

多 Agent 系统中的本体论

刘志忠¹ 王怀民¹ 丛广年²

(国防科技大学计算机学院 长沙410073)¹ (国防科技大学航天与材料工程学院 长沙410073)²

摘要 多 Agent 系统之间的本体异构严重妨碍了系统之间的知识共享与集成以及 Agent 之间的协作。文章首先介绍了解决本体异构的多种方法,并分析了各种方法所存在的不足。为了克服这些方法的不足,文章给出了一种结合元本体理论和本体协商的方法。该方法将元本体引入本体协商过程,从而提高了本体协商的效率并且为协商最终达成一致提供了客观的标准。文章最后通过一个简单的实例说明了基于元本体进行本体协商的大致过程。

关键词 多 Agent 系统,本体论,本体异构,元本体,本体协商

Ontologies in Multi-Agent Systems

LIU Zhi-Zhong¹ WANG Huai-Min¹ CONG Guang-Nian²

(Institute of Computer, National University of Defense Technology, HuNan Changsha 410073)¹

(Institute of Aerospace and Material Engineering, National University of Defense Technology, Changsha 410073)²

Abstract Ontological heterogeneity blocks the sharing and integration of knowledge and cooperation between agents among multi-agent systems greatly. The paper describes many methods to solve this problem and analyzes the flaws of these methods firstly. To overcome these shortcomings, the paper presents a novel method that combines the theory of meta-ontology and ontology negotiation. Due to introduce meta-ontology into the process of ontology negotiation, the method improves the efficiency of ontology negotiation as well as provides an external standard that imply agents reach consensus. At last, it illustrates the process of ontology negotiation based on meta-ontology using a simple example.

Keywords Multi-agent system, Ontological heterogeneity, Meta-ontology, Ontology negotiation

1 引言

本体论是一个哲学概念,区别于认知论,是研究世界的本源与基质的一个哲学分支。该概念引入到人工智能领域也有很多年了,但是到目前为止,关于人工智能中本体的概念尚未形成完全统一的认识。但是大致都认为:本体是针对某一特定领域的共享的形式化的概念规格说明^[1]。

多 Agent 理论以及多 Agent 系统的通信语言 KQML 的提出已经很多年,但多 Agent 系统目前还没有得到广泛的应用,其中一个主要问题就是多 Agent 系统中的本体论没有得到彻底的解决^[2]。多 Agent 系统本体理论的研究主要包括两个方面。第一方面,本体理论需要为多 Agent 系统中的各个 Agent 提供了一个对共享知识统一认识。这是本体引入的根本原因。这方面的研究主要是着重于本体建造、本体表示、本体进化以及本体内部的一致性维护等方面;另一方面,随着多 Agent 理论不断发展,Agent 之间的协作已经不仅仅是在多 Agent 系统内部进行。因此本体理论还需要解决多 Agent 系统之间的本体异构问题,为多 Agent 系统之间的协作提供必要的技术保障。这方面的研究主要就是关于各种不同的解决本体异构的方法。文章的第2节我们将详细介绍各种不同的解决方法,并做一定的分析比较。

当然,本体论的研究不仅仅局限于多 Agent 领域。在分布式计算领域,无论是 CORBA、COM/DCOM 还是 J2EE,为了让数据能够“自由”地流动,各个计算实体都应该对共享的

知识具有统一认识,也就是说需要建立本体库来保证各实体对概念认识的一致性。另外,在语义 Web 或者是智能 Web 中,为了让机器能够直接从 Web 中抽取知识并理解其语义,也需要为其共享的知识建立标准的框架,也就是本体。此外在机器翻译、知识工程以及数据挖掘等领域的研究也需要本体的支持,但是这些领域对本体的研究基本上都是基于本体论本身而对于多个本体之间的集成却研究不多。

2 解决多 Agent 系统中本体异构的方法

由于多 Agent 系统的开放性,系统实际运行过程中,Agent 可能需要和一些陌生的 Agent 协作,这些 Agent 和该 Agent 由于针对的应用领域以及建立本体的人员不同,在本体建造可能会存在不一致甚至矛盾。因此导致 Agent 之间协作困难,为了保证 Agent 之间进行有效协作,我们必须解决这种不一致。这些不一致有两个不同的层次。首先可能是在语言表达的层次。知识的表示具有各种各样不同的表示方法:基于一阶逻辑的表示、基于框架的表示等等。由于知识表示方法不同,多 Agent 系统的本体之间可能存在异构。不过这个问题比较容易解决,我们可以设计一种通用的具有丰富表达能力的知识表示方法来表示本体知识,现在一般都采用 KIF^[3]。另外一个层次上的异构是指本体上的异构。这种异构也包括两种,首先是由于不同领域导致的对同一个概念的不同理解。比如说在社会学中研究的人和生理学中研究的人的定义肯定不会完全一样。如果一个社会学家和一个生理学家需要进行

协作,如何保证他们对人的概念的认识完全一致?这是本体论需要解决的问题,还有就是同一个领域中不同的概念表示的是同一个概念,比如英语中的 Apple 和中文中的苹果其实是同一件事物,但是如果建立的本体不一样我们可能会认为这是两个不同的概念,这个层次上的本体异构比较复杂,也比较难于处理。针对这一层次上的异构,已经有不少学者提出了各种各样的解决方法。下面将简单介绍各种解决方案。

2.1 标准本体

我们很自然能想到的办法就是建立一个为某个应用领域建立一个标准的本体。这种方法非常直接,但是要建立一个本体是非常困难的。首先,即使是同一个应用领域,由于具体的应用背景不同,对一些概念的认识也不尽相同,要建立标准的本体可能性不大。其次,即使是建立相对较为完善的本体,由于需要考虑各种可能的应用,建立本体将耗费大量的时间和费用,而结果却未必能达到预期的效果。尽管如此,还是有一些学者正在研究建立一个标准的本体库,如:IEEE 的标准上层本体(Standard Upper Ontology)的目标就是希望通过建立一个通用目标的形式化的本体来实现异质信息系统之间的推理以及语义互操作等。为了达到这个目标 IEEE 分别采用了两种不同的途径。一种是企图在当前本体的基础之上融合这些本体以得到一标准本体。另外一条途径企图通过根据分类学的原理建立一个很小的元本体(Meta-Ontology)之后根据一定的原则对元本体进行扩充而得到一个相对比较完善的标准本体^[4]。

2.2 本体集合

由于建立标准本体的想法并不能容易实现,因此有人就想建立一个更大的本体来包含已有的各个本体。这种方法实现起来非常简单,只是将所有本体组织到一起就可以了,但是它有着非常明显的不足:由于应用领域的不同各个本体之间可能存在不一致,该方法并不能处理这些不一致。因此这种方法只适用于本体之间不一致比较小的情况,或者是该不一致对整个系统影响不大的情况,如 StepLib 系统以及 RiboWeb 系统就采用了这种方法。

2.3 本体集成

为了解决上面方法中的明显不足,有人就提出了本体集成的办法。该方法采用系统集成的方法来解决本体之间存在的的不同。但是这同时导致一个问题:为了达到集成的目的,各多 Agent 系统势必要对自己的本体做一定的映射,也就是说需要对本体本身进行修改,从而可能会影响系统内部的协作进行。为了解决这个问题,有些学者提出了在系统集成过程当中保留概念使用时的上下文,从而减少不正常的产生。该方法在 SMART 系统中得到了使用。此外在医疗诊断领域也有一定的应用,如 StepLib。

2.4 本体仲裁

该思想是由 Wiederhold 等人提出来的。本体仲裁是指在多个本体之上设立一个仲裁,在 Agent 访问本体库时,通过仲裁来决定应该访问哪个本体库。在这种方法当中,原来的各个本体都保持了自身的特性,而本体之间的不一致也不会影响到 Agent 之间的协作。该方法的实现也比较复杂:如何确定每次访问本体库应该访问哪个本体库?以及仲裁器本身需要向外界提供什么样的接口。所有这些都和具体的应用相关,因此实现起来比较复杂。当然,该方法本身也具有一定的灵活性,仲裁器可以灵活地配置到系统的任何地方。该方法在 COIN 系统以及 OBSERVER 系统中得到了应用。

2.5 开放本体与分布式开放本体

该方法类似于数据库访问中的 ODBC 访问,即本体库对外提供了一套标准的接口来允许其他 Agent 来访问自身的本体库。这样的话可以解决前面方法中仲裁器实现所带来的麻烦。而且 Agent 可以根据自身的需要来决定概念访问所需要的精确程度,这种方法是 Visser 等人提出来的,并且提供了一套类似于 ODBC 的 OKBC (Open Knowledge Base Connection) 接口来实现这一想法。此外,随着语义 Web 的逐渐广泛使用,集中式的 OKBC 访问已经不适用于当前的网络环境。因此 W3C 组织基于当前的 XML 的发展,提出了用 RDF 来对本体进行编码,并且提出了基于 XML 的本体交换语言 OXL 和本体推理层 OIL。在这个标准之下,美国的 DARPA 提出了将主体标识语言 DAML 和 OIL 相结合的本体访问方法:DAML+OIL。该方法实际上是为本体库的分布式访问提供了一个标准。

2.6 本体交换

除了上面的开放式访问方法以外还有另外一种访问方法,即直接通过消息交换来通信本体信息。这样使得 Agent 访问其他 Agent 的本体库时不需要和其他的主体的位置保持一致。该方法并不要求所有主体采用一样的语言,而只要能包含一些自解释的消息。这样 Agent 就可以将有关本体的交互嵌入到消息内部来进行处理。在这当中可能就要存在某种折衷:本体的规模以及自包含消息的数目。一般说来,本体的规模越大那么所需要的自包含的消息数目也就越少。

2.7 本体协商

本体协商是基于本体交换的一种解决异构的方法^[5]。该方法不仅仅在将关于本体的交换嵌入在 Agent 之间进行通信的消息中,并且在本体交换完成之后,主体之间还按照一定的协议来进行协商,对本体进行解释,说明以及确认。该方法可以使用较少的几个消息通过不断地进行循环和迭代对本体不断地进行理解与确认,直到达到双方都满意的结果。这种方法存在一些问题:由于本体事先都是完全独立定义的,如何能够对解释、说明以及确认的消息原语的理解上达成一致?为了达到这个目标,各 Agent 系统在建立各自的本体库时必须遵守某些一般的法则,这使得各本体库之间还存在一定的联系。该方法可以通过 Agent 之间的本体协商时来完成本体库自身的进化,使得经过不断的使用,各 Agent 的本体库最终可能会收敛到一致的状态。该方法的一个缺点是 Agent 之间进行本体协商需要花大量的时间来进行概念求精。因此,在系统建立之初整个系统进行协作的效率比较低下。

2.8 其他

还有些人认为,Agent 之间协作困难并不仅仅是由于本体异构所造成的,其另一重要原因是当前所采用的本体形式化工具的表达力不够强,并不能完全表示本体所应该具有的属性。因此,这些学者就试图通过对当前本体形式化工具予以扩充,增强其表达能力,使得 Agent 之间进行协作更为方便,在一定程度上可以消除本体异构所带来的麻烦。如:Valentina 等提出了扩展本体模型^[6]。该模型在传统的本体模型中引入了可能世界理论,以及考虑了概念随时间的变化以及继承关系中的属性变化。该模型一定程度上是将本体中的概念进行不同层次上的抽象,从而增强本体的表达能力。

3 基于元本体的多 Agent 本体协商

基于元本体的多 Agent 本体协商视图结合元本体理论

和本体协商的思想来解决本体异构的问题。该方法主要是在本体协商过程中以元本体为基础,这样元本体就为本体协商最终达成一致提供了一个客观标准,并在一定程度上提高了 Agent 之间进行协作的效率。

3.1 元本体理论

元本体理论将本体分为多个抽象层次,各个层次上的本体具有不同的抽象级别和普遍适应性。元本体包括描述本体自身的基本概念、必要假设、相关约束以及在此基础上的必要扩充。元本体是最抽象的本体知识,该部分本体适用于任意的应用领域(与应用领域以及世界观都无关),其包含的概念比较少。因此,元本体对于所有的本体库来说应该是一致的。各多 Agent 系统的本体可以基于元本体在不同的抽象层次上进行扩充来获得。这样不同的多 Agent 系统的本体之间在一定的抽象层次上都存在一致性。如果这些基于不同本体的 Agent 能够利用这些一致性来进行通信与协作,这样协作的效率相对会有所提高。Austin Tate 等人就试图利用元本体理论来统一计划的表示,即提出了基于本体的计划表示模型^[7]。此外,IEEE 的 Standard Upper Ontology 也尝试着通过元本体来建立一个普遍使用的应用领域本体,但结果似乎不是很理想。

图1是计划本体中本体的一个结构图^[8],但是对一般的元本体理论这个结构也是适用的。当然本体的抽象层次比图1中的抽象层次多得多。

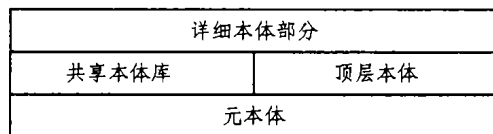


图1 本体层次结构图

元本体主要用来描述本体本身的元本体元素以及相关的假设。顶层本体是在详细本体部分的表示框架中所用到的最小本体集合。详细本体部分是对顶层本体进行扩展和细化。共享本体元素库主要包括一些在详细本体部分中共享的一些本体元素,但是这些元素不需要在顶层本体中进行描述。建立共享本体部分的目标是为了确保系统能够较容易地进行集成,并且共享部分将在详细本体部分进行标准化。详细本体部分是说明本体细节的特殊部分。这一部分反映了在该领域的经验知识,同时也反映了本体某些方面的责任划分。这一部分将顶层本体描述得更为详细,同时运用了共享本体库中的元本体元素。每个 Agent 的本体知识都包含有从元本体到最具体的详细本体的所有本体知识。并且在不同的抽象本体层次都具有就下一个抽象层次上的解释、请求解释以及确认解释原语。

3.2 基于元本体的多 Agent 本体协商

基于元本体的本体协商主要是针对本体协商过程当中, Agent 之间进行协商是效率比较低,并且用于协商的通信原语不能比较灵活定义的缺点在本体协商中引入了元本体理论的思想。在协商过程当中加入了不同抽象层次的本体来提高 Agent 之间协商的效率。

基于元本体的多 Agent 协商过程在最具体的抽象层次所引入的附加解释和说明性的原语和本体协商中的原语保持一致。而且所有这些原语都在上一个抽象层次的共享本体中做了说明。除了这些原语以外,根据具体应用的需要可以在不同的本体抽象层次上说明相应的通信以及解释原语。多 Agent 系统之间基于元本体进行本体协商的过程和一般的本

体协商过程基本上是一致的。但是在协商过程当中,由于系统自身可以在某一抽象层次上理解通信消息,而不需要通过请求说明来辅助理解。这样 Agent 之间通信的次数要减少,同时扩展的通信原语也将减少,而且这种理解的正确性应该比较高,因此,在请求确认时 Agent 大多情况下都能够得到确定的回答。基于元本体的本体协商过程基本如图2所示。

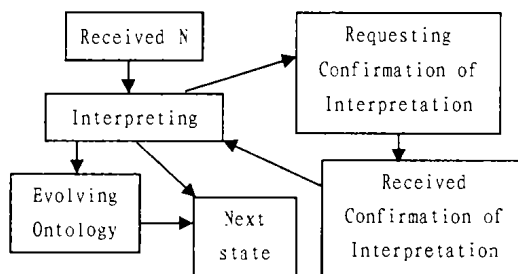


图2 基于元本体的本体协商过程

本体协商主要包括三个阶段:理解、理解确认与相关性分析以及本体进化。如果 Agent 接收到一条消息,Agent 自身对其按自身的本体进行理解。Agent 根据其自身的本体不断地进行抽象,直到 Agent 能够理解该通信消息内容为止。之后 Agent 将自己理解的结果发送给对方予以确认。由于各 Agent 的本体总在一定的抽象层次基于相同的元本体,Agent 的理解获得确认的可能性比较高。当 Agent 的理解出现错误时,对方将根据理解加以一定的说明后再将消息传送给 Agent 理解,重复以上过程直到 Agent 的理解获得确认。Agent 根据自身最后的理解以及自身的本体知识,对其相关性加以分析,并对原本体知识加以修改以完成本体知识的进化过程。此过程是相互的,进行协作的两个 Agent 都可能要对本体进行进化。

4 实例

下面将通过一个简单的实例来说明元本体的基本结构以及基于元本体的本体协商过程。假设多 Agent 系统的本体结构如下:多 Agent 系统基于不同的本体知识,但是这些本体知识在某个抽象层次上是完全一致的。

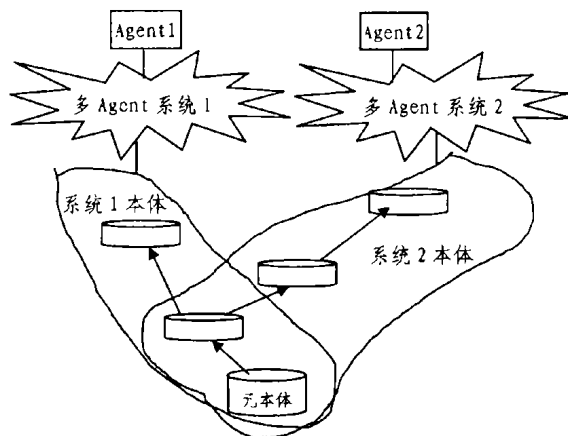


图3 多 Agent 系统本体组成关系

Agent1和 Agent2的协作过程基本描述如下: Agent1向 Agent2发送一条协作消息(KQML 消息), Agent2接收到消息以后首先在最具体的抽象层次上理解,由于两个系统的本体并不一致, Agent2可能不能完全理解,此时 Agent2不断将所收到的消息进行抽象,直到 Agent2自身认为已经正确理解的

(下转第187页)

为 $O(n \times \log n)$ 。

5 算法的实验结果及分析

在测试平台 Windows 2000 Professional, CPU 为赛扬 III, 内存 256M 下, 我们实现了文[3]中的算法和本文提出的算法。在算法中遗传算法的种群数为 30, 遗传代数为 50, 交叉概率为 0.90, 变异概率为 0.05, 终止条件为适应度达到 0.980, 调整系数 $w_1=0.05, w_2=0.95$ 。仿真结果列于表 1 中。

表 1 仿真实验结果

序号	样本数目	属性个数	核属性个数	文[3]中算法		本文中算法	
				代数	时间(秒)	代数	时间(秒)
1	100	10	4	15	1.192	7	0.2
2	100	20	6	25	3.356	12	1.342
3	1000	20	8	32	36.465	18	24.802
4	4000	22	8	36	164.607	24	108.548
5	10000	26	10	44	422.095	29	311.206

由于本算法是基于核属性的, 因此在数据样本的选择上, 我们选择了核属性个数较多的数据进行测试, 实验表明, 本算法在属性较多且具有核属性时较文[3]中的算法有较明显的优势。

结束语 本文所提出的方法利用了决策系统本身的启发信息, 通过采用系统核属性的改进算法, 使得时间复杂度降为 $O(m \times n \times \log n)$ 。本文给出的算法的特点是充分利用属性核信息, 提高算法的效率, 采用简洁的编码方式满足算法思想,

利用遗传提高全局搜索的能力, 避免局部最优。合适的适应函数的选择也是此方法的一大特点。

实验表明, 在样本具有核属性时是一个较文[3]中的算法更有效的算法。

参考文献

- 1 Khoo L-P, Zhai L-Y. A Prototype Genetic Algorithm-enhanced Rought Set-based Rule Induction System. *Computers in Industry*, 2000, 46: 95~106
- 2 王国胤. Rough 集理论与知识获取. 西安: 西安交通大学出版社, 2001
- 3 熊晖, 肖人岳. 用遗传算法求解粗糙集约简的改进算法. *计算机科学*, 2002, 29(9)
- 4 [Http://www.idi.ntnu.no/~aleks/rosetta/help/manual.pdf](http://www.idi.ntnu.no/~aleks/rosetta/help/manual.pdf)
- 5 Skowron A. Rough Sets in KDD, Presented during PAKDD 2000, Kyoto, Japan Pacific-Asia Conference on Knowledge Discovery and Data mining
- 6 Hu X. Knowledge discovery in databases: An attributes-oriented Rough Sets approach. [Ph. D. Thesis]. University of Regina, 1995
- 7 Nguyen S H, Nguyen H S. Some Efficient Algorithms for Rough Set Methods. In: Proc. of the Conf. of Information Processing and Management of Uncertainty in Knowledge-Based Systems IPMU'96, Granada, Spain, 1996. 1451~1456
- 8 尹旭日, 陈世福. 一种基于 Rough 集缺省规则挖掘算法. *计算机研究发展*, 2000, 37(12): 1441~1445
- 9 石云, 孙玉芳, 左春. 基于 Rough Set 的空间数据分类方法. *软件学报*, 2000, 11(5): 673~678
- 10 Ester M, Kriegel H-P, Sander J. Knowledge Discovery in Spatial Databases. *Lecture notes in computer science*, 1999, 1701: 61~74

(上接第 165 页)

层次(可能不是共享的抽象层次)。Agent2 将自身的理解发送给 Agent1, Agent1 根据 Agent2 的理解情况以及自身的本体对 Agent2 的理解予以确认或者是加以更多的解释。如果加以了更多的解释, Agent2 将重复以上的理解过程直到收到 Agent1 的确认消息为止。收到/发送确认消息以后, Agent1 和 Agent2 将分别地根据最终理解结果以及自身的本体知识进行相关性分析, 以完成各自的本体进化过程。

总结 多 Agent 系统之间的本体异构对知识的共享与集成以及 Agent 之间的协作都是一个非常大的障碍。为了解决这个问题, 我们提出了一种结合元本体理论以及本体协商的方法。该方法以本体协商为框架, 基于元本体进行协商。该方法通过引入元本体使得 Agent 之间就本体知识进行协商的效率大大提高。由于本体协商过程中不再需要进行请求说明, 使得协商的原语大大减少。同时将协商的扩展原语在本体中加以定义也提高了其灵活性。

但是, 该方法必须基于一定的元本体, 如何建立普遍的通用的元本体是该方法所面临的一个重大问题。只有建立了相应的标准, 才能建立一个大家都认可的元本体知识。此外, 该方法中 Agent 进行理解和抽象时如何判定其最终的抽象层次也是尚待研究的问题。由于 Agent 理解时需要不断地进行抽象, 因此 Agent 理解可能需要更多的时间, 但是与通过通信来进行说明相比, Agent 自身进行抽象可能是一个更好的

方法。

参考文献

- 1 Uschold M. Ontologies: Principles Methods and Applications. *The Knowledge Engineering Review*, 1996, 11(2): 93~136
- 2 Cranefield S, Willmott S. Introduction to the Special Issue on Ontologies in Agents Systems. *The Knowledge Engineering Review*, 2002, 17(1): 1~5
- 3 金芝. 知识工程中的本体论研究. 世纪之交的知识工程, 陆汝铃编著, 2001, 6: 447~465
- 4 Pease A, Niles I. IEEE Standard Upper Ontology: a Progress Report. *The Knowledge Engineering Review*, 2002, 17(1): 65~70
- 5 Bailin S C, Truszkowski W. Ontology Negotiation Between Intelligent Information Agents. *The Knowledge Engineering Review*, 2002, 17(1): 7~19
- 6 Tamma V, Bench-Capon T. An Ontology model to facilitate knowledge-sharing in multi-agent systems. *The Knowledge Engineering Review*, 2002, 17(1): 41~60
- 7 Tate A. Towards a Plan Ontology. *AI-IA Notizie, Special Issue on Aspects of Planning Research*, 9(1): 19~26
- 8 刘志忠, 姚莉. 基于本体的计划表示. 见: 第七届中国人工智能年会, 中国, 广西
- 9 Dimitrov M, et al. OntoMap: Upper-Ontology Service Agent. In: Proc. of Workshop on Ontologies in Agent Systems Autonomous Agent 2001, Montreal Canada on May, 2001
- 10 Omelayenko B. RDFT: A Mapping Meta-Ontology for Business Integration. In: Proc. of Workshop on Knowledge Transaction for Semantic for Semantic Web at the 15th European Conf. on AI (KTSW-2002), Lyon France 2002. 77~84