描述逻辑上进行的扩展, 该逻辑为分布式、异构的信息集成提供了理论基础。分布式动态描述逻辑是对传统描述逻辑进行的扩充[24][23]。分布式动态描述逻辑D3L充分考虑了动态描述逻辑在分布式环境下的特性, 用于形式化的描述若干本地知识库所组成的环境。每个动态描述逻辑都有自己的表示形式和推理机制, 分布式动态描述逻辑是研究多个动态描述逻辑系统的统一表示形式和推理机制。

分布式动态描述逻辑通过桥规则进行知识传播, 但是它只针对两个局部DDL本体通过桥规则进行连接的情况, 而没有将知识在本体间的传播用于分布式推理。知识传播是分布式动态描述逻辑区别于传统动态描述逻辑的主要特征。多个DDL智能体之间在桥规则构成链的情况下, 并不总是按照预期的方式进行传播, 为此引入组合一致性语义, 提出了支持链式桥规则的分布式动态描述逻辑 (CD3L)。CD3L组成为3部分, 分布式TBox (DTB)、分布式ABox (DAB) 和分布式ActBox (DActB)。从而可以更好的为分布式信息检索提供逻辑基础[123][124][85]。

在智能体中, 符号可以用来表示知识。在现实的世界中, 由于每个智能体的能力都是有限的, 对于较大的任务来说, 单个的智能体不足以完成任务, 因此, 需要多个智能体协作完成。分布式动态描述逻辑是对动态描述逻辑的扩展, 且其具有广泛的应用。

这对这种情况, 本章研究在由多个智能体之间基于信息共享的协同认知理论。通

过CD3L，我们可以方便的进行知识的传播和检索，从而方便了多智能体之间的信息传输和信息检索。

本章分别给出了各个部分语法和语义。针对动态描述逻辑系统中存在的知识不能按照预期的方式进行传播的特点，引入了组合一致性语义，并研究了CD3L 的推理机制。

## 4.2 知识传播

包含传播是形式化描述D3L的知识重用机制，因而是D3L的重要特性之一。以下给出了包含传播的基本形式。通过桥规则，智能体之间可以进行知识的定向传播。

**定义 4.1 (DTB知识传播)** 给定分布式  $TBox\ DTB$ ，如存在  $i : C \sqsupseteq j : E$  以及  $i : D \sqsubseteq j : F$ ，则有以下式成立： $DTB \models_d i : C \sqsubseteq D \Rightarrow DTB \models_d j : E \sqsubseteq F$ 。

**定义 4.2 (DActB知识传播)** 给定分布式  $ActBox\ DActB$ ，如存在  $i : \alpha \sqsupseteq j : \beta$  以及  $i : \pi \sqsubseteq j : \rho$ ，则有以下式成立： $DActB \models_d i : \alpha \sqsubseteq \pi \Rightarrow DActB \models_d j : \beta \sqsubseteq \rho$ 。

上面的定理只适用于两个DDL智能体通过桥规则进行连接的情况。当多个DDL的智能体进行连接的时候，存在知识不能够按照预期传播的情况。如图4.1所示

在图4.1(a)中， $i : A3 \sqsubseteq j : A2$ ， $i : B3 \sqsupseteq j : B2$ ，由  $i : B3 \sqsubseteq i : A3$ ，根据传播定理得： $j : B2 \sqsubseteq j : A2$ ，同理可以得到  $k : B2 \sqsubseteq k : A2$ 。

但是在图4.1(b)中语义并没有按照我们想象的传播下去，按照传统的包含传播定理，并不能够得出  $k : B1 \sqsubseteq k : A1$ 。因为，它并不满足知识传播的条件。

针对此问题，赵晓非等[128]提出了组合一致性链式桥规则导致的D3L知识传播，对D3L语义进行了扩展。定义如下：

**定义 4.3 (组合一致性)** 给定分布式解释  $\mathfrak{DI}$ ，对于任意的  $i, j, k \in \mathbb{I}$  及  $x \in \Delta^{I_i}$ ， $r_{ij}(x) = d$  成立蕴含  $r_{jk}(d) = r_{ik}(x)$ ，则称领域关系  $r$ （即  $\mathfrak{DI}$ ）满足组合一致性。

在组合一致语义的基础之上，图4.1(b)的解释如下： $r_{jk}(r_{ij}(A3^{I_i})) \sqsubseteq r_{jk}(A2^{I_j}) \sqsubseteq A1^{I_k}$ ， $B1^{I_k} \sqsubseteq r_{j'k}(B2^{I_{j'}}) \sqsubseteq r_{j'k}(r_{ij'}(B3^{I_i}))$ 。组合一致性语义意味着  $r_{ik}(A3^{I_i}) = r_{jk}(r_{ij}(A3^{I_i}))$  以及  $r_{j'k}(r_{ij'}(B3^{I_i})) = r_{ik}(B3^{I_i})$ 。根据  $DTB \models_d i : B3 \sqsubseteq A3$ ，得  $r_{ik}(B3) \sqsubseteq r_{ik}(A3)$ ，从而得  $B1 \sqsubseteq A1$ 。

将组合一致性链式桥规则应用于分布式动态描述逻辑（D3L）系统，弥补了传统分布式动态描述逻辑系统的不足。

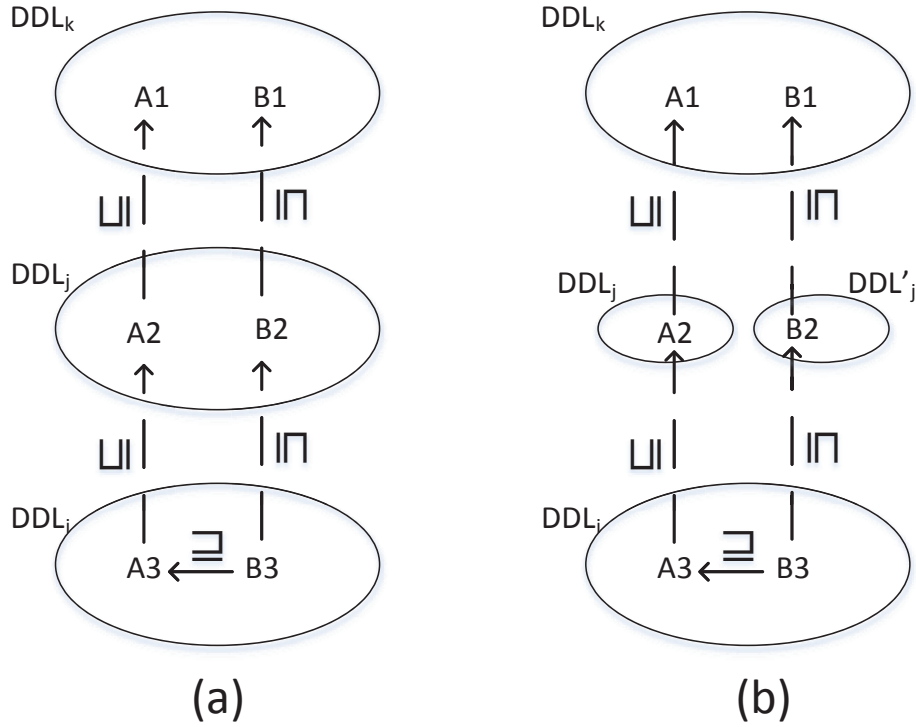


图 4.1: 分布式知识传播

### 4.3 CD3L知识库

本节给出支持链式桥规则的分布式动态描述逻辑 (CD3L) 的基本定义。假定  $DDL_i$ 、 $DDL_j$  和  $DDL_k$  是动态描述逻辑系统，每个动态描述逻辑系统都有自己的表示形式和推理机制。CD3L 研究多个动态描述逻辑系统的统一表示形式和推理。

#### 4.3.1 CD3L语法

CD3L 系统包括 5 个基本组成部分：分布式动态描述逻辑，如  $DDL_i$ 、 $DDL_j$ 、 $DDL_k$  等；分布式 TBox，分布式 ABox，分布式 ActBox；分布式 Tbox、ABox 和 ActBox 上的推理机制。

令  $\mathbb{I}$  是序号的非空集合，对于每个  $i \in \mathbb{I}$ ， $DDL_i$  表示一个独立的动态描述逻辑系统。一个序列  $KB = \{DDL_i\}$  且  $i \in \mathbb{I}$ ，组成了一个分布式动态描述逻辑系统。在  $DDL_i$  系统中的概念  $C$  前标注  $i$ ， $i:C$  表示  $DDL_i$  系统的概念  $C$ ；在  $DDL_i$  系统中的动作  $\alpha$  前标注  $i$ ， $i:\alpha$  表示  $DDL_i$  系统的动作  $\alpha$ 。

**定义 4.4 (桥规则)** 给定  $DDL_i$  的概念  $C$  和  $DDL_j$  的概念  $E$ ， $DDL_i$  的动作  $\alpha$  和  $DDL_j$  的动作  $\beta$ ，则  $i$  到  $j$  的桥规则是下列四种形式之一：

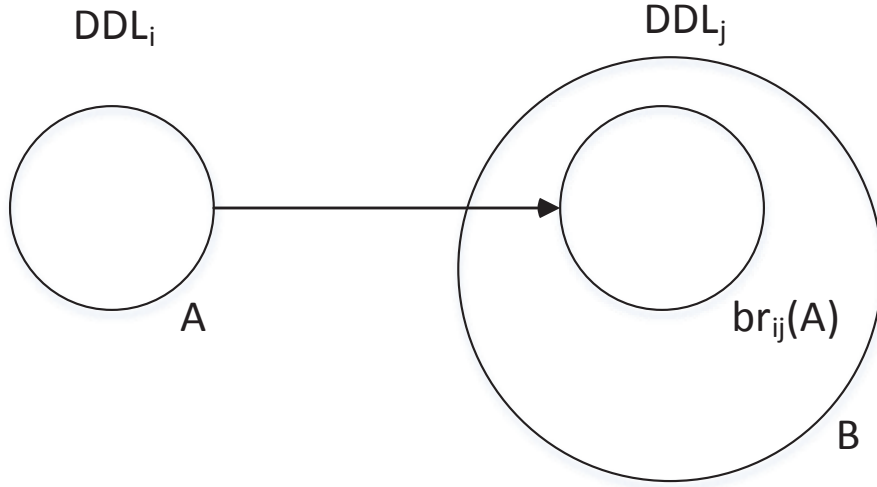


图 4.2: into桥规则

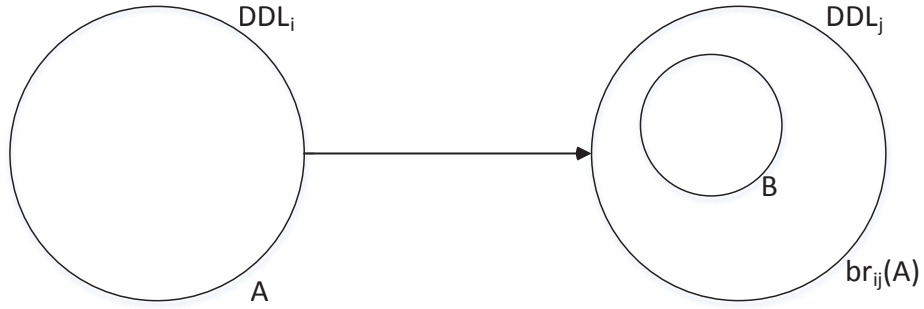


图 4.3: onto桥规则

1.  $i : C \xrightarrow{\sqsubseteq} j : E : into$  概念桥规则
2.  $i : C \xleftarrow{\sqsupseteq} j : E : onto$  概念桥规则
3.  $i : \alpha \xrightarrow{\sqsubseteq} j : \beta : into$  动作桥规则
4.  $i : \alpha \xleftarrow{\sqsupseteq} j : \beta : onto$  动作桥规则

桥规则具有方向性，即从i到j的桥规则 $br_{ij}$ 并不一定是从j到i的桥规则 $br_{ji}$ 的逆。

如图4.2所示为into桥规则， $DDL_i$ 中的A通过 $br_{ij}$ 映射到 $DDL_j$ 中的B，A为B 的子集。

如图4.3，表示onto桥规则， $DDL_i$ 中的A通过桥 $br_{ij}$ 映射到 $DDL_j$ 中的B，B为A的子集。

**定义 4.5 (对应关系)** 如果 $x$ 是动态描述逻辑 $DDL_i$ 的个体,  $y_1, y_2, \dots, y_n$ 是动态描述逻辑 $DDL_j$ 中的个体, 则有:

1.  $i : x \rightarrow j : y$ :称为部分个体对应
2.  $i : x \xrightarrow{=} j : y$ : 称为完全个体对应

D3L利用桥规则和个体对应来建立多个动态描述逻辑系统(DDL)之间的应对关系。

**定义 4.6 (分布式TBox)**  $\mathbb{I}$ 是顺序号的非空集合, 一个分布式TBox  $DTB = \langle \{T_i\}, \mathfrak{B} \rangle$   $i \in \mathbb{I}$ 。包含一个TBox的集合 $\{T_i\} i \in \mathbb{I}$ 和从 $T_i$ 到 $T_j$ 的桥规则集合 $\{\mathbb{B}_{ij}\}$ 且 $i \neq j \in \mathbb{I}$ 。对于任意的 $k \in \mathbb{I}$ , 所有在 $T_k$ 中的描述必须同 $DDL_k$ 相对应。对于每一个概念桥规则 $\mathfrak{B}_{ij}$ , 有 $i : C \xrightarrow{=} j : E$  或者  $i : C \xrightarrow{\neq} j : E$ , 概念 $C$ 必须在 $DDL_i$ 中存在且概念 $E$ 必须在 $DDL_j$ 中存在。

**定义 4.7 (分布式ABox)**  $\mathbb{I}$ 是顺序号的非空集合, 一个分布式ABox  $DAB = \langle \{A_i\}, \mathfrak{C} \rangle$   $i \in \mathbb{I}$ , 包含一个ABox的集合 $\{A_i\} i \in \mathbb{I}$ 和从 $A_i$ 到 $A_j$ 的部分或者是全部的个体对应集合 $\{\mathfrak{C}_{ij}\}$ 且 $i \neq j \in \mathbb{I}$ 。对于任意的 $k \in \mathbb{I}$ , 所有在 $A_k$ 中的描述必须同 $DDL_k$ 相对应。对于每一条对应规则,  $\mathfrak{C}_{ij}$ 中的 $i : x \rightarrow j : y$ 部分个体对应或者是 $i : x \xrightarrow{=} j : y$ 完全对应, 个体名字 $x$ 必须存在于 $DDL_i$ 中, 而个体名字 $y$ 必须存在于 $DDL_j$ 中。

**定义 4.8 (分布式ActBox)**  $\mathbb{I}$ 是顺序号的非空集合, 一个分布式ActBox  $DActB = \langle \{Act_i\}, \mathfrak{D} \rangle$   $i \in \mathbb{I}$ 。包含一个ActBox的集合 $\{Act_i\} i \in \mathbb{I}$ 和从 $Act_i$ 到 $Act_j$ 的桥规则集合 $\{\mathbb{D}_{ij}\}$ 且 $i \neq j \in \mathbb{I}$ 。对任意的 $k \in \mathbb{I}$ , 所有在 $Act_k$ 中的描述必须同 $DDL_k$ 相对应。对于每一个动作桥规则 $\mathfrak{D}_{ij}$ 有 $i : \alpha \xrightarrow{=} j : \beta$  或者  $i : \alpha \xrightarrow{\neq} j : \beta$ , 动作 $\alpha$ 必须在 $DDL_i$ 中且 $\beta$ 必须在 $DDL_j$ 中。

**定义 4.9 (分布式知识库)** 一个支持桥规则的分布式动态描述逻辑CD3L知识库 $DKB = \langle DTB, DAB, DActB \rangle$ 。其中 $DTB$ 是一个分布式TBox,  $DAB$ 是一个分布式ABox,  $DActB$ 是一个分布式ActBox。

### 4.3.2 CD3L语义

由于CD3L是由多个动态描述逻辑通过桥规则或者是个体对应而成, 因而CD3L的语义解释也可以根据DDL的语义和论域连接关系来给出。

**定义 4.10 (解释)** 一个分布式解释 $\mathfrak{DI} = \langle \{I_i\}_{i \in \mathbb{I}}, r \rangle$ , 其中 $I_i$ 是分布式动态描述逻辑 $DDL_i$ 的在解释域 $\Delta^{I_i}$ 语义解释, 二元关系 $r_{ij} \in \Delta^{I_i} \times \Delta^{I_j}$ 。

**定义 4.11 (DTB解释)** 一个分布式解释  $\mathfrak{DI} = \langle \{I_i\}_{i \in \mathbb{I}}, r \rangle$  满足分布式  $TBox$   $DTB = \langle \{T_i\}_{i \in \mathbb{I}}, \mathfrak{B} \rangle$  中的元素或者是公理, 记作  $\mathfrak{DI} \models_d$ , 对于  $i, j \in \mathbb{I}$  有:

1.  $\mathfrak{DI} \models_d i : C \sqsubseteq j : E$ , 如果  $r_{ij}(C^{I_i}) \subseteq E^{I_j}$  成立;
2.  $\mathfrak{DI} \models_d i : C \sqsupseteq j : E$ , 如果  $r_{ij}(C^{I_i}) \supseteq E^{I_j}$  成立;
3.  $\mathfrak{DI} \models_d i : C \sqsubseteq E$  成立, 如果  $I_i \models C \sqsubseteq E$  成立;
4.  $\mathfrak{DI} \models_d T_i$ , 如果  $I_i \models T_i$  成立;
5.  $\mathfrak{DI} \models_d DTB$ , 如果对于任意的  $i \in \mathbb{I}$ ,  $I_i \models T_i$ , 并且  $\mathfrak{DI}$  满足  $\mathfrak{B}$  中的每一条规则;
6.  $\mathfrak{DI} \models_d i : C \sqsubseteq E$ , 如果对每一个分布式解释  $\mathfrak{DI}$  有,  $\mathfrak{DI} \models DTB$  蕴含  $\mathfrak{DI} \models_d i : C \sqsubseteq E$ .

**定义 4.12 (DAB解释)** 一个分布式解释  $\mathfrak{DI} = \langle \{I_i\}_{i \in \mathbb{I}}, r \rangle$  满足分布式  $ABox$ ,  $DAB = \langle \{A_i\}_{i \in \mathbb{I}}, \mathfrak{C} \rangle$  中的元素或者是公理, 记作  $\mathfrak{DI} \models_d$ , 对于任意的  $i, j \in \mathbb{I}$  有:

1.  $\mathfrak{DI} \models_d i : x \rightarrow j : y$ , 如果有  $y^{I_j} \in r_{ij}(x^{I_i})$  成立;
2.  $\mathfrak{DI} \models_d i : x \rightarrow j : \{y_1, y_2, \dots, y_n\}$ , 如果有  $r_{ij}(x^{I_i}) \in \{y_1^{I_j}, y_2^{I_j}, \dots, y_n^{I_j}\}$  成立;
3.  $\mathfrak{DI} \models_d C(a)$ , 如果有  $I_i \models C(a)$  成立;
4.  $\mathfrak{DI} \models_d p(a, b)$ , 如果有  $I_i \models p(a, b)$  成立;
5.  $\mathfrak{DI} \models_d A_i$ , 如果对  $A_i$  的任意断言  $C(a)$  和  $p(a, b)$ , 有  $\mathfrak{DI} \models_d C(a)$  和  $\mathfrak{DI} \models p(a, b)$  成立;
6.  $\mathfrak{DI} \models_d DAB$  如果对于任意的  $i \in \mathbb{I}$ ,  $\mathfrak{DI} \models_d A_i$ , 并且  $\mathfrak{DI}$  满足  $\mathfrak{C}$  中的所有个体;
7.  $DAB \models_d i : C(a)$ , 如果对于任意分布式解释  $\mathfrak{DI}$ , 有  $\mathfrak{DI} \models_d DAB$  蕴含  $\mathfrak{DI} \models_d C(a)$  成立;
8.  $DAB \models_d i : p(a, b)$ , 如果对于任意分布式解释  $\mathfrak{DI}$ , 有  $\mathfrak{DI} \models_d DAB$  蕴含  $\mathfrak{DI} \models_d p(a, b)$  成立;

**定义 4.13 (DActB解释)** 一个分布式解释  $\mathfrak{DI} = \langle \{I_i\}_{i \in \mathbb{I}}, r \rangle$  满足分布式  $ActBox$   $DActB = \langle \{Act_i\}_{i \in \mathbb{I}}, \mathfrak{D} \rangle$  中的元素或者是公理, 记作  $\mathfrak{DI} \models_d$ , 对于  $i, j \in \mathbb{I}$  有:

1.  $\mathfrak{DI} \models_d i : \alpha \sqsubseteq j : \beta$ , 如果  $r_{ij}(\alpha^{I_i}) \subseteq (\beta^{I_j})$  规则成立;



2.  $\mathfrak{DI} \models_d i : \alpha \sqsupseteq j : \beta$ , 如果  $r_{ij}(\alpha^{I_i}) \sqsupseteq (\beta^{I_j})$  规则成立;
3.  $\mathfrak{DI} \models_d i : \alpha \sqsubseteq \beta$ , 如果  $I_i \models_d \alpha \sqsubseteq \beta$  成立;
4.  $\mathfrak{DI} \models_d Act_i$ , 如果  $I_i \models_d Act_i$  成立;
5.  $\mathfrak{DI} \models_d DActB$ , 如果对于任意的  $i \in \mathbb{I}$ ,  $I_i \models_d Act_i$  并且  $\mathfrak{DI}$  满足  $Act_i$  中的每条规则;
6.  $DTB \models_d i : \alpha \sqsubseteq \beta$ , 如果对每一个分布式解释  $\mathfrak{DI}$  有,  $\mathfrak{DI} \models_d ActB$  蕴含  $\mathfrak{DI} \models_d i : \alpha \sqsubseteq j : \beta$  成立。

**定义 4.14 (分布式知识库解释)**  $\mathfrak{DI} = \langle \{I_i\}_{i \in \mathbb{I}}, r \rangle$ , 满足  $CD3L$  的知识库

$DKB = \langle DTB, DAB, DActB \rangle$  记为  $DTB \models_d DKB$ , 当且仅当  $\mathfrak{DI} \models_d DTB$ ,  $\mathfrak{DI} \models_d DAB$  和  $\mathfrak{DI} \models_d DActB$ 。

#### 4.4 支持链式桥规则的分布式动态描述逻辑推理

$CD3L$  的推理可以分为局部推理和全局推理两部分。局部推理指  $CD3L$  中的某个动态描述逻辑推理, 可以直接化为某个动态描述逻辑的推理, 所以局部的  $CD3L$  推理可以直接采用文献中的推理方法。

全局推理指利用桥规则和个体对应规则在所有的动态描述逻辑上进行推理。本文主要介绍  $CD3L$  的全局推理算法。本文介绍的  $CD3L$  推理主要是利用动态描述逻辑的推理方法、桥规则和个体对应来实现  $CD3L$  的推理, 该方法是 Serafini[95] 等人提出的 tableaux 推理算法的推广。

$CD3L$  直接推理算法考虑了两种情况, 一种是不考虑链式桥规则的情况, 一种是考虑链式桥规则的情况。

$\mathbb{I}$  是序号的非空集合, **selected** 是已经选择的顺序号集合,  $\mathbb{I}/\mathbf{selected}$  表示将 **selected** 集合从  $\mathbb{I}$  中除去。

##### 4.4.1 $CD3L$ 算法的一致性

算法7调用8。在算法的推理的过程中, 主要考虑了三种情况下的推理:

- 在单个  $DDL$  中进行推理。此种情况, 考虑的是简单的情况。在这种情况下, 单个智能体内的知识就可以将我们对所需要的知识进行推理。由于我们所需要检测的知识在单个的描述逻辑中就可以实现, 并不需要借助桥规则实现。此时, 算法退化为最简单的形式。

- 由于单个动态描述逻辑系统中的知识有限，在某些情况下我们并不能够在单一的动态描述逻辑系统中将其推导出来。在此种情况下，系统可以借助桥规则进行推理。系统首先查找与现有的动态描述逻辑同时存在into和onto规则动态描述逻辑系统。然后依次对候选系统进行测试。只要找到满足条件的候选系统，此时测试结束，返回结果为真。如果所选择的候选系统不能够测试出当前的信息情况，则对剩余的进行测试。当候选系统中所有的候选着都测试完毕之后，如果公理可以推导出来，则返真，否则继续进行下一步的推导。
- 如果经过第二步，任然不能够将公推出公理成立，则继续选择候选系统。其中一个into桥规则 $DDL_j$ 与 $DDL_i$ 系统相关联，一个onto桥规则 $DDL_{j'}$ 与 $DDL_i$ 相关联，然后继续选择一个候选系统，该系统存在到 $DDL_i$ 的onto桥规则，同时存在到 $DDL_{j'}$ 的into桥规则。找出多个这种的候选系统，然后进行推理测试。如果系统可以成功进行推理，则返回推理结果。否则，返回FALSE，表示现有的知识不能够进行公式的推理。

#### 4.4.2 一致性讨论

断言公式的一致性问题是CD3L推理中的最基本问题，其它推理的问题都可以转化为这种推理或者是利用这种推理的结果。一个断言公式是一致的，即表示该集合中不存在冲突。

一个公式集是一致的，当且仅当公式集不存在冲突，否则称公式集是不一致的。一个冲突具有如下形式之一，其中C是概念，R是任意关系，a、b为任意常元：

- $\{\perp(a)\}$
- $\{C(a), \neg C(a)\}$
- $\{R(a, b), \neg R(a, b)\}$

一致性需要我们对公式的一致性进行检测。对于断言公式的一致性检测，主要是利用推理规则对需要推理的公式进行扩充。然后在对扩充后的公式进行检测。由于动态描述逻辑还可以进行动作的表示，因此，还需要对动作的一致性进行检测。动作描述是一般性的描述，因此，在检测的时候我们需要对动作的实例记性构造。当公式的构造完毕之后，整个系统的检测又转化为断言公式的一致性检测问题。

对于第一种形式，在公式系统 $DDL_i$ 中，根据自身的知识可以进行相应的推理判断。从而整个系统简化为一个标准的DDL系统。而DDL系统的一致性已经由常亮[3]进行证明。

对于第二种形式, 公式系统在 $DDL_i$ 中不能够根据自身的内在知识进行推理。迭代选择同时具有into规则和onto规则与 $DDL_i$ 进行连接的系统进行推理。此种类型的推理已经由王竹晓[122]进行论证, 系统的一致性得到了保证。

对于第三种形式, 公式在 $DDL_i$ 系统中不能够进行推理。系统分别选择具有只具有into桥规则和onto桥规则与系统 $DDL_i$ 相连接的系统, 然后在选择同时与into桥规则和onto桥规则进行相关联的系统。此种类型推理的一致性已经被赵晓非[128]等进行了证明。

根据以上分析, 系统的一致性得到了保证, 即系统中不存在矛盾。该算法一致性得到了保证, 从而算法可以在有效的时间内运行完毕。

### 4.5 小结

本章提出了一种基于分布式动态描述逻辑的推理算法。动态描述逻辑可以用来组成智能体内部的知识系统。对于我们需要推理的知识, 首先我们在单个动态描述逻辑系统中进行推理。如果不能得出结果, 我们选择同时和待推理系统具有into桥规则和onto桥规则的系统进行推理。如果在此种情况下, 仍然不能进行推理, 我们继续尝试第三种情况。首先选择与待推理系统分别只具有into桥规则和onto桥规则的系统, 然后在选择同时与这两个系统连接的系统。此时, 可以应该四个动态描述逻辑系统继续进行推理。与传统的分布式动态描述逻辑相比, 该CD3L支持知识的传播, 可以为语义网提供更好的逻辑基础。

---

**Algorithm 7** CD3L Reasoning.

---

**Input:**

Multiple DDL systems  $\{DDL_i\}_{i \in \mathbb{I}}$ ;  
 bridge rules  $\{\mathfrak{B}_{ij}\}_{ij \in \mathbb{B}}$ ;  
 judge whether  $i : C \sqsubseteq i : D$ ;

**Output:**

Boolean;

```

1: if ( $T_i \models_d i : C \sqsubseteq i : D$ ) then
2:   return TRUE;
3: end if
4: if (Candidate(selected,  $i, \mathbb{I}$ )) then
5:   return TRUE;
6: end if
7: //(onto bridge, into bridge) to  $DDL_i$ 
8:  $Q = \{(a_1, b_1), \dots, (a_n, b_n)\}$ 
9: while ( $Q \neq \phi$ ) do
10:  TMP = DequeueHead(Q);
11:  E = TMP.first;
12:  F = TMP.second;
13:  selected = {};
14:  if ( ( $i : C \sqsubseteq m : E$ ) && ( $n : F \sqsubseteq i : D$ ) && ( $m : E \sqsubseteq j : G$ ) && ( $j : H \sqsubseteq n :$ 
       $F$ ) &&  $i \neq j$  &&  $j \neq m$  &&  $m \neq n$  ) then
15:    if ( $T_i \models_d j : G \sqsubseteq j : H$ ) then
16:      return TRUE;
17:    else
18:      selected = selected  $\sqcup \{E, F, i, j\}$ 
19:      if (CandidateTwo(selected, E, F,  $\mathbb{I}$ )) then
20:        return TRUE;
21:      end if
22:    end if
23:  end if
24: end while
25: return FALSE;

```

---

---

**Algorithm 8** CD3L Candidate.

---

**Input:**

selected Set selected;

pervious index k;

index Set  $\mathbb{I}$ ;

**Output:**

Boolean;

```

1: if ( $j \in \mathbb{I} \setminus \text{selected}$  && exist onto bridge and into bridge to k) then
2:   if ( $T_j \models_d j : E \sqsubseteq j : F \Rightarrow i : C \sqsubseteq i : D$ ) then
3:     return TRUE;
4:   else
5:     selected = selected  $\sqcup$  j;
6:     if ( $\mathbb{I} \setminus \text{selected} \neq \phi$ ) then
7:       Candidate(selected, k,  $\mathbb{I}$ );
8:     else
9:       return FALSE;
10:    end if
11:  end if
12: end if
13: return FALSE;

```

---

---

**Algorithm 9** CD3L CandidateTwo.**Input:**

selected Set selected;  
pervious index E;  
pervious index F;  
index Set  $\mathbb{I}$ ;

**Output:**

Boolean;

```
1: if ( $j \in \mathbb{I} \setminus \text{selected}$  && exist onto bridge to E and into bridge to F) then  
2:   if ( $(T_j \models_d j : G \sqsubseteq j : H \Rightarrow i : C \sqsubseteq i : D)$ ) then  
3:     return TRUE;  
4:   else  
5:     selected = selected  $\sqcup$  j;  
6:     if ( $\mathbb{I} \setminus \text{selected} \neq \phi$ ) then  
7:       CandidateTwo(selected, E, F,  $\mathbb{I}$ );  
8:     else  
9:       return FALSE;  
10:    end if  
11:  end if  
12: end if  
13: return FALSE;
```

---

## 第五章 智能体协同认知原型系统

### 5.1 引言

近年来，随着计算机技术的不断发展，其应用领域在不断的进行扩展。随着计算机网络通信技术的进步，计算机软硬件系统的，系统的设计越来越偏向智能化的方向发展。作为人工智能的分支，智能体的研究越来越受到重视。在现实生活中，存在着大量的问题，这样的问题超过了单个智能体自身的能力，不能自己独立完成。为此，需要多个智能体共同协作完成。

从20实际80年代起，对于多智能体呈明显增长的趋势。人们对于多智能体的研究，已经成为人工智能领域中研究的热点。在多智能体系统中，每一个智能体是一个实体，该实体或者是物理的，或者是抽象的。智能体能作用于外部的自然环境，同时能与其它智能体进行应有的通信。通过智能体之间的通信，智能体之间可以进行信息的交换，通过智能体之间的协作，可以共同完成复杂艰巨的任务。

当我们在开发这样一种系统的时候，由于智能体所处的环境的复杂性，我们需要考虑较多的方面，例如：

外部环境一直在进行着动态的变化，智能体所处的环境进行着动态的变化，为了快速对外的环境进行响应，智能体需要对外部信息进行实时的感知，对于紧急的事物需要进及时的处理；

由于内部和外部环境的变化，智能体会在自己的内部生成相应的动机。智能体中的执结构需要对动机进行相应，因而，我们需要对动机进行相应的建模。

对于复杂的任务，需要多个智能体通过协作来完成。这样，需要智能体之间进行信息的交换。

在本文中，我们将具体讨论该原型的结构、功能、以及实现细节。该方法的主要优点总结如下：

- 对内部和外部感知到的环境信息，对动机建立起自身的模型，便于智能体进行计算；
- 当智能体选择需要执行的动作时候，检索智能体内部的情景记忆中存储的知识，对动作进行规划；
- 对多智能体信息进行推理，该推理过程建立在CD3L上。

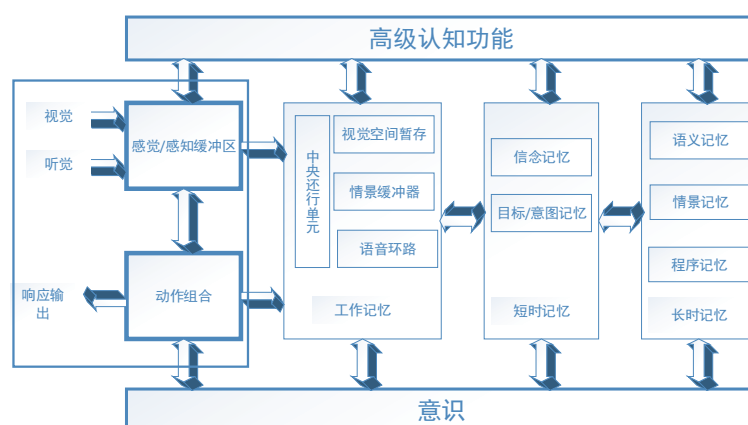


图 5.1: Agent-0型智能体

## 5.2 智能体类型

在图3.1，标识了整个CAM的逻辑框架。该框图是整个CAM系统的实现框架。图3.1详细描述了CAM系统的结构。从中可以看出，CAM系统主要是有感知模块、动作模块、工作记忆、短期记忆、长期记忆、意识以及高层认知组成。感知模块主要是用来多外部的环境进行感知，可以感知的信息如视觉和听觉。工作记忆主要是智能体进行推理规划。短期记忆实现了BDI推理系统，长期记忆主要存储系统中的历史知识，提供智能体在推理的过程中所需要的知识。但是为了方便系统的开发，我们提出了多种类型的智能体类型。

Aent-0型智能体，如图5.1所示： Agent-0型智能体主要是突出了智能体对环境的快速响应。Agent-0型智能体，在满足质量标准的情况下，进行快速的反应。Agent-0型智能体是反应智能体。反应智能体中不包含有符号表示的世界模型，同时也不包含复杂的符号推理过程[7]。在感知到外部的信息之后，智能体快速的对外部的环境进行响应。这种类型的智能体适合环境动态变化，但是环境所处的状态和历史中的信息相吻合的情况。

Agent-1型智能体是在Agent-0的基础之上添加了执行处理单元和高级认知功能，从而使得系统具有了推理功能。外界环境的信息通过智能体的感知通道输入到智能体的执行单元中，执行简单的高级功能。对于智能体选定的目标，通过智能体通过中央执行单元进行进行处理，然后想外部的环境输出自己的动作序列。CAM-1如图5.2所示

在Agent-1的基础之上，我们将意识和BDI推理引入到智能体的组成部分，成为我们的Agent-2型智能体。意识是智能体一种主观的概念，是对外部世界、自己的身体以及心理过程体验的整合。BDI是典型的心理学的智能体结构，可以用来进行推理。如图5.3所示



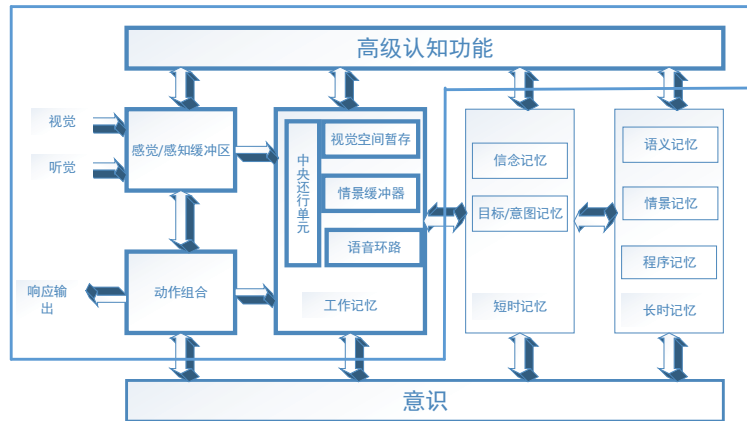


图 5.2: Agent-1型智能体

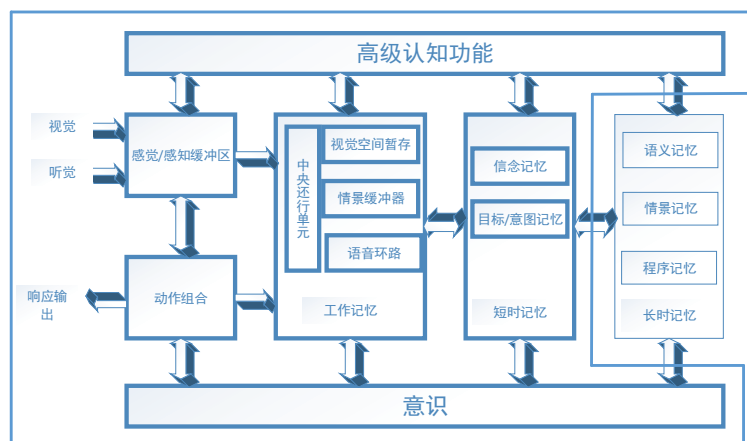


图 5.3: Agent-2型智能体

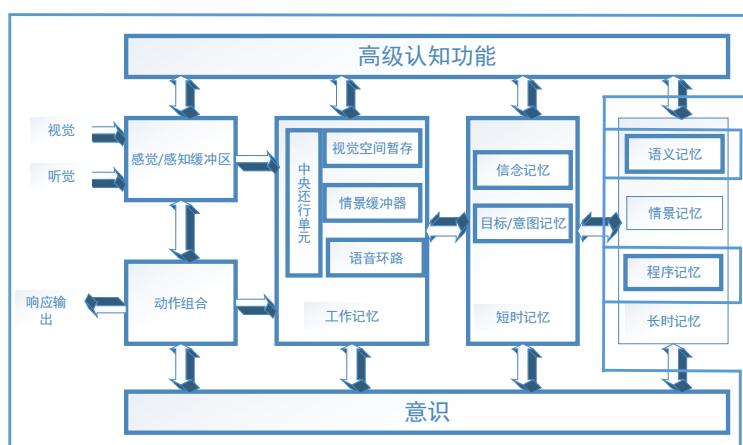


图 5.4: Agent-3型智能体

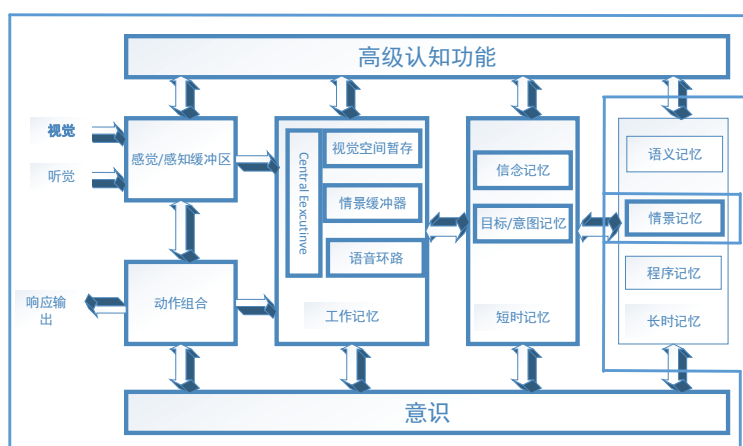


图 5.5: Agent-4型智能体

在Agent-2的基础之上，我们将长期记忆中的语义记忆和程序性记忆添加上，即成为我们的Agent-3型智能体。在Agent-3智能体中，由于有了程序性记忆和语义记忆，智能体在用进行推理的时候，访问智能体中以前的知识，加快智能体的推理过程和准确度。Agent3型智能体主要用来模拟大脑的抽象思维。如图5.4所示

在Agent-2型智能体的基础之上，我们将长期记忆中的情景记忆添加之上，则成为我们的Agent-4型智能体。Agent-2型智能体的典型特征是智能体在推理的过程中，可以借助以往历史存储的情景记忆进行动作的推理。在Agent-4中，情景记忆存储的是事件的动作序列，即事件是由一个个的目标序列组成。在我们的智能体中有原始目标和最终目标，通过将动作不断的作用于外部的自然环境，智能体一步步的向最终的目标进行靠近。Agent-4如图5.5所示 Agent-4型智能体主要对我们大脑中的形象思维进行模拟。

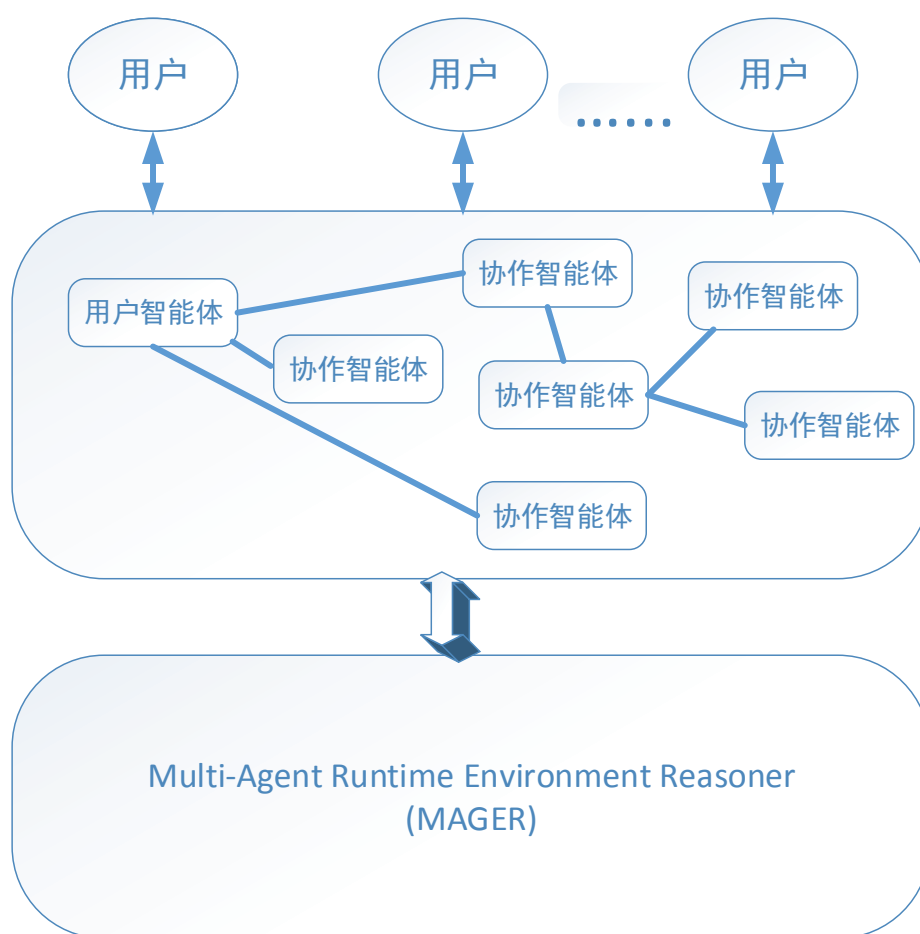


图 5.6: 多智能体信息共享

### 5.3 多智能体Web服务组合

我们上文中提到的智能体的模型，运行于MAGER平台上。MAGER是一个多主体运行环境，为方便多智能体的开发，其本身自带了一系列的开发工具。该系列工具的支持极大的方便了多智能体的开发。

如图5.6所示：多智能体信息共享系统主要由用户智能体、规划智能体组成。用户智能体和规划智能体本质上是相同的智能体。

**用户智能体**用来接收查询信息。当有查询信息请求的时候，用户智能体首先在自己的知识库中进行查询。当自己的知识库中的知识不能过对用户的查询信息进行响应的时候，这时候智能体求助去其它智能体。

**协作智能体**用来协助智能体完成推理任务。当用户查询的智能体不能够完成推理任务的时候，该协作智能体给予帮助。从全局来看，多个智能体一起来完成知识的查询和推理，可以并行的执行，加快了系统推理效率，并加大了系统的求解范围。对于

某些原来不能够进行推理的任务，现在有可能在协作智能体的共同合作下，完成推理过程。

本质上，用户智能体和协作智能体是具有平等的关系，发起方成为用户智能体，合作方成为协作智能体。

随着Web服务技术的迅速发展，Internet上Web服务无论是类型还是数量都有了爆炸式增长。Web应用从局部发展到了全球，从集中式发展到分布式，变成了一种新的分布式服务计算模型。然而单个Web服务的功能有限，如何协调和组织多个Web服务来构造新的Web服务、提高服务组件的重用性和利用率，成为Web服务领域最具有挑战性的问题之一。

在语义Web的服务的环境下，需要对静态知识和动态知识进行表示和推理。为了适应这一要求，我们将利用动态描述逻辑描述Web服务的语义信息，在此基础上实现Web服务的自动组合。基于多智能体平台的Web服务组合主要有如下特点：

- **Web服务语义化**

系统可以利用Web语义，对用户的查询请求和Web资源进行描述。为Web的组合奠定基础，提高Web服务组合的准确度。

- **智能性**

将Web服务封装成智能体服务，每个智能体具有其所在领域的丰富知识，可以运用其知识和能力进行逻辑推理，同一般的推理相比，提供的服务更加的智能。

- **推理自动化**

系统将Web服务语义映射成符合动态描述逻辑的语义信息，从而可以借助动态描述逻辑的推理进行自动化执行。

用动态描述逻辑进行推理具有如下优点：

- **具有较强的知识表示能力**

动态描述逻辑既可以对动态知识的表示，又可以对动态知识进行表示。可以对动作进行刻画，每个服务对应一个动作。将服务表示为(P,E)形式，每个原子服务对应系统中的一个动作。

- **较强的知识推理能力**

在动态描述逻辑系统中，所有领域中关于动作的推理都是可判定的。具有可判定算法对其进行支撑。

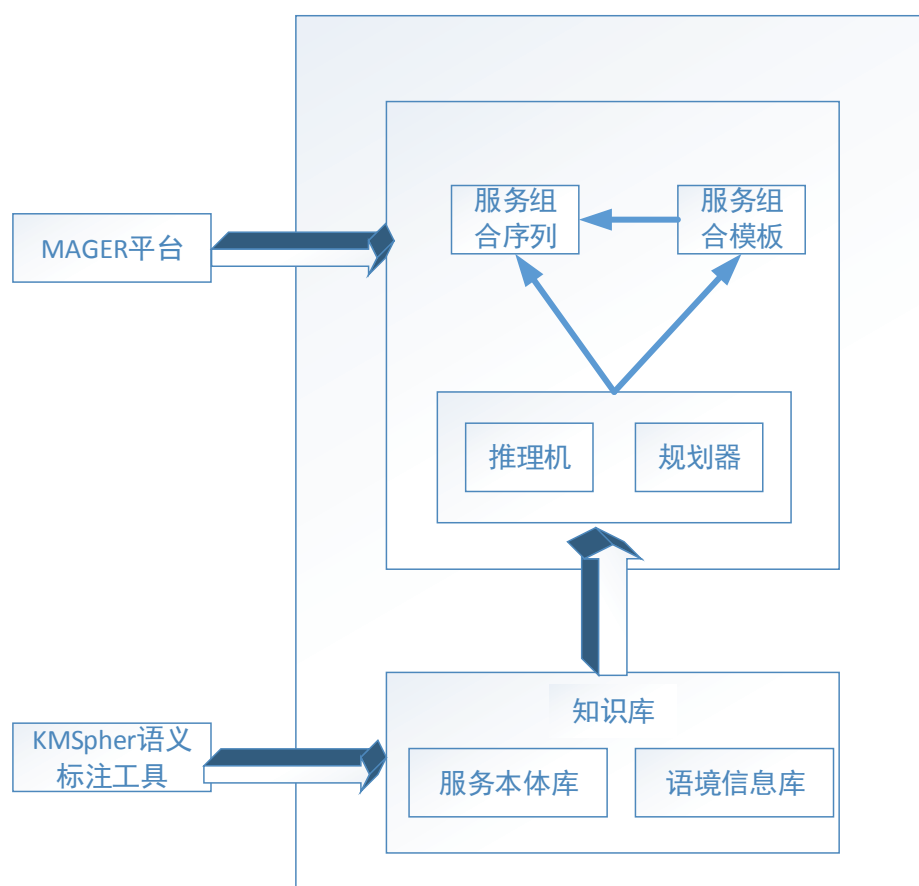


图 5.7: Web服务组合平台架构图

- 描述方式丰富

在动态描述逻辑中，我们有多种动作进行选择。主要包括原子动作、顺序动作、选择动作等。

Web服务组合平台利用动态描述逻辑的推理功能和其规划功能实现了服务层的组合。在前面章节的基础之上，我们以现有的MAGER平台和KMSphere平台作为支撑，开发设计了Web服务组合工具。该工具主要由Web知识库，Web服务组合推理机和Web服务组合模板库组成。推理平台如图5.7所示

为了进一步说明服务组合的过程，此节给出一个Web服务组合的案例。

假定一个旅行者要从北京到海南岛去旅游，希望在提供旅游服务的网站上完成整个旅行安排。客户的要求如下：提交旅游计划的时间，系统返回需要乘坐的交通工具和预订的酒店信息。用户希望输入时间和个人喜好后，系统给出乘车路线并预订酒店房间。顾客可以乘坐火车也可以乘坐飞机。

在这个案例中，用户需求涉及到3个Web服务。预订火车W1服务，预订飞机W2服

务，预订酒店W3服务。但是这些服务分布在不同的服务器中，任何单个的服务器都不能满足客户的需要。为了简单说明多个服务的推理过程，下面给出单个Web服务集合的精简描述，但是不考虑单个Web服务的细节。

预订火车服务：

$w11=\{P11,E11\};$   
 $P11=\{\text{Departure\_data}(d)\};$   
 $E11=\{\text{Airplane\_Ticket},\text{Arrive\_date}(d)\};$

预订飞机服务：

$w21=\{P21,E21\};$   
 $P21=\{\text{Departure\_data}(d)\};$   
 $E21=\{\text{Train\_Ticket},\text{Arrive\_date}(d)\};$

预订酒店服务：

$w31=\{P31,E31\};$   
 $P31=\{\text{Arrive\_date}(d)\};$   
 $E31=\{\text{hotel\_booking}\};$

根据服务的类型，系统中将相同类型的服务组合在一个集合中，同一集合中服务区别主要在于非功能属性上。即 $ws1=\{w11,w12,\dots,w1n\}$ ，其中 $n$ 为 $ws1$ 中服务的个数， $ws2=\{w21,w22,\dots,w2m\}$ ，其中 $m$ 为 $ws2$ 中服务的个数， $ws3=\{w31,w32,\dots,w3k\}$ ，其中 $k$ 为 $ws3$ 中服务的个数。将3种类型的Web服务封装成3种类型的action。

以上3种类型的服务都定义在无环的TBox中， $T=\{\text{Traffic\_tool}=\text{Airplane\_Ticket} \cup \text{Train\_Ticket}\}$ 。

假设客户具有个人偏好，对时间没有要求，而希望花费的代价最小。添加个人喜好动作， $\text{action}=\{\phi,\text{Min\_cost}\}$ 。目标 $g=\{\text{Traffic\_tool},\text{hotel\_booking}\}$ 。系统将目标 $g$ 分解为2个子目标， $g1=\text{Traffic\_tool}$ 和 $g2=\text{hotel\_booking}$ 。根据子目标执行顺序， $g1$ 子目标优先于 $g2$ 执行。目标 $g$ 分解到不能再分解的子目标之后，就可以从Web服务中选择相应。在本例中，根据客户输入的时间可以从Web服务中选择乘坐铁路还是乘坐飞机。

利用上述算法，在服务计算阶段就可以根据实际应用的语义信息定义各个服务部件需要满足的前提条件并以逻辑的形式表示。在选择出的旅游出行工具后，按着客户的个人喜好在后选的出行工具中选择价格最便宜的。出行工具确定后，然后就确定了达到的日期，从而可以进一步给客户预订放假。

经过上述执行Web服务组合后，客户就可以获知出行的交通工具以及预订的酒店

信息。

### 5.4 小结

本章主要在前述研究的基础之上，实现并设计了智能体认知模型。按照功能的不同将智能体分为多个类型，从而使其更加的适应多种环境。智能体所处的外部环境进行着动态的变化，为此，我们在智能体推理中增加了环境感知，使得智能体关注的信息不仅仅是智能体之间的交互信息，外部和内部环境的信息也非常重要，不断的生成动机和目标。并将多智能体应于Web服务组合平台。在文中，我们还着重介绍了各个模块的功能和运行流程。





## 第六章 结束语

在这一章节中，我们主要对我们全文的工作进行总结，并说明下一步的研究计划。

### 6.1 本文主要贡献

本文的研究目标是研究多智能体协同认知的基本问题，建立认知动机模型。

本文的主要贡献包括如下几个方面：

#### (1) 提出NGI动机模型

动机是一种非常复杂的现象，根据我们承担的具体课题，我们提出了一个动机模型。N表示Need(需求)，G表示Goal(目标)，I表示Intensity(强度)。智能体不断的同外界进行交互，在交互的过程中会产生各种动机。为了对动机更好的进行建模，我们将动机分为感知动机、适应性动机和合作动机。动机的生成需要目标来解决，因此，我们的动机模型中确定了目标。不同动机具有不同的重要性，为此，需要对动机的强度进行标注。

#### (2) 提出智能体ABGP协同认知

其中A表示Awareness（感知），B表示Belief（信念），G表示目标（Goal），P表示Plan（规划）。建立动机模型，并对生成的目标进行规划。智能体所处的环境进行着动态的变化，为此，我们需要对其所处的外部和内部环境进行感知，将信息及时传输到智能体。一个目标可能对应这多个动机，因此，我们需要按照动机的优先级别生成目标。目标生成后，选择优先级别最高的进行执行。对于已生成的目标，包括其初始状态和最终状态。智能体从其情景记忆中选择动作序列，进行动作的规划。

#### (3) 提出智能体信息共享协同认知

该方法通过采用信息桥接方式，打破了消息传输的二义性，方便了智能体之间信息的传输和检索。桥规则增强了知识库之间的互操作，使得信息可以按照我们预想的方式传输。在进行推理的时候，首先在智能体内部进行推理；在不能得出推理结果的时候，选择同时具有into和onto桥规的的候选智能体进行推理；如果不能得出结果，再选择三个智能体，其中一个只有into桥规则连接，一个只有onto桥规则连接，然后在选择一个智能体同时和这两个有智能体有连接，进行推理。

## 6.2 下一步研究方向

回顾整篇论文，我们对智能体的认知模型进行了建模。涉及到了认知过程中非常重要的动机模型，认识模型动作规划。

### 1. 动机模型

动机模型是认知模型的基础。动机模型是认知模型的理论基础。动机如何产生，智能体为什么来完成动机。关于动机，最核心的问题是如何进行量化，且相应的动机如何转换为目标，目标如何选择这都是我们需要继续深入研究的问题。

### 2. 认知模型

在有了动机模型的理论基础之后，我们可以构建认知模型。认知模型有多重，需要根据智能体所处的环境进行相应的设计。在认知模型中有需要需要研究的问题，其中最关键的就是如何验证我们的实验，而不是仅仅的停留在理论阶段或者演示阶段。应该考虑如何更好的应用与我们的生活各个方面。

### 3. 多智能体协作

问题复杂多变，智能体单个能力有限，故而需要多个智能体通力合作来完成。这里涉及到多智能体是分布式还是集中式管理，是层次化管理还是扁平化管理。在未来的研究中，可以对考虑如何将多智能体应用于我们客观世界。理论和实践更好的结合。

### 4. 动机规划

人类的思维非常的复杂，不同的人具有不同的思考问题的方式。面对相同的问题，往往会产生不同的解决方式。最终的结果有的殊途同归，有的南辕北辙。在未来的研究中，要继续深入的研究动机的规划问题。

## 参 考 文 献

- [1] 史忠植, 董明楷, 蒋运承, and 张海俊. 语义web 的逻辑基础. 中国科学: *E* 辑, 34(10):1123–1138, 2005.
- [2] 邱莉榕, 史忠植, 林芬, and 常亮. 基于主体的语义web 服务自动组合研究. 计算机研究与发展, 44(4):643–650, 2007.
- [3] 常亮, 史忠植, 邱莉榕, and 林芬. 动态描述逻辑的tableau 判定算法. 计算机学报, 31(6):896–909, 2008.
- [4] 毛新军, 王怀民, and 吴刚. 联合意图的理论框架. 计算机科学, 30(007):139–143, 2003.
- [5] 雷仲魁, 仲筱艳, and 钱默抒. 无人驾驶飞机飞行控制新技术. 南京航空航天大学学报, 41(B12):12–14, 2009.
- [6] 郝永军 and 周德云. 多无人机协同攻击路径规划研究. 计算机仿真, 27(3):69–72, 2010.
- [7] 史忠植. 智能主体及其应用. 北京: 科学出版社, 2000.
- [8] 史忠植. 智能科学 (第二版). 北京: 清华大学出版社, 2013.
- [9] H Levent Akin, Nobuhiro Ito, Adam Jacoff, Alexander Kleiner, Johannes Pellenz, and Arnoud Visser. Robocup rescue robot and simulation leagues. *AI Magazine*, 34(1), 2013.
- [10] Rama Akkiraju, Joel Farrell, John A Miller, Meenakshi Nagarajan, Amit Sheth, and Kunal Verma. Web service semantics-wsdl-s. 2005.
- [11] Clayton P Alderfer. Existence, relatedness, and growth: Human needs in organizational settings. 1972.
- [12] Lidia Alonso-Nanclares and Javier DeFelipe. Alterations of the microvascular network in the sclerotic hippocampus of patients with temporal lobe epilepsy. *Epilepsy & Behavior*, 2014.

- [13] John R Anderson. Act: A simple theory of complex cognition. *American Psychologist*, 51(4):355, 1996.
- [14] John R Anderson, Daniel Bothell, Michael D Byrne, Scott Douglass, Christian Lebiere, and Yulin Qin. An integrated theory of the mind. *Psychological review*, 111(4):1036, 2004.
- [15] John Robert Anderson and Christian Lebiere. *The atomic components of thought*. Psychology Press, 1998.
- [16] Daniel Austin, Abbie Barbir, Christopher Ferris, and Sharad Garg. Web services architecture requirements. *W3C working draft*, 19, 2002.
- [17] Franz Baader. *The description logic handbook: theory, implementation, and applications*. Cambridge university press, 2003.
- [18] Joscha Bach. A motivational system for cognitive ai. In *Artificial General Intelligence*, pages 232–242. Springer, 2011.
- [19] Steve Battle, Abraham Bernstein, Harold Boley, Benjamin Grosz, Michael Gruninger, Richard Hull, Michael Kifer, David Martin, Sheila McIlraith, Deborah McGuinness, et al. Semantic web services ontology (swso). *Member submission, W3C*, 2005.
- [20] Birol Berkem. From the business motivation model (bmm) to service oriented architecture (soa). *Journal of Object Technology*, 7(8):57–70, 2008.
- [21] Alain Berthoz. *The brain's sense of movement*. Harvard University Press, 2002.
- [22] Dan Bohus and Alexander I Rudnicky. The ravenclaw dialog management framework: Architecture and systems. *Computer Speech & Language*, 23(3):332–361, 2009.
- [23] Alex Borgida and Luciano Serafini. Distributed description logics: Directed domain correspondences in federated information sources. In *On the Move to Meaningful Internet Systems 2002: CoopIS, DOA, and ODBASE*, pages 36–53. Springer, 2002.
- [24] Alex Borgida and Luciano Serafini. Distributed description logics: Assimilating information from peer sources. In *Journal on Data Semantics I*, pages 153–184. Springer, 2003.

- [25] Ronald J Brachman and James G Schmolze. An overview of the kl-one knowledge representation system\*. *Cognitive science*, 9(2):171–216, 1985.
- [26] Donald E Broadbent. Modules in models of memory. *Behavioral and Brain Sciences*, 7(01):86–94, 1984.
- [27] Liang Chang, Fen Lin, and Zhongzhi Shi. A dynamic description logic for representation and reasoning about actions. In *Knowledge Science, Engineering and Management*, pages 115–127. Springer, 2007.
- [28] Philip R Cohen and Hector J Levesque. Intention is choice with commitment. *Artificial intelligence*, 42(2):213–261, 1990.
- [29] Jos De Bruijn, Holger Lausen, Axel Polleres, and Dieter Fensel. The web service modeling language wsml: An overview. In *The Semantic Web: Research and Applications*, pages 590–604. Springer, 2006.
- [30] Etienne de Sevin and Daniel Thalmann. A motivational model of action selection for virtual humans. In *Computer Graphics International 2005*, pages 213–220. IEEE, 2005.
- [31] Wlodzislaw Duch, Richard Jayadi Oentaryo, and Michel Pasquier. Cognitive architectures: Where do we go from here? In *AGI*, volume 171, pages 122–136, 2008.
- [32] Chris Eliasmith and Oliver Trujillo. The use and abuse of large-scale brain models. *Current Opinion in Neurobiology*, 25:1–6, 2014.
- [33] Mica R Endsley. Toward a theory of situation awareness in dynamic systems. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 37(1):32–64, 1995.
- [34] H Farrokhi, I GhodratiToostani, M Farasatkah, and H Ekhtiari. Editorial: Sustainable development of cognitive science and technology ecosystem; an overview to the “human brain project” as a functioning sample. *Basic and Clinical Neuroscience*, 5(1):4–10, 2014.
- [35] George Ferguson and James Allen. Mixed-initiative systems for collaborative problem solving. *AI magazine*, 28(2):23, 2007.

- [36] George Ferguson and James Allen. A cognitive model for collaborative agents. In *Proceedings of the AAAI 2011 Fall Symposium on Advances in Cognitive Systems*, pages 112–120, 2011.
- [37] Robert James Firby. *Adaptive execution in complex dynamic worlds*. PhD thesis, Citeseer, 1990.
- [38] Stan Franklin and Art Graesser. Is it an agent, or just a program?: A taxonomy for autonomous agents. In *Intelligent agents III agent theories, architectures, and languages*, pages 21–35. Springer, 1997.
- [39] Marc-Oliver Gewaltig and Robert Cannon. Current practice in software development for computational neuroscience and how to improve it. *PLoS computational biology*, 10(1):e1003376, 2014.
- [40] GH Granlund. Cognitive vision—background and research issues. *Unpublished manuscript, Linköping*, 2002.
- [41] Gösta H Granlund. The complexity of vision. *Signal Processing*, 74(1):101–126, 1999.
- [42] Gösta H Granlund. Does vision inevitably have to be active? In *Proceedings of the Scandinavian Conference on Image Analysis*, volume 1, pages 11–20, 1999.
- [43] Barbara Gray. Collaborating: Finding common ground for multiparty problems. 1989.
- [44] Stephen Grossberg. Adaptive pattern classification and universal recoding: I. parallel development and coding of neural feature detectors. *Biological cybernetics*, 23(3):121–134, 1976.
- [45] Barbara Grosz and Sarit Kraus. Collaborative plans for group activities. In *IJCAI*, volume 93, pages 367–373, 1993.
- [46] Barbara J Grosz and Sarit Kraus. Collaborative plans for complex group action. *Artificial Intelligence*, 86(2):269–357, 1996.
- [47] Alejandro Guerra-Hernández, Amal El Fallah-Seghrouchni, and Henry Soldano. Learning in bdi multi-agent systems. In *Computational logic in multi-agent systems*, pages 218–233. Springer, 2005.

- [48] Carl Gutwin and Saul Greenberg. A descriptive framework of workspace awareness for real-time groupware. *Computer Supported Cooperative Work (CSCW)*, 11(3-4):411–446, 2002.
- [49] Carl Gutwin and Saul Greenberg. The importance of awareness for team cognition in distributed collaboration. *Team cognition: Understanding the factors that drive process and performance*, 201:1–33, 2004.
- [50] John Haugeland. Semantic engines: An introduction to mind design. *Mind design*, pages 1–34, 1981.
- [51] John J Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the national academy of sciences*, 79(8):2554–2558, 1982.
- [52] Leslie Pack Kaelbling, Michael L Littman, and Andrew W Moore. Reinforcement learning: A survey. *arXiv preprint cs/9605103*, 1996.
- [53] John M Keller and Katsuaki Suzuki. Use of the arcs motivation model in courseware design. 1988.
- [54] JA Scott Kelso. *Dynamic patterns: The self-organization of brain and behavior*. MIT press, 1995.
- [55] Jeffrey O Kephart and William E Walsh. An artificial intelligence perspective on autonomic computing policies. In *Policies for Distributed Systems and Networks, 2004. POLICY 2004. Proceedings. Fifth IEEE International Workshop on*, pages 3–12. IEEE, 2004.
- [56] John F Kihlstrom. The cognitive unconscious. *Science*, 237(4821):1445–1452, 1987.
- [57] Benjamin Lacar, Sarah L Parylak, Krishna C Vadodaria, Anindita Sarkar, and Fred H Gage. Increasing the resolution of the adult neurogenesis picture. *F1000prime reports*, 6, 2014.
- [58] John Laird. *The Soar cognitive architecture*. MIT Press, 2012.
- [59] John E Laird. Extending the soar cognitive architecture. *Frontiers in Artificial Intelligence and Applications*, 171:224, 2008.

- [60] John E Laird and Paul Rosenbloom. The evolution of the soar cognitive architecture. *Mind matters: A tribute to Allen Newell*, pages 1–50, 1996.
- [61] John E Laird, Paul S Rosenbloom, and Allen Newell. Chunking in soar: The anatomy of a general learning mechanism. *Machine learning*, 1(1):11–46, 1986.
- [62] Nada Lavrac and Saso Dzeroski. Inductive logic programming. In *WLP*, pages 146–160. Springer, 1994.
- [63] Christian Lebiere and John R Anderson. A connectionist implementation of the act-r production system. 2008.
- [64] Chang Liang, Shi Zhongzhi, Qiu Lirong, and L Fen. A tableau decision algorithm for dynamic description logic. *Chinese Journal of Computers*, 31(6):896–909, 2008.
- [65] George F Luger, Peder Johnson, Carl Stern, JeanE Newman, and Ronald Yeo. *Cognitive science: the science of intelligent systems*. Academic Press San Diego, CA, 1994.
- [66] David JC MacKay. *Information theory, inference and learning algorithms*. Cambridge university press, 2003.
- [67] Henry Markram. The blue brain project. *Nature Reviews Neuroscience*, 7(2):153–160, 2006.
- [68] David Marr. Artificial intelligence—a personal view. *Artificial Intelligence*, 9(1):37–48, 1977.
- [69] David Marr and A Vision. A computational investigation into the human representation and processing of visual information. *WH San Francisco: Freeman and Company*, 1982.
- [70] David Martin, Mark Burstein, Drew Mcdermott, Sheila Mcilraith, Massimo Paolucci, Katia Sycara, Deborah L McGuinness, Evren Sirin, and Naveen Srinivasan. Bringing semantics to web services with owl-s. *World Wide Web*, 10(3):243–277, 2007.
- [71] Abraham Harold Maslow. A theory of human motivation. *Psychological review*, 50(4):370, 1943.



- [72] Humberto R Maturana and Francisco J Varela. *The tree of knowledge: The biological roots of human understanding*. New Science Library/Shambhala Publications, 1987.
- [73] James L McClelland, David E Rumelhart, PDP Research Group, et al. Parallel distributed processing. *Explorations in the microstructure of cognition*, 2, 1986.
- [74] Marvin Minsky. A framework for representing knowledge. 1974.
- [75] Tom M Mitchell, Richard M Keller, and Smadar T Kedar-Cabelli. Explanation-based generalization: A unifying view. *Machine learning*, 1(1):47–80, 1986.
- [76] Michael C Móra, José G Lopes, Rosa M Viccariz, and Helder Coelho. Bdi models and systems: Reducing the gap. In *Intelligent Agents V: Agents Theories, Architectures, and Languages*, pages 11–27. Springer, 1999.
- [77] Manuel Muñoz, Pau Muñoz, Eduardo Munera, J Francisco Blanes, and José Simó. Limited resources management in a robocup team vision system. In *ROBOT2013: First Iberian Robotics Conference*, pages 27–39. Springer, 2014.
- [78] Allen Newell. The knowledge level. *Artificial intelligence*, 18(1):87–127, 1982.
- [79] Allen Newell. *Unified theories of cognition*. Harvard University Press, 1994.
- [80] Allen Newell and Herbert A Simon. Computer science as empirical inquiry: Symbols and search. *Communications of the ACM*, 19(3):113–126, 1976.
- [81] Gualtiero Piccinini and Andrea Scarantino. Information processing, computation, and cognition. *Journal of biological physics*, 37(1):1–38, 2011.
- [82] Steven Pinker. Visual cognition: An introduction. *Cognition*, 18(1):1–63, 1984.
- [83] Alexander Pokahr, Lars Braubach, and Winfried Lamersdorf. A goal deliberation strategy for bdi agent systems. In *Multiagent System Technologies*, pages 82–93. Springer, 2005.
- [84] Zenon W Pylyshyn. *Computation and cognition*. Cambridge Univ Press, 1984.
- [85] Guilin Qi and Andreas Harth. Reasoning with networked ontologies. In *Ontology Engineering in a Networked World*, pages 363–380. Springer, 2012.

- [86] Anand S Rao and Michael P Georgeff. Modeling rational agents within a bdi-architecture. *KR*, 91:473–484, 1991.
- [87] Charles Rich and Candace L Sidner. Collagen: When agents collaborate with people. In *Proceedings of the first international conference on Autonomous agents*, pages 284–291. ACM, 1997.
- [88] Giacomo Rizzolatti, Luciano Fadiga, Leonardo Fogassi, and Vittorio Gallese. The space around us. *Science*, 277(5323):190–191, 1997.
- [89] Giacomo Rizzolatti, Leonardo Fogassi, and Vittorio Gallese. Parietal cortex: from sight to action. *Current opinion in neurobiology*, 7(4):562–567, 1997.
- [90] Steven Rose. On megabrain projects. *EMBO reports*, 2014.
- [91] P Justin Rossi, Michael Okun, and James Giordano. Translational imperatives in deep brain stimulation research: Addressing neuroethical issues of consequences and continuity of clinical care. *AJOB Neuroscience*, 5(1):46–48, 2014.
- [92] Stuart Jonathan Russell, Peter Norvig, John F Canny, Jitendra M Malik, and Douglas D Edwards. *Artificial intelligence: a modern approach*, volume 2. Prentice hall Englewood Cliffs, 1995.
- [93] Giulio Sandini, Giorgio Metta, and David Vernon. Robotcub: An open framework for research in embodied cognition. In *Humanoid Robots, 2004 4th IEEE/RAS International Conference on*, volume 1, pages 13–32. IEEE, 2004.
- [94] Manfred Schmidt-Schauß and Gert Smolka. Attributive concept descriptions with complements. *Artificial intelligence*, 48(1):1–26, 1991.
- [95] Luciano Serafini and Andrei Tamilin. Local tableaux for reasoning in distributed description logics. In *Proc. of the International Workshop on Description Logics, DL*, volume 4, pages 100–109, 2004.
- [96] Roger N Shepard and Shelley Hurwitz. Upward direction, mental rotation, and discrimination of left and right turns in maps. *Cognition*, 18(1):161–193, 1984.
- [97] Zhongzhi Shi. Research on brain-like computer. In *Brain Informatics*, pages 5–5. Springer, 2009.

- [98] Zhongzhi Shi, Mingkai Dong, Yuncheng Jiang, and Haijun Zhang. A logical foundation for the semantic web. *Science in China Series F: Information Sciences*, 48(2):161–178, 2005.
- [99] Zhongzhi Shi, Xiaofeng Wang, Zhiping Shi, Limin Chen, and Zhuxiao Wang. A mind model for brain-like computer. In *Cognitive Informatics (ICCI), 2010 9th IEEE International Conference on*, pages 257–264. IEEE, 2010.
- [100] Zhongzhi Shi, Jianhua Zhang, Jinpeng Yue, and Xi Yang. A cognitive model for multi-agent collaboration. *International Journal of Intelligence Science*, 4:1, 2013.
- [101] H Simon. Cognitive science: Relationship of ai to psychology and neuroscience. AAAI, 2010.
- [102] Norman K Sondheimer, Ralph M Weischedel, and Robert J Bobrow. Semantic interpretation using kl-one. In *Proceedings of the 10th International Conference on Computational Linguistics and 22nd annual meeting on Association for Computational Linguistics*, pages 101–107. Association for Computational Linguistics, 1984.
- [103] John F Sowa. *Conceptual structures: information processing in mind and machine*. Addison-Wesley Longman Publishing Co., Inc., 1984.
- [104] Ron Sun. Motivational representations within a computational cognitive architecture. *Cognitive Computation*, 1(1):91–103, 2009.
- [105] Milind Tambe. Towards flexible teamwork. *arXiv preprint cs/9709101*, 1997.
- [106] Paul Thagard. *Mind: Introduction to cognitive science*. MIT press, 2005.
- [107] Esther Thelen. *A dynamic systems approach to the development of cognition and action*. MIT press, 1996.
- [108] Pablo Toharia, Juan Morales, Octavio de Juan, Isabel Fernaud, Angel Rodríguez, and Javier DeFelipe. Musical representation of dendritic spine distribution: A new exploratory tool. *Neuroinformatics*, pages 1–13, 2014.
- [109] Toby Tyrrell. *Computational mechanisms for action selection*. PhD thesis, University of Edinburgh Edinburgh, Scotland, 1993.

- [110] Manuela M Veloso and Jaime G Carbonell. Integrating analogy into a general problem-solving architecture. *Intelligent Systems*, pages 29–51, 1990.
- [111] David Vernon, Giorgio Metta, and Giulio Sandini. A survey of artificial cognitive systems: Implications for the autonomous development of mental capabilities in computational agents. *Evolutionary Computation, IEEE Transactions on*, 11(2):151–180, 2007.
- [112] Y Wang. Special issue on cognitive computing, on abstract intelligence. *International Journal of Software Science and Computational Intelligence*, 1(3), 2009.
- [113] Yingxu Wang. On cognitive informatics. In *Cognitive Informatics, 2002. Proceedings. First IEEE International Conference on*, pages 34–42. IEEE, 2002.
- [114] Yingxu Wang. The real-time process algebra (rtpa). *Annals of Software Engineering*, 14(1-4):235–274, 2002.
- [115] Yingxu Wang. On cognitive informatics. *Brain and Mind*, 4(2):151–167, 2003.
- [116] Yingxu Wang. *Software engineering foundations: A software science perspective*. CRC Press, 2007.
- [117] Yingxu Wang. The theoretical framework of cognitive informatics. *International Journal of Cognitive Informatics and Natural Intelligence (IJCINI)*, 1(1):1–27, 2007.
- [118] Yingxu Wang. The theoretical framework of cognitive informatics. *Simulation*, 8:A5, 2009.
- [119] Yingxu Wang and W Kinsner. Editorial recent advances in cognitive informatics. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 36(2):121–123, 2006.
- [120] Yingxu Wang, Witold Kinsner, James A Anderson, Du Zhang, Yiyu Yao, Philip Sheu, Jeffrey Tsai, Witold Pedrycz, Jean-Claude Latombe, Lotfi A Zadeh, et al. A doctrine of cognitive informatics (ci). *Fundamenta Informaticae*, 90(3):203–228, 2009.

- [121] Yingxu Wang and Ying Wang. Cognitive informatics models of the brain. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 36(2):203–207, 2006.
- [122] Zhuxiao Wang, Zhitao Guan, Wei Li, Kehe Wu, Jing Guo, and Guanhua Tian. A tableau-based reasoning algorithm for distributed dynamic description logics. In *Information Computing and Applications*, pages 192–199. Springer, 2012.
- [123] Zhuxiao Wang, Jing Guo, Fei Chen, Kehe Wu, and Peng Wang. An extension of distributed dynamic description logics for the representation of heterogeneous mappings. *Journal of Software (1796217X)*, 8(1), 2013.
- [124] Zhuxiao Wang, Hui Peng, Jing Guo, Ying Zhang, Kehe Wu, Huan Xu, and Xiaofeng Wang. An architecture description language based on dynamic description logics. In *Intelligent Information Processing VI*, pages 157–166. Springer, 2012.
- [125] Terry Winograd and Fernando Flores. *Understanding computers and cognition: A new foundation for design*. Intellect Books, 1986.
- [126] DONG Ying-dong. Multi-agent autoepistemic logic. *Journal of Southwest University (Social Sciences Edition)*, 5:017, 2009.
- [127] Richard M Young and Richard L Lewis. The soar cognitive architecture and human working memory. *Models of working memory: Mechanisms of active maintenance and executive control*, pages 224–256, 1999.
- [128] Xiaofei Zhao, Dongping Tian, Limin Chen, and Zhongzhi Shi. Reasoning theory for d3l with compositional bridge rules. In *Intelligent Information Processing VI*, pages 106–115. Springer, 2012.



## 致 谢

时光如梭，五年的求学时光不知不觉间即将走过。值此博士论文完成之际，谨向所有给予我关心和帮助的老师、同学、朋友和家人表示真挚的感谢。

首先衷心地感谢我的导师安波副研究员，感谢他对我无微不至的关怀，以及孜孜不倦的教诲。安老师渊博的知识、严谨的治学态度，让我受益匪浅。尤其是忘我的工作热情，持之以恒的工作态度，旺盛的工作精力，深深地打动了我，感染了我，成为我学习的榜样，是我以后工作中的楷模。

感谢史忠植研究员。他对科学一丝不苟的态度深深的影响这我，对认知科学执着的精神意志感染着我。是他第一次将智能体理论引入到国内，为国内的研究人员找到一个全新的研究领域。史老师每天忘我的工作状态，早起晚睡，为我的学习带来了无限的动力。与史老师进行学术讨论，是我很受启发。

感谢胡宏副研究员在工作和生活上的无私指导和热心帮助。感谢杨熙博士后在学业上给予的无私帮助。特别感谢王晓峰博士对我的指导，从研究方法、论文写作到实验验证等各个方面他都给予我很多有益的帮助。感谢胡兰萍女士和田卫平女士，她们的工作使我感到实验室的温馨。感谢研究生部李琳老师、宋守礼老师、周世佳老师、冯钢老师、李丹老师和郭晓康老师等在生活中给我的关心和帮助。

感谢在智能科学课题组一起学习和工作过的同学们，大家的和谐相处使我在生活中保持愉快的心情，也感谢他们在工作和生活上的帮助与支持。他们是张博、岳金朋、姜广，田东平、马刚、张文波、庞亮、欧阳潘奕。

感谢所有帮助过和关心过我的人！





## 作者简历

### 基本情况

姓名：张建华      性别：男      出生日期：1982年12月      籍贯：河北保定

### 作者简历

2009年9月—2014年7月，中科院计算所计算机应用技术专业博士研究生

2006年9月—2009年7月，北京科技大学计算机应用技术专业硕士研究生

2002年9月—2006年7月，北京科技大学计算机科学与技术专业本科生

### 【攻读博士学位期间发表的论文】

- [1] 张建华，史忠植，岳金朋，齐保元，蒋运承，王晓峰. 支持链式桥规则的分布式动态描述逻辑. 高技术通讯（已接受）.
- [2] Jianhua Zhang, Wenjie Wang, Zhongzhi Shi, Dongping Tian, Jinpeng Yue, Bo Zhang. Dynamic Description Logic based Semantic Web Service Composition. Proceedings of the 9th International Conference on Semantics, Knowledge and Grids, 2013, pp.211-214.(SKG2013)
- [3] Jianhua Zhang, Wenjie Wang, Zhongzhi Shi, Jinpeng Yue, Baoyuan Qi, Dongping Tian. Bridge Rule Based D3L Reasoning. Proceedings of the 9th International Conference on Semantics, Knowledge and Grids, 2013, pp.190-193.(SKG2013)
- [4] Jianhua Zhang, Zhongzhi Shi, Jinpeng Yue, Baoyuan Qi. Joint intention based collaboration. Proceedings of the Workshop on Intelligence Science, twenty-third International Joint Conference on Artificial Intelligence, 2013, pp.58-63.(WIS2013)
- [5] Zhongzhi Shi, Jianhua Zhang, Jinpeng Yue. Mind Modeling In Intelligence Science. Proceedings of the Workshop on Intelligence Science, twenty-third International Joint Conference on Artificial Intelligence, 2013, pp.31-36.(WIS2013)

- [ 6 ] Zhongzhi Shi, **Jianhu Zhang**, Jinpeng Yue, Baoyuan Qi. A Motivational System for Mind Model CAM. AAAI Symposium 2013, pp.79-86.
- [ 7 ] Zhongzhi Shi, Jianhua Zhang, Jinpeng Yue, and Xi Yang. A Cognitive Model for Multi-Agent Collaboration. International Journal of Intelligence Science, 2013, (4), pp.1-6.
- [ 8 ] 张博, 史忠植, 赵晓非, **张建华**. 一种基于跨领域典型相关性分析的迁移学习方法. 计算机学报, (已接受) .

### 【攻读博士学位期间参加的科研项目】

- [ 1 ] 国家973项目子课题“脑机协同的认知计算模型” No.2013CB329502
- [ 2 ] 国家自然科学基金重点项目“基于云计算的海量数据挖掘” No.61035003
- [ 3 ] 国家自然科学基金重点项目“用于复杂多智能体系统资源分配的协商理论的研究” No.61202212
- [ 4 ] 国家自然科学基金重点项目“神经电路的模糊逻辑框架及其在仿大脑皮层功能的认知计算机设计中的应用” No.61072085