

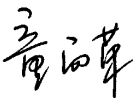




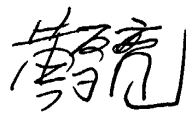
学位论文版权使用授权书

本学位论文作者完全了解北京交通大学有关保留、使用学位论文的规定。特授权北京交通大学可以将学位论文的全部或部分内容编入有关数据库进行检索，并采用影印、缩印或扫描等复制手段保存、汇编以供查阅和借阅。同意学校向国家有关部门或机构送交论文的复印件和磁盘。

(保密的学位论文在解密后适用本授权说明)

学位论文作者签名: 

签字日期: 2009 年 12 月 2 日

导师签名: 

签字日期: 2009 年 12 月 2 日

中图分类号: TP18
UDC:

学校代码: 10004
密级: 公开

北京交通大学

博士学位论文

多 Agent 合作求解中的信任与协商研究

Research on Trust and Negotiation in Cooperative Solving of
Multiple Agents

作者姓名: 童向荣

学 号: 06112066

导师姓名: 黄厚宽

职 称: 教授

学位类别: 工学

学位级别: 博士

学科专业: 计算机应用技术

研究方向: 多 Agent 系统

北京交通大学

2009年9月

致谢

首先，值此论文完成之际，谨向尊敬的导师黄厚宽教授致以衷心的感谢！感谢导师几年来的辛勤培养，感谢他为我们取得的每一点进步所付出的巨大心血。导师以年过六旬的高龄仍然奋斗在科研工作的第一线，无论是书写论文还是具体实验，导师都是我们的榜样。正是导师以他渊博的学识、深邃的思想、不倦的教诲和悉心的指导，我的博士论文才得以顺利地完成。导师严谨的治学态度、勤勉的工作精神、务实的生活作风和睿智的学者风范给我留下了深刻的印象；他对科学的热情和对自己信念的执著将会对我今后的学习、工作和生活产生潜移默化的长远影响，令我终身受益。导师不仅为我创造了良好的学习和科研环境，同时，在日常生活中给予了我无微不至的关怀和照顾，从师近四年来，使我受益颇多、终生难忘。我会永远记住黄老师为我为所获得的每一次成功而露出的欣慰的笑容。

在本文的研究与撰写过程中，得到了清华大学石纯一教授热心的指导和无私的帮助，正是他在研究过程中对我提出的宝贵意见让我能够按期完成博士论文工作。在此对石老师表示深深的谢意。同时还要感谢田盛丰教授、于剑教授、王志海教授、林友芳副教授、瞿有利高工等给予的大力帮助，他们热情和深刻而富有洞察力的见解使我获益良多。

我有幸参与了陆汝钤院士牵头的国家自然科学基金重大项目“非规范知识处理的基本理论和核心技术”的研究工作，在与国内外知名学者的交流中获益颇多，理论水平和眼界得到很大提高。在此特别对他们表示感谢。

在博士学习期间，还得到实验室许多同学的帮助，与各位同学的讨论和交流使我受益匪浅，使我得到了很多有价值的启示和建议，他们是：牟少敏、赵翔、董兴业、邓大勇、裴振奎、董红斌、张莹、赵静、陈萍、李向军、李昆仑、贺志、李万、穆成坡、邵超、王黎明、陈景年、汪廷华、恽佳丽、张小平、周丽萍、李冬梅、廖年冬、蔺旭东、蔺源、秦伦明、李宁、凌明清、胡琦、贺利坚、王中锋、冯奇、朱岩、肖宇、刘佳等。他们的真诚合作与帮助使我愉快地度过了三年多的学习生活，在此一并向他们表示诚挚的谢意！

特别提出要感谢我的父母和岳母，他们的关心、理解和支持是我顺利完成学业的保障，尤其是我的父亲，他的支持和帮助对我影响非常大。同时还要感谢我的妻子和孩子，博士期间大部分工作正是在她们的支持下完成的。

感谢以色列的 R. Lin 博士，与他的多次交流使我受益匪浅。感谢《Expert Systems With Applications》杂志社、《计算机研究与发展》杂志社对我的论文提出的意见。

最后，谨以本文献给所有关心、帮助和支持我的老师、亲人和朋友们！

中文摘要

在大规模开放环境中, Agent 不可避免地要与其它 Agent 反复交互以实现其合作求解目标, 因而每个 Agent 都会形成自己的历史交互。智能 Agent 自然能够从中得到更多的信息和知识并将其应用于之后的交互中, 优化多 Agent 的合作求解效果。

在 Agent 理论中, 信任与协商是两个基本问题。信任是合作求解的前提和基础, 协商是合作求解的基本方法, 因而信任和协商关系密切。信任计算是一个有意义的研究方向, 它能够保证软件 Agent 在大规模开放环境中的良好交互。不完全信息条件下的多议题协商一直是多 Agent 系统合作求解研究的一个至关重要又有挑战性的课题, 一直是众多学者关注的焦点之一。然而, 多 Agent 合作求解中的信任与协商问题至今没有得到很好的解决, 存在一些可以继续改进的地方。从多 Agent 的交互历史中学习, 得到信任计算和协商优化方法, 正是本文研究的动机。

关于多 Agent 合作求解中的信任与协商研究, 主要存在以下问题。

目前 Agent 信任研究大多是基于概率论以平均交互成功率来计算, 较少考虑信任动态变化, 因而信任的准确预测和异常行为的检测能力不能令人满意。另外, 很少有工作致力于长期联盟信誉的研究。而且开放网络环境中存在着大量不精确和不完全信息, 导致信任计算置信度不高, 如何提高信任的应对噪声能力仍然需要进行探讨。

在目前不完全信息条件下的 Agent 多议题协商中, 最优回价策略一般采用间接学习对手偏好的方式, 尚不能令人满意, 而实际上 Agent 一般拥有或多或少的协商经验和领域知识, 目前这些经验和知识都未得到很好的利用。多议题协商中效用函数的选择一直没有得到应有的重视, 很多学者采用了线性的效用函数, 这导致了在计算 Agent 的协商效用时涵盖范围较小。

本文针对以上问题开展工作, 主要工作如下:

(1) 提出了一种 Agent 动态交互信任计算模型。以概率论为工具, 按时间分段交互历史信息, 结合信任的变化率, 给出信任计算的置信度和异常行为检测机制。实验以网上电子商务为背景, 实验结果表明预测误差比 TRAVOS

少一倍，计算量也较少；改进了 Jennings 等人关于 Agent 信任的工作。

(2) 提出了一种 Agent 长期联盟信誉模型 LCCM。还给出了联盟信誉与联盟收益之间的关系函数。实验结果表明 LCCM 能够有效地计算联盟信誉，并能反映不同参数对联盟信誉的影响。

(3) 提出了一种不完全信息条件下基于案例和对策论的 Agent 多议题 Pareto 最优协商模型。当案例库规模控制在一定范围内时低于 Fatima 工作的计算复杂度。实验结果表明该协商模型能够取得更优的效用和更短的达成一致时间。改进了 Fatima 等人的工作。

(4) 将多议题协商的效用函数由线性扩展为非线性，基于 Sigmoid 函数提出了一种改进的符合边际效用递减原理的效用函数，给出了一种两阶段多资源配置协商模型和可行的算法。其算法的计算复杂度为多项式级。实验结果显示该模型的优化效率高于其它协商模型和算法。

关键词：多 Agent 系统，合作求解，信任，协商，动态信任，长期联盟信誉，多议题协商，多资源配置

分类号：TP18

ABSTRACT

Generally, there are iterative interactions among agents in large open environment. So each agent usually forms its own interaction history. Sequentially, intelligent agent would derive some useful information and knowledge from history data to be reused in the future for optimizing the performance of interactions.

Trust and negotiation are two important problems in cooperative solving of multi-agent systems. Trust is the base of cooperative solving and negotiation is a basic method of cooperative solving. So it is necessary to research on trust and negotiation together. Computation of trust is an interesting direction in multi-agent systems, for good trust relationship would guarantee the success of the future interactions in large open environment. Furthermore, multi-issue negotiation with incomplete information is always an important and challenging problem in cooperative solving. Unfortunately, trust and negotiation have not been resolved ideally up to now. It is the motivation of this dissertation to obtain optimal methods of trust and negotiation from the history of interactions.

There are some shortages in the researches of trust and negotiation of multi-agent systems.

Previous work on trust is only based on the average probability of historical interactions and there is a lack of attention to dynamic variety of agent trust. So the ability of precise prediction of trust and abnormal behavior detection is not satisfied. Few studies have been done on agent coalition credit to this day and there is a need to investigate it in detail. Furthermore, there are lots of imprecise and lying information in large open environment, which leads to a low confidence of trust computation.

Previous work on multi-issue negotiation usually uses indirect approaches to acquire the preferences of the opponent such as a variety of data mining methods. On the other hand, agents usually have some negotiation experiences and domain knowledge which may help them get better negotiation results. Furthermore, the choice of utility functions has not been paid more attention to. Previous papers mostly adopted linear utility functions which is not widely used in most circumstances.

To this end, this dissertation introduces the followings.

(1) We propose a computational model of agent dynamic interaction trust (CMAIT), where interaction history is divided by time. Sequentially, based on the first derivative

of trust, we give the confidence of computational information and that of computational deviation of CMAIT. The mechanism of abnormal behavior's detection of CMAIT is also given. We conduct Experiments on E-commerce at taobao website. Experimental results demonstrate that the computational error of CMAIT is half of that of TRAVOS model and its computational complexity is also lower than TRAVOS model. It improves the work of Jennings on agent trust.

(2) We present a long-term coalition credit model (LCCM). Sequentially, the relationship between coalition credit and coalition payoff is also given. Generalization of LCCM can be demonstrated through experiments applied in both cooperative and competitive environment. Experimental results show that LCCM is capable of coalition credit computation efficiently and can properly reflect the effect of various factors on coalition credit.

(3) We propose an agent multi-issue negotiation model under incomplete information based on cases and game theory. The computational complexity of the proposed algorithm is polynomial order and it is commonly lower than that of Fatima, as long as the scale of cases base is limited to a bounded quantities. Experimental results indicate that the utility and the reaching time of our experiments have an advantage of that of human beings and the method of Lin. It improves the work of Fatima.

(4) We expand linear utility function to a nonlinear one. Particularly, we propose an improved utility function based on sigmoid function in neural network, according to the principle of marginal utility decreasing. Sequentially, we present a negotiation model over multiple divisible resources with two phases, as well as its feasible algorithm. The computational complexity of this model is polynomial order. Experimental results show that the optimal efficiency of this model takes an advantage over the previous work.

KERWORDS: Multi-agent systems, Cooperative solving, Trust, Negotiation, Dynamic trust, Long-term coalition credit, Multi-issue negotiation, Multiple resources allocation.

CLASSNO: TP18

目录

中文摘要.....	iii
ABSTRACT.....	V
第一章 绪论.....	1
1.1 Agent 理论和技术的发展.....	1
1.2 课题背景与意义.....	4
1.3 多 Agent 合作求解研究进展.....	5
1.4 存在的问题.....	6
1.5 论文的主要工作和创新点.....	7
1.6 论文的组织结构.....	8
第二章 多 Agent 合作求解.....	9
2.1 基础知识.....	9
2.1.1 概率论.....	9
2.1.2 对策论.....	10
2.2 联盟.....	11
2.2.1 联盟结构生成.....	11
2.2.2 长期联盟.....	12
2.3 信任机制.....	13
2.3.1 Beth 信任度评估模型.....	15
2.3.2 Jøsang 信任度评估模型.....	15
2.3.3 FIRE 模型.....	16
2.3.4 TRAVOS 模型.....	18
2.4 协商.....	19
2.4.1 协商的基本模型.....	20
2.4.2 协商的策略与学习.....	22
2.4.3 协商策略的改进.....	23
2.4.4 拍卖.....	24
2.5 小结.....	28
第三章 一种 Agent 动态交互信任预测与异常行为检测模型.....	29
3.1 引言.....	29
3.2 相关工作分析.....	30
3.3 问题描述.....	32

3.4 Agent 交互信任计算.....	34
3.4.1 计算原理.....	34
3.4.2 动态交互信任计算.....	35
3.4.3 置信度计算.....	37
3.4.4 编程.....	38
3.5 实验结果及分析.....	39
3.5.1 实验背景和参数.....	39
3.5.2 信任计算实验.....	40
3.6 小结	44
第四章 一种 Agent 长期联盟信誉模型	45
4.1 引言	45
4.2 问题描述	47
4.3 长期联盟信誉模型.....	50
4.3.1 长期联盟信任计算.....	51
4.3.2 长期联盟声誉计算.....	53
4.3.3 编程.....	55
4.4 实验	56
4.4.1 电子商务实验.....	56
4.4.2 不同参数实验.....	58
4.4.3 实验 2.....	60
4.4.4 实验 3.....	62
4.5 小结	64
第五章 一种基于案例的 Agent 多议题协商模型	65
5.1 引言	65
5.2 问题描述	66
5.2.1 协商场景.....	66
5.2.2 Agent 信息状态.....	67
5.2.3 相似度评估.....	69
5.2.4 一个例子.....	69
5.3 协商协议	71
5.3.1 接受条件.....	71
5.3.2 最优回价的产生.....	72
5.3.3 算法.....	73
5.3.4 与其它工作的比较.....	75

5.4 实验.....	76
5.4.1 实验设置.....	76
5.4.2 实验方法.....	78
5.4.3 实验结果.....	79
5.4.4 协商过程例子.....	80
5.5 小结.....	81
第六章 一种非线性效用函数的 Agent 多资源配置协商模型.....	83
6.1 引言.....	83
6.2 问题描述.....	84
6.3 两阶段的协商模型.....	87
6.3.1 第一阶段.....	87
6.3.2 第二阶段.....	89
6.4 实验.....	92
6.4.1 实验结果与分析.....	92
6.4.2 不同类型 Agent 的实验.....	93
6.4.3 与 Fatima 等人工作的比较.....	94
6.5 小结.....	95
第七章 结束语.....	96
7.1 本文工作的总结.....	96
7.2 进一步研究的展望.....	97
参考文献.....	99
攻读博士期间发表和已录用的学术论文.....	105
独创性声明.....	106
学位论文数据集.....	107



第一章 绪论

1.1 Agent 理论和技术的发展

Minsky 在 1986 年出版的《思维的社会》一书中提出了 Agent, 认为社会中的某些个体经过协商之后可以求得问题的解, 这些个体就是 Agent, Agent 应具有社会交互性和智能性。Hewitt 认为定义 Agent 与定义智能一样困难。Woodridge 和 Jennings 等人^[76-77]认为 Agent 应具有自治性、社会交互性、反应能力、预动能力。

Agent 一般可译为智能体, 主体, 代理等。Agent 是一个程序, 它有知识, 有智能。Wooldridge 和 Jennings 等人^[76]在 1995 年给出了其定义: Agent 是处在某个环境中的计算机系统, 该系统有能力在这个环境中自主行动以实现其设计目标, 如图 1.1 所示。一群相互作用的 Agent 的集合称为 Multi-Agent Systems(MAS, 多 Agent 系统), 如图 1.2 所示。MAS 是分布式程序设计深入发展的产物, 它已经成为 Agent 研究的主要内容。

例如一个恒温调节器可看成一个对环境变化具有反应能力的 Agent。多个机场协调航班控制, 可以看成是 MAS 合作求解。协商和拍卖, 可看成是 MAS 通过谈判协调利益。机器人足球赛可看成是 MAS 之间的对抗。一般地说, 可认为 MAS 是分布式自治计算系统。

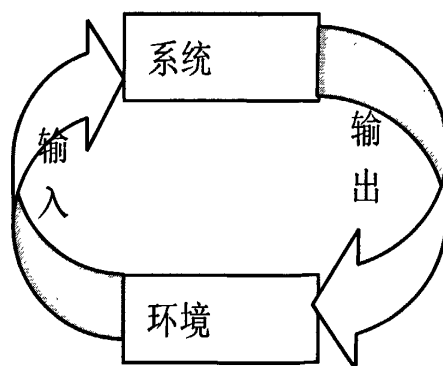


图 1.1 个体 Agent 示例

Figure 1.1 An example of individual agent

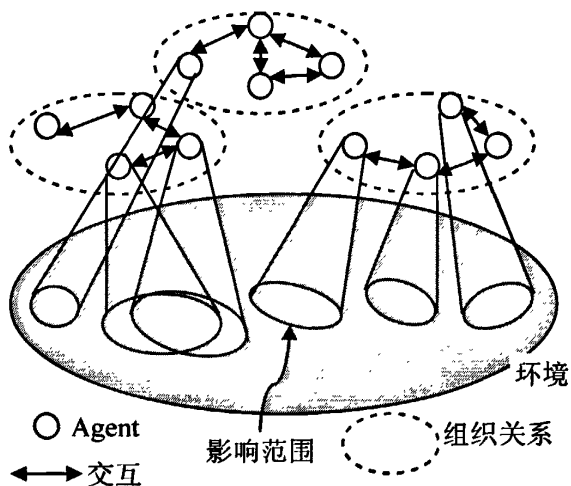


图 1.2 多 Agent 系统示例

Figure 1.2 An example of multi-agent systems

多 Agent 系统是计算机科学中一个新的分支，其研究始于 20 世纪 80 年代，直到 20 世纪 90 年代中才得到广泛的认同，如为广大研究者所熟知的 BDI 模型及其改进工作^[56-57,96-97,100-101,109-110,115]。此后国际上对这一领域的兴趣大为增加，目前，该领域已经成为人工智能中一个非常活跃的研究分支。

Agent 技术的研究以欧洲居多，尤其以英国利物浦大学的 Wooldridge 和南安普敦大学的 Jennings 为代表。国外的 Agent 研究论文主要发表在《Artificial Intelligence》杂志、《Journal of Artificial Intelligence Research》杂志、《Journal of Autonomous Agents and Multi-Agent Systems》杂志，以及 IJCAI（国际联合人工智能大会）、AAMAS（国际自治 Agent 和多 Agent 系统大会）、IAT（国际智能 Agent 技术大会）等系列会议上。国内对 Agent 和 MAS 的研究始于 20 世纪 90 年代，主要集中在 Agent 模型、MAS 合作求解方法和面向 Agent 的软件工程等方面，也有很多学者关注于 Agent 技术的应用。目前已举办了两届全国 Agent 技术与应用学术会议，第三届会议将于 2010 年在国防科技大学召开。

AgentLink III（2004-2005）是由欧盟资助的基于 Agent 技术的计划，由利物浦大学的 Peter McBurney 教授和南安普敦大学的 Terry Payne 教授领导。截至 2005 年底，已有 206 个组织加入了 AgentLink III，包括 134 所大学，32 所研究机构和 40 家公司。AgentLink III 的目标是使欧洲处于 Agent 研究的国际领先地位，促进和提高 Agent 技术的普及率，以获得欧洲工业的竞争性优势；支持 Agent 技术的标准化并促进相互协作，推进欧洲在基于 Agent 的计算机系统研究的质量、框架和相关工业方面的改进。

多 Agent 系统与计算机科学和人工智能等领域相关，是一个高度交叉的研究领域，它吸取了不同领域的内容，如经济学^[5,10-11,27,30,62-63,80-83,103-106]、哲学^[64,66,115]、

逻辑学^[2-4,40,89,114,143]、生态学和社会科学等。Agent 是一种合适的软件范例^[77], 为研究大规模分布式开放系统提供了可能, 对于推动软件工程的进展起到了很大的作用^[33,87-88,107,140]。现在业界已经公认, 交互是复杂软件最重要的一个特征。软件结构中有很多动态的互相交互的构件, 每个构件都有自己的控制线程, 并且服从复杂的合作协议, 这样的结构相当复杂, 实际上, 许多应用恰恰具有这样的特点。许多研究人员相信, 将来计算本身可以理解为主要是一个交互过程, 可以把许多系统理解成由交互的、自治的 Agent 组成, 而这种认识增加了把 Agent 作为软件工程新风格的兴趣。对于理解和构造各种所谓的人工社会系统来说, 多 Agent 系统似乎是一个自然的比喻。多 Agent 系统也出现在许多不同的应用领域中, 如语言处理、工业制造、信息系统组织、交通控制、分布传感和解释、运输调度、监控以及机器人等。

目前学术界对多 Agent 系统的认识和理解有所差别, 因而有必要对此做一个简单的说明。Wooldridge^[77]给出多 Agent 系统的特性如下。

(1) 多 Agent 系统不完全是分布式/并发系统

因为 Agent 是自治的, 因而很难保证多 Agent 系统的同步和协调动作。而且多 Agent 之间有利益的冲突, 因而它们之间的目标可能不一致。这不同于分布式或并发系统, 因为它们一般拥有共同的目标和集中式控制。

(2) 多 Agent 系统不完全是人工智能

人工智能关注于智能的构成, 如学习能力、规划能力和图像理解能力等, 而多 Agent 系统更多的关注于这些成分的集成, 重点是合作的机制研究。正如 Oren Etzioni^[18]明确指出的: “智能 Agent 是 99% 的计算机科学, 1% 的人工智能”。其次, 经典的人工智能忽略了 Agent 的社会性, 而 Agent 则具有通信、合作、达成一致等能力。这种能力在多 Agent 系统合作求解中显得尤为重要。

(3) 多 Agent 系统不完全是经济学和对策论

对策论是一种数学理论, 用来研究自利的 Agent 之间的交互问题。对策论历来就与计算机科学紧密相关, 而且现在更成为计算机科学研究的核心工具之一。Neumann^[47]作为计算机科学的奠基人之一, 也是对策论的创始人之一; Alan Turing^[74]是计算机技术的另一个创始人, 他对研究对策论也很有兴趣, 他的论文《计算机器和智能》可以被看做是人工智能理论的基础。近年来, 对策论的工具和技术在求解多 Agent 系统的研究中有许多应用, 特别是用来解决协商问题^[12-13,19-23,34,38-39,41-43,46,53,55,59,71,91-92,124,126-129,131], 因而对策论已经成为分析多 Agent 系统的主要理论工具。然而对策论中的许多解决方案和计算理论无关, 它们只是描述性的概念和特点, 而没有告诉我们如何计算出这个解决方案。

(4) 多 Agent 系统不完全是社会科学

在多 Agent 系统的研究中, 观察和模仿人类社会是有用的, 但是并不能由此得出可以用完全同样的方式来建造人类社会的结论。众所周知, 完全准确地模仿人类社会行为是困难的, 因为人类社会的行为依赖于太多的因素。而且, 尽管使用对人类社会模仿和比喻的方法设计多 Agent 系统是合理的, 但并非说这种方法就是最好的。

1.2 课题背景与意义

多 Agent 之间的合作求解是多 Agent 系统研究的核心内容之一, 多个 Agent 之间为了完成所设定的任务必须和其它 Agent 相互作用, 共同实现目标。信任是合作求解的基础, 协商是合作求解的基本方法。多 Agent 合作求解一般以信任 [7,9,25,29,31-32,35-37,58,69-70,73,84,90,94-95,108,113,116,123,125,130,132] 为基础, 根据自身的效用函数, 通过通信 [120]、联盟 [60-61,67-68,78-79]、协商 [19-23,26]、学习 [133-134,142,144]、组织 [24,93,112,135-139,141] 等方式, 自治的选择合作方式, 最终实现自身效用最大化。

作为简化复杂系统的一种有效方法, 信任在社会学、心理学、经济学、管理学领域早已引起人们的关注。信任也在人类社会的交往中发挥着重要的作用。在开放的网络环境下, 多 Agent 系统中存在着大量的不确定、不完全、不精确信息, 甚至有欺骗、共谋等现象。多 Agent 系统作为分布式人工智能中的一个重要研究方向, 试图用 Agent 来模拟人的理性行为, 用多 Agent 合作求解解决智能的社会性问题。因此, 将信任引入多 Agent 合作求解成为一种必然的选择。

信任可视为一种主观信念, 是一种主观的、模糊的、由不精确和不完全信息推理而得到的对伙伴完成合作求解任务的信念。信任是“agency”和“on behalf of”的基础, 也是开放网络环境下多 Agent 交互的基础, 对联盟、团队、组织、合作、协商起关键作用。一般而言, 良好的信任能防止欺骗, 优化 Agent 对合作伙伴的选择, 降低 Agent 合作风险, 最终提高 Agent 合作求解中各方的收益。

在 Wooldridge、Jennings 和 Sandholm 等人关于联盟、协商、组织等合作求解的研究中, 多关注于 Agent 的对策和效用, 满足 Agent 自利性的要求, 前提假设通常是资源是可得的、服务是可靠的、信息是可信的。当多 Agent 系统合作求解的研究逐步向开放式网络环境扩展, 以上假设不再成立。开放的网络环境中存在诸多复杂的和不确定性的因素, 使信任成为多 Agent 合作求解面临的一个重要挑战。目前, 尽管在电子商务、P2P 网络、网格计算、医疗咨询等领域已有基于信任方法的局部应用, 但相对独立于应用的、在多 Agent 合作求解中对抽象的信任机制的全面的理论研究尚未见到。开展在开放网络环境下多 Agent 合作求解中的信任问

题研究,能够更加深刻地发现信任在多 Agent 合作求解中的作用,揭示信任在多 Agent 系统中的性质,在此基础上,提出针对具体应用需求的信任模型与方法。

协商是合作求解的基本和有效方法之一,不完全信息条件下的多 Agent 多议题协商研究是目前多 Agent 系统的研究重点和热点之一。多 Agent 协商可应用在电子商务、电子贸易、资源配置等方面,多 Agent 之间的协商能够使双方求同存异,通过通信和妥协等方式达成相互有利的一致,因为对参与协商的双方而言,成则双赢、分则双败。

目前,在金融危机的条件下,电子商务或许会成为下一个新的经济增长点,如淘宝网、亚马逊网、当当网等吸引了越来越多的创业者投身其中。调查显示,越来越多的大学毕业生选择了开网店创业。然而目前网上电子商务面临的一个问题是不能如传统商务那样进行面对面的交流和协商,缺少一种有效的协商机制和方法,因而导致交易成功率和效率较低。多 Agent 多资源配置的多议题协商将提供更有效的方法,力求使自主 Agent 完全可以代表人类进行网上协商,并能取得更好的网上电子交易结果。

多 Agent 多议题协商的研究将给网上电子商务提供一个更好的协商机制和方法,改变目前网上电子商务活动中单一的报价方式,从而可以大大提高电子商务的效率。Agent 自主协商必将大大改善人类的生活方式,把人类从繁杂的多议题协商中解放出来,提高社会生产力。

本论文的主要研究内容包括多 Agent 合作求解中的信任和协商问题。信任是合作求解的前提和基础,在信任的基础上进行合作求解才是有意义的,协商是合作求解的基本方法,二者之间的关系非常密切。

1.3 多 Agent 合作求解研究进展

Agent 协商、信任和联盟等已经成为了多 Agent 合作求解研究的主要内容。近年来,大量的文献给出了多 Agent 合作求解的框架、机制、算法等。这些问题的难点集中在有限理性、信息不完全和时间条件限制上,即如何在有限的资源、信息和时间条件下取得最大化收益,主要内容则是协议、机制和算法设计等。

联盟的研究主要集中于联盟结构生成^[8,15-17,44,49-52,65,72,75,98-99,102,111,118,121-122,145]和联盟收益分配这两个问题上,Sandholm 等人^[60-61]提出的最优联盟结构核能够保证个体 Agent 和 Agent 联盟的收益都实现最优,但确定核非空的计算复杂度被证明为 NP 完全的^[14]。Wooldridge 等人^[78-79]研究的定性联盟对策以 Agent 完成目标为约束求解定性的联盟结构生成。联盟逻辑^[2-4]也是重要的研究内容。

信任推理模型以认知逻辑、不精确推理等方法为工具，将信任视为一种信念进行推理与修正。信任计算模型以概率论为工具，基于交互历史修正信任关系，如 Beth 信任模型^[7]、Jøsang 信任模型^[35-36]、REGRET 模型^[58]、FIRE 模型^[31-32]、TRAVOS 模型^[69-70]等。Vassileva 等人^[75]给出基于经验的长期联盟信任函数，Mui 等人^[45]从社会学和进化论的角度给出群体 Agent 信任计算模型，Ramchurn 等人^[54]建立了一个多 Agent 交互的信任计算模型，Teacy 等人^[69-70]给出信息不完全条件下 Agent 信任计算模型 TRAVOS。其次是集成信任和声誉的方法，如 Ashri 等人^[6]给出基于信心和名誉的信任计算法，Huynh 等人^[31-32]提出开放多 Agent 系统的集成信任和声誉模型 FIRE，它集成了交互信任、基于角色的信任、旁观者声誉和鉴定声誉。但是，这些模型主要关注信任的表示、计算和传播，并未涉及如何在多 Agent 合作求解中使用这些模型。

开放环境不完全信息条件下的多 Agent 多议题协商是多 Agent 系统合作求解研究面临的一个重要挑战，涉及多资源分配^[5,20-21]、不可分割资源配置^[59]、无嫉妒的资源分配^[12]等方面。核心问题是哪种方法能够取得最优的协商效果（Pareto 最优），Fatima 等人^[21]指出只有一揽子交易多议题协商才能保证 Pareto 最优解。另外，有些学者提出了多议题之间的折衷算法^[43]与学习等^[38-39,46,91,128-129]，以实现 Pareto 最优，提高协商的效率。

多议题协商研究中的许多实验显示自主 Agent 协商能取得比人更好的效果，如 Kraus 等人^[39]的实验显示在加拿大和西班牙之间的渔业冲突问题协商中，Agent 可以至少和（在某种程度上好于）西班牙一方的人类谈判者一样好，Lin 等人^[41]设计的一种软件 Agent 则可以使自身效用和社会福利（双方效用之和）都扩大化。

1.4 存在的问题

在多 Agent 合作求解的信任和协商研究方面，目前存在如下几个问题。

（1）在 Agent 理论中，信任计算是一个有意义的研究方向，它能够保证软件 Agent 在大规模开放环境中的良好交互。然而目前 Agent 信任研究一般以平均交互成功率来计算，较少考虑信任动态变化，因而信任的准确预测和异常行为检测能力不能令人满意。

（2）个体 Agent 经常与长期联盟交互，如一些电子商务网站淘宝、卓越等，而 Agent 选择这些网站不能仅仅以效用为标准，还需要考虑到它们的信誉，因为网站的信誉会影响个体 Agent 的效用。然而，目前很少有工作致力于联盟信誉的研究，因此，这方面仍然需要进行探讨。

(3) 目前不完全信息条件下的 Agent 协商研究中, 最优回价策略一般采用间接学习对手偏好的方式, 尚不能令人满意, 如 Fatima、Luo 等人的工作; 另一方面, Agent 一般拥有或多或少的协商经验和领域知识, 这将帮助它们取得更好的协商结果, 因此这些经验有可利用的价值。

(4) 不完全信息条件下多资源配置的 Agent 多议题协商是一个有意义的课题, 目前大多将 Agent 分配得到的某种资源直接作为其议题效用使用, 是一种线性的效用函数。然而, 在很多情况下效用和资源之间的关系是饱和和非线性的, 这就需要扩展这方面的工作。

1.5 论文的主要工作和创新点

针对目前多 Agent 系统合作求解存在的问题, 我们尝试在 Agent 动态信任形成机制和异常行为监测方面做出深入的工作, 将静态信任研究引向动态模型、将个体信任扩展到联盟信誉。将基于案例的方法引入到多议题协商中, 节省回价生成的时间, 改进回价生成机制; 将线性效用函数扩展为非线性效用函数, 提高协商的效率。主要的研究内容有如下几点结果。

(1) 关于 Agent 动态信任的研究

提出一种 Agent 动态交互信任计算模型 CMAIT。以概率论为工具, 按时间分段交互历史信息, 计算得到 Agent 的动态信任; 结合信任的变化率, 给出信任计算的置信度和异常行为检测机制。实验结果表明该模型的预测误差比 TRAVOS 模型少一倍, 计算量也较少; 改进了 Jennings 等人关于 Agent 信任的工作。

(2) 关于 Agent 长期联盟信誉

提出了一种 Agent 长期联盟信誉模型 LCCM, 还给出了联盟信誉与联盟收益之间的关系函数。实验结果表明 LCCM 能够有效地计算联盟信誉, 并能反映不同参数对联盟信誉的影响。

(3) 基于案例的多议题协商研究

提出了不完全信息条件下, 基于案例和对策论的 Agent 多议题 Pareto 最优协商模型。相应算法的计算复杂度为多项式级, 且当案例库规模控制在一定范围内时低于 Fatima 等人工作的计算复杂度。实验结果表明该算法能够取得更优的效用和更短的达成一致时间, 改进了 Fatima 等人的工作。

(4) 基于非线性效用函数的多议题协商模型

提出一种改进的符合边际效用递减原理的效用函数, 给出了一种两阶段多资源一揽子交易协商模型和可行的算法, 第一阶段将资源视为不可分割的整体以达

成初步一致并推理获得完全信息，第二阶段用贪心算法将资源分割求得 Pareto 最优解。实验结果显示该模型的优化效率优于 Fatima 等人的工作。

1.6 论文的组织结构

本论文其余章节的安排如下。

第二章：给出本论文所研究内容的综述，包括多 Agent 合作求解的联盟研究进展、协商研究进展和信任研究进展及相应的基本概念和理论。

第三章：提出 Agent 动态交互信任预测与异常行为检测模型。以概率论为工具，按时间分段交互历史，给出 Agent 交互信任计算模型 CMAIT；结合信任的变化率，给出信任计算的置信度和异常行为检测机制。

第四章：提出一种长期联盟信誉模型(LCCM)，并给出联盟信誉和联盟收益之间的关系。实验结果显示 LCCM 能够有效的计算联盟信誉，并能反映不同参数对联盟信誉的影响，而且 LCCM 能够有效地过滤不精确或撒谎信息。

第五章：提出一种基于案例的 Agent 多议题协商模型，并通过实验验证该模型优于目前的工作。

第六章：提出一种改进的符合边际效用递减原理的效用函数，给出一种两阶段多资源一揽子交易协商模型和可行的算法，并通过实验结果显示该模型的优化效率优于 Fatima 等人的工作。

最后，给出全文的总结，并对今后的研究工作进行展望。

第二章 多 Agent 合作求解

合作求解是 MAS 研究的核心问题之一，交互是合作求解的基本形式，涉及信任、协商、拍卖、联盟等。一个完整的合作求解模型由群体 Agent 思维状态模型、合作求解过程以及规划库等部分组成。对于合作问题求解模型来说，Agent 之间是合作关系，一起完成某项任务，合作求解的研究内容主要是群体规划、合作策略等。

本章详细讨论多 Agent 合作求解的基本概念和理论，并介绍当前几种主要的合作求解方法，同时介绍其基本理论。主要包括联盟、协商、信任机制和拍卖等。

2.1 基础知识

2.1.1 概率论

本论文以概率论为基本工具研究信任计算，因此，本章首先简要回顾本论文所用到的概率论的基本知识，包括均匀分布和 Beta 分布。

均匀分布的参数是 $a < b$ ，其概率密度为

$$f(x) = \begin{cases} \frac{1}{b-a} & a < x < b \\ 0 & \text{其它} \end{cases} \quad (2.1)$$

其数学期望为

$$\frac{a+b}{2}$$

其方差为

$$\frac{(b-a)^2}{12}$$

Beta 分布的参数是 $\alpha > 0, \beta > 0$ ，其概率密度为

$$f(x) = \begin{cases} \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} x^{\alpha-1} (1-x)^{\beta-1} & 0 < x < 1 \\ 0 & \text{其它} \end{cases} \quad (2.2)$$

其数学期望为

$$\frac{\alpha}{\alpha + \beta}$$

其方差为

$$\frac{\alpha\beta}{(\alpha + \beta)^2(\alpha + \beta + 1)}$$

2.1.2 对策论

在本论文的协商研究中, Agent 采用的协商策略都基于对策论, 因此有必要对此做简要的介绍, 引用 Wooldridge 的《多 Agent 系统引论》的部分内容进行说明^[77]。

当两个 Agent 交互时, 假设每个 Agent 刚好可以执行两个可能的动作, 称这两个动作为 C (代表合作) 和 D (代表不合作)。设 $A_c = \{C, D\}$ 是这些动作的集合。环境改变的方式由函数

$$\tau: \underbrace{A_c}_{\text{Agent } i \text{ 的动作}} \times \underbrace{A_c}_{\text{Agent } j \text{ 的动作}} \rightarrow \Omega \quad (2.3)$$

决定 (这本质上是一个状态转移函数), 基于 Agent i 选择的动作 (C 或 D) 和 Agent j 选择的动作 (C 或 D) 会产生一个结局。例如:

$$\tau(C, C) = \omega_1, \quad \tau(C, D) = \omega_2, \quad \tau(D, C) = \omega_3, \quad \tau(D, D) = \omega_4$$

可以用下面的方式来刻画 Agent i 对于可能的结局的偏好。

$$\omega_1 \succ_i \omega_2 \succ_i \omega_3 \succ_i \omega_4$$

假设有两个 Ω 的子集 Ω_1 和 Ω_2 , $\Omega = \{\omega_1, \omega_2, \omega_3, \omega_4\}$ 。结局的集合 Ω_1 强优于 Ω_2 , 如果下面的条件为真。

$$\forall \omega_1 \in \Omega_1, \forall \omega_2 \in \Omega_2, \text{ 有 } \omega_1 \succ_i \omega_2$$

因此, 可以说策略 s_1 优于策略 s_2 , 如果执行 s_1 产生的可能的结局集合优于执行 s_2 的可能的集合, 即如果 s_1^* 优于 s_2^* 。

另一个理解优势策略的方式是如果有其它策略 s' 优于策略 s , 则一个理性的 Agent 不会遵循 s (因为遵循 s' 会更好)。

一般来说, 两个策略 s_1 和 s_2 处于 Nash 均衡, 如果

- (1) 在 Agent i 执行 s_1 的前提下, Agent j 最好执行 s_2 ;

(2) 在 Agent j 执行 s_2 的前提下, Agent i 最好执行 s_1 。

平衡的形式是重要的, 它把 Agent 锁定在一对策略上, 两个 Agent 中的任何一个都没有脱离 Nash 均衡的动机。在一个对局中, Nash 均衡策略的存在, 可能是对任何给定情况下采取什么动作的问题的最好回答。但是对策论中有两个重要的结论, 它们使决策产生困难。

(1) 并不是每个交互的情形都有 Nash 均衡;

(2) 有些交互的情形有一个以上的 Nash 均衡。

尽管有这些负面的结论, Nash 均衡仍然是一个重要的概念, 并且在多 Agent 系统合作求解分析中起着重要的作用。

2.2 联盟

在多 Agent 系统研究中, 联盟主要应用于电子商务, 多 Agent 系统形成联盟可以取得更大的收益, 获得超加性。联盟研究从合作对策论、经济学及分布式人工智能等方面入手。Sandholm 等人^[60-61]认为联盟研究主要包括联盟结构的生成, 即对 Agent 的划分; 以及联盟的收益如何在成员之间进行分配, 这种分配应满足 Pareto 最优, 同时还需考虑计算复杂度和时间成本。

2.2.1 联盟结构生成

Wooldridge 等人^[78-79]认为, 假设存在 Agent 集合

$$Ag = [1, 2, \dots, n]$$

则一般联盟 C 是 Ag 的真子集, 而平凡的联盟是 Ag 。生成联盟时需要对 Agent 进行分组划分, 以生成多个划分块 (联盟), 这种划分称为联盟结构 CS 。 CS 是 Ag 的互不相交的非空子集簇, 并且这些子集的并等于 Ag , 其形式化定义为

$$CS := (\cup CS = Ag) \wedge \forall B, \forall D (B, D \in CS \wedge B \neq D \neq \phi) \rightarrow B \cap D = \phi \quad (2.4)$$

含有 n 个 Agent 的 Ag 可生成的联盟个数为 $2^n - 1$; 联盟结构集 L 是所有可能生成的联盟结构集合, 即

$$L = \{CS_1, CS_2, \dots, CS_M\}$$

其中 M 是 n 个 Agent 所能生成的联盟结构数目。

联盟结构的生成问题主要解决在近似于天文数字的联盟结构图中搜索一个最优值。Sandholm 等人^[60-61]基于超加性提出了最坏情况下核稳定联盟设想。给出了联盟生成计算的最低要求,即必须搜索联盟树的最低两层。合作对策论核的概念解释了 Agent 应该加入的联盟^[60]。对策论的核是当前联盟结构 CS 的收益配置组 (\bar{x}, CS) 。Agent 留在核中比分离出去加入新联盟更能满足总体收益最大化,这样的解是稳定的。

$$Core = \left\{ (\bar{x}, CS) \mid \forall S \subset Ag, \sum_{i \in S} x_i \geq v_S \text{ and } \sum_{i \in Ag} x_i = \sum_{j \in CS} v_{S_j} \right\} \quad (2.5)$$

其中, \bar{x} 是收益分配, x_i 是 Agent i 的收益。核是联盟结构生成的最强约束,因而经常导致联盟核为空。

Conitzer 等人^[14]证明不论是否存在传递效用,确定核非空是 NP 完全问题。使这一问题变得困难是由于决定是否存在勾结(小团队)的可能性。Manistersk 等人^[44]特征化了核的复杂性,显示在有传递收益的情形下该问题属于 co-NP。Rahwan 等人^[51]提出了一种新的算法,将联盟值的计算分布到合作环境中的各个 Agent 上。算法是非集中式的,不需要 Agent 之间的通信,且有最小的内存需求,能反映 Agent 计算速度的变化。

2.2.2 长期联盟

以前的 Agent 联盟模型只维持一次交易的时间,当任务完成时联盟自动解散。但是联盟中 Agent 选择加入或退出联盟时没有先验知识为基础,不能从中学习,这增加了搜索适当联盟的时间、空间等资源消耗,另一方面联盟的多次形成和解散增加了联盟中 Agent 的不稳定性。针对以上问题,长期联盟适时的被提出来了,随之而来的是基于信任的长期联盟形成问题。

基于对策论, Vassileva 等人^[75]基于培育客户-投资者关系提出了电子市场长期联盟,认为建立一个考虑个体 Agent 长期效用、偏好及其在同伴中的信任关系的长期联盟,能减少 Agent 之间通信量并降低计算的复杂度。基于信任方程可以实现长期联盟的以上目标

$$trust : E \times T \rightarrow T \text{ and } trust(e, t) = dt + (1-d)e \quad (2.6)$$

其中 $E = [-1, 1]$ 是 Agent 的以往经验, 也可以认为是历史, $T = [-1, 1]$ 是 Agent 对其他 Agent 的信任值, $d \in [-1, 1]$ 是膨胀系数。式 2.6 表明 Agent 基于以往经验值和现在信任值形成新的信任值。

为避免不同目标 Agent 交互, Griffiths 等人基于动机和信任提出了 Agent 党派^[28]。一群相互信任、有着共同目标的 Agent 组成党派。通过对 Agent 动机建模, 其他 Agent 可以选择目标相同的 Agent 来形成联盟。这种机制的优点是简单、可扩展, 并且不需要 Agent 之间的通信, 它能减少系统的动态性, 保证个体 Agent 的利益更大化。

童向荣等人^[122]提出了基于模糊盟友关系的长期联盟生成。根据 Agent 之间的交互次数或频率来模糊推理信任关系, 通过监控 Agent 之间的交互更新联盟中的盟友关系, 该值与联盟收益有关, 降低了盟友关系的计算复杂度, 提高了盟友关系计算的实时性。

2.3 信任机制

在多 Agent 系统信任研究中, 有很多关于信任(Trust)、声誉(Reputation)、信誉(Credit)等提法, 此三者之间互相联系又互相区别。信任一般是多个 Agent 之间通过直接交互建立起来的对对手完成所承诺任务的期望, 一般以概率论为工具, 通过对交互历史的计算得到其数学期望。声誉是没有直接交互 Agent 之间通过间接的信息传递得到的间接信任, 此方法一般通过第三方的间接信息推理得到, 可能存在不精确、不完全、撒谎等信息, 因而置信度较低。信誉一般可理解为一种信任和声誉的加权和, 如本论文提出的长期联盟信誉模型, 将联盟的内部信任和外部声誉通过联盟内 Agent 数量来加权得到联盟信誉。

信任管理的概念最早由 Blaze 等人^[9]于 1996 年提出, 旨在“采用一种统一的方法描述和解释安全策略、安全凭证以及用于直接授权关键性安全操作的信任关系”。目前主要工作可分为三类: 信任推理模型、信任管理、信任计算模型等。信任管理主要研究如何在网络开放环境下集成信任和声誉, 以及信任的保存、更新和传递等, 存在交互的 Agent 之间计算交互信任, 在没有交互的 Agent 之间以推荐信任或声誉作为补充。

Agent 领域的信任评估机制的研究主要有两种方法, 一种方法是研究信任推理模型, 以认知逻辑、不精确推理等方法为工具, 将信任视为信念的一种形式进行推理与修正。另外一种方法是研究信任计算模型, 以概率论为工具, 基于交互历史修正信任关系, 如 REGRET 模型^[58]、TRAVOS 模型^[69-70]等。

信任推理机制的研究主要以模态逻辑和认知逻辑为工具,如 Sen 等人^[66]研究了开放环境条件下 Agent 如何相信其它 Agent,避免由于盲目相信导致被利用;Liau 等人^[40]以模态逻辑为工具研究信任在信息向信念转化过程中的影响,给出了信念(KD45)、信息获取(KD)和信任的逻辑关系;Schwartz 等人^[64]研究了知识和信念条件下的主观认知推理,主观信念表示为绝对相信、相当相信和不太相信等,知识则表示为确定的信念。

模糊论可以很自然地应用到信任关系,在精度要求不高的情况下,这是一种有效而简单的方法。Ramchurn 等人^[54]建立的一个多 Agent 交互的信任计算模型,可以指导 Agent 在模糊集合上评估过去的交互;王平^[130]则对 Ramchurn 的工作进行了扩展和改进。唐文等人^[119]通过考查主观信任的模糊性,将语言变量、模糊逻辑引入主观信任管理研究中,提出了信任的度量机制,运用模糊 IF-THEN 规则对人类信任推理的一般知识和经验进行了建模,提出了一种灵活直观,具有很强描述能力的形式化的信任推理机制。童向荣等人^[122-123]以模糊逻辑为工具对联盟信任进行推理,可以求得联盟盟友关系,提高盟友收益计算的实时性。

本文主要关注于信任计算模型,概率论工具可以利用交互历史计算得到信任关系。国际上流行的主要有以下几种信任计算模型: Beth 信任度评估模型^[7]、Jøsang 信任度评估模型^[35-37]、FIRE 模型^[30-31]、TRAVOS 模型^[69-70]、Abdul-Rahman 信任度评估模型^[54]。国内也有一些学者提出自己的计算模型,如窦文^[90]、李景涛^[108]、彭冬生^[116]等人提出的信任计算模型。

Beth 等人^[7]信任度评估模型引入了经验的概念来表述和度量信任关系,基于 Agent 完成任务一次任务的可能性在 $[0,1]$ 服从均匀分布这一假设。Jøsang 等人^[35-36]给出了一个由观察到的肯定事件数和否定事件数决定概率确定性密度函数 *pcdf*,并以此来计算实体产生某个事件概率的可信度。Huynh 等人^[31-32]提出了开放多 Agent 系统的集成信任和声誉模型 FIRE。集成了交互信任、基于角色的信任、观察声誉和认证声誉,它可以帮助 Agent 获得更好的收益,更好的适应环境的变化,陆萍萍^[113]对此工作进行了扩展,提出了 FIRE⁺模型。Teacy 等人用概率论研究信息不精确条件下 Agent 之间的信任关系,提出了 TRAVOS 模型^[69-70],Ashri 等人^[6]给出了基于信心和名誉的信任算法,Sabater 等人^[58]提出了 REGRET 模型。这些模型的共同不足之处在于: Agent 本身没有提供数据的义务,且不能保证提供数据是可靠的。另外,无法应对恶意欺诈,因而要求 Agent 是诚实的,而这本身是信任管理需要解决的问题。而 Yu and Singh 等人^[84]的声誉模型侧重于推荐信任度的计算,没有涉及直接经验的计算,因而不可取。

2.3.1 Beth信任度评估模型

Beth 信任度评估模型^[7]是较早为大家所接受的信任评估模型，与 Vassileva 等人的工作类似，他同样引入了经验来表述和度量信任关系，即主体 P 对客体 Q 的信任关系。其中经验被分为直接交互经验和间接推荐经验，并给出了由直接经验和第三方间接推荐经验所引出的信任计算公式。直接经验是 Agent 之间的直接交互，推荐经验是 Agent 之间的间接交互（Agent 不存在直接交互时，客体 Agent 与第三方 Agent 有直接交互关系），即主体对客体的经验既可以由交互直接获得，又可以通过第三方提供的信息间接获得，第三方提供的信息也可以再由其它与其有交互的第四方提供，以此类推。由此，该计算模型将信任分为直接信任和推荐信任。从以上分析可知，直接信任和推荐信任其实就是所熟知的信任和声誉。

基于 Q 完成一次任务的可能性在 [0, 1] 上服从均匀分布这一假设，直接信任与成功经验的关系表示为

$$v_s(p) = 1 - \alpha^p \quad (2.7)$$

其中， p 是 P 所获得的关于 Q 的成功经验数， α 则是对 Q 成功完成一次任务的可能性期望。

Beth 将成功经验与否定经验结合起来描述推荐信任度。推荐信任与经验的关系表示为

$$v_r(p, n) = \begin{cases} 1 - \alpha^{p-n} & p > n \\ 0 & \text{其它} \end{cases} \quad (2.8)$$

其中， p, n 分别是 P 所获得的关于 Q 的成功经验和失败经验数。

2.3.2 Jøsang信任度评估模型

Jøsang 等人提出了一种信任管理模型^[35-36]，所有的交互根据成功与失败的情况被划分为肯定事件(Positive Event)和否定事件(Negative Event)， r 和 s 分别表示观测到的实体所产生的肯定事件和否定事件数。Jøsang 基于 Beth 分布函数描述二项事件(Binary Event)后验概率的思想，给出了一个由观察到的肯定事件数和否定事件数决定概率确定性密度函数 pdf ，并以此来计算实体产生某个事件概率的可信度。设概率变量为 θ ，则 pdf 公式表述为

$$\phi(\theta|r,s) = \frac{\Gamma(r+s+2)}{\Gamma(r+1)\Gamma(s+1)} \theta^r (1-\theta)^s \quad 0 \leq \theta \leq 1, r \geq 0, s \geq 0 \quad (2.9)$$

其实,我们发现该 *pdf* 与 Beta 分布的概率密度是一致的。由此,后来有学者提出了基于 Beta 概率分布的信任计算模型 REGRET^[58]。主观信任度由三元组 $w = \{b, d, u\}$ 描述。该三元组满足

$$b + d + u = 1, \quad \{b, d, u\} \in [0,1]^3 \quad (2.10)$$

其中, b, d, u 分别描述对陈述的信任程度、不信任程度和不确定程度。Jøsang 使用公式 2.11 将 w 定义为观念空间中肯定事件数 r 和否定事件数 s 的函数

$$\begin{cases} b = \frac{r}{r+s+1} \\ d = \frac{s}{r+s+1} \\ u = \frac{1}{r+s+1} \end{cases} \quad (2.11)$$

在 Jøsang 信任管理模型中提供了一套主观逻辑算子,用于信任之间的运算,主要包括合并(Conjunction)、合意(Consensus)和推荐(Recommendation)。其中,合并用于不同信任内容的信任度综合计算;合意根据参与运算的信任之间的关系分为独立观念间的合意、依赖观念间的合意和部分依赖观念间的合意三类;推荐主要用于间接信任的推导。

与 Beth 模型不同,该模型没有明确区分直接信任和推荐信任,而提供了如合并、合意和推荐算子等用于信任的推导。

2.3.3 FIRE模型

Huynh 和 Jennings 等人提出的 FIRE 模型^[30-31]中,考虑了丰富的信任来源,主要分为四类:直接经验、推荐声誉、基于角色的规则以及第三方引用。Agent 之间采用议题的形式进行交互。评估 Agent 在自己的本地评估数据库分别保存与每个目标 Agent 交互的最近 1 次评估记录,通过查询本地评估数据库计算目标 Agent 的信任度。

该模型分为四种具体的信任和声誉,分别为

- (1) 直接经验:称为 Interaction Trust(IT),由直接交互得到;
- (2) 推荐声誉:称为 Witness Reputation (WR),愿意共享的 Agent 提供的 IT,

由于证人视角不同可能会有偏差；

(3) 基于角色的规则：称为 Role-based Trust (RT)，依据评价者与目标 Agent 之间的不同关系及基于领域的知识；

(4) 目标 Agent 提供的第三方参考：称为 Certified Reputation (CR)，由目标 Agent 提供的第三方作出的评价。由被评价者收集，主动向他人提供。

在该模型中有一个假设：各方为了共同的目标而愿意共享信息；在信息传输过程中诚信交换(WR、CR 来自第三方，暂不考虑撒谎与不精确信息)。

在此，用 Rating 表示由其合作者通过之间的交互给出的性能评价；Rating 的表示法：形如 $r = (a, b, c, i, v)$ 的元组。 a 是被服务者， b 是服务提供者，他们按术语 c (专题、质量、诚信等) 开展合作，在标识为 i 的交互中， a 记录 b 的 rating v ， $v \in [-1, 1]$ 。因有限存储及信息的时效性，因而只保存历史中最近的若干记录。

$$T_K(a, b, c) = \frac{\sum_{r_i \in R_K(a, b, c)} \omega_K(r_i) \cdot v_i}{\sum_{r_i \in R_K(a, b, c)} \omega_K(r_i)} \quad (2.12)$$

其中， $T_K(a, b, c)$ 是 Agent a 对 Agent b 在 c 问题上的 K 部分信任，而 K 可能是 I, W, R 或 C 。

$R_K(a, b, c)$ 是 K 部分的 rating 的集合， $\omega_K(r_i)$ 是 r_i 的相关程度或可靠程序（可能过时，可能不可靠，故不简单算术平均）， v_i 是 r_i 的值。

Rating 可靠性： ω_K 之和反映 rating 的可靠性，因 $\sum \omega_K$ 无限归到 $[0, 1]$ 。
 γ_K 用于调整 Rating 变化率的可靠性

$$\rho_{RK}(a, b, c) = 1 - e^{-\gamma_K \cdot (\sum_{r_i \in R_K(a, b, c)} \omega_K(r_i))} \quad (2.13)$$

偏移可靠性：与大家的差距越大，越不可靠，故用偏差反映

$$\rho_{DK}(a, b, c) = 1 - \frac{1}{2} \cdot \frac{\sum_{r_i \in R_K(a, b, c)} \omega_K(r_i) \cdot |v_i - T_K(a, b, c)|}{\sum_{r_i \in R_K(a, b, c)} \omega_K(r_i)} \quad (2.14)$$

则总的可靠性为

$$\rho_K(a, b, c) = \rho_{RK}(a, b, c) \cdot \rho_{DK}(a, b, c) \quad (2.15)$$

完整的信任值的计算

$$T(a, b, c) = \frac{\sum_{K \in \{I, R, W, C\}} w_K \cdot T_K(a, b, c)}{\sum_{K \in \{I, R, W, C\}} w_K} \quad (2.16)$$

其中, $w_k = W_k \cdot \rho_k(a, b, c)$ 。 W_I, W_R, W_W, W_C 是与 IT, RT, WR, CR 等各部分有关的系数, 该系数由最终用户确定, 反映各个部分在特定领域的重要性。

总信任计算的可靠性为

$$\rho_T(a, b, c) = \frac{\sum_{K \in \{I, R, W, C\}} w_K}{\sum_{K \in \{I, R, W, C\}} W_K} \quad (2.17)$$

2.3.4 TRAVOS模型

Teacy 等人^[69-70]用概率论研究不精确信息条件下 Agent 之间的信任关系, 提出 TRAVOS 模型来描述 Agent 之间的信任关系, 成功的历史交互赋值为 1, 不成功的交互赋值为 0, 应用 Beta 概率密度函数来模型化成功交互概率, 用 Beta 分布的数学期望表示为成功概率。TRAVOS 模型还给出信任值计算的信心, 如果该信心值低于预设的最小值, 则采用观察值。实验表明能够滤除一定的噪声和应对欺骗。

首先给出 Agent truster(a_r)和 Agent trustee(a_{te})之间交互结果的定义, 如式 2.18 所示。

$$O_{a_r, a_{te}} = \begin{cases} 1 & \text{if contract is fulfilled by } a_{te} \\ 0 & \text{otherwise} \end{cases} \quad (2.18)$$

在任何时刻 t , a_r 和 a_{te} 之间的交互历史结果记为

$$R_{a_r, a_{te}}^t = (m_{a_r, a_{te}}^t, n_{a_r, a_{te}}^t)$$

其中, $m_{a_r, a_{te}}^t$ 的值表示 a_r 和 a_{te} 之间的成功交互, $n_{a_r, a_{te}}^t$ 的值表示 a_r 和 a_{te} 之间的失败交互。

该模型用 $B_{a_r, a_{te}}$ 表示 a_{te} 在与 a_r 交互时, 完成任务的概率, 如式 2.19 所示。

$$B_{a_r, a_{te}} = p(O_{a_r, a_{te}} = 1), \quad B_{a_r, a_{te}} \in [0, 1] \quad (2.19)$$

该模型定义了给定交互历史 $O_{a_r, a_{te}}^{ht}$ 的情况下, 在时刻 t 时 $B_{a_r, a_{te}}$ 的期望值 $\tau_{a_r, a_{te}}$, 如式 2.20 所示。

$$\tau_{a_r, a_{te}} = E[B_{a_r, a_{te}} | O_{a_r, a_{te}}^{ht}] \quad (2.20)$$

为此, 给出 Beta 分布的 *pdf*, 如式 2.21 所示。

$$f(B_{a_r, a_{t_e}} | \alpha, \beta) = \frac{(B_{a_r, a_{t_e}})^{\alpha-1} (1 - B_{a_r, a_{t_e}})^{\beta-1}}{\int_0^1 U^{\alpha-1} (1-U)^{\beta-1} dU}, \quad \text{where } \alpha, \beta > 0 \quad (2.21)$$

其中, α, β 分别定义如式 2.22 所示。

$$\alpha = m_{a_r, a_{t_e}}^{t_e} + 1 \quad \text{and} \quad \beta = n_{a_r, a_{t_e}}^{t_e} + 1 \quad \text{where } t \text{ is the time of assessment} \quad (2.22)$$

最终, 得到信任计算的公式 2.23。

$$E[B_{a_r, a_{t_e}} | \alpha, \beta] = \frac{\alpha}{\alpha + \beta} \quad (2.23)$$

信任计算的置信度 (Confidence) 为

$$\gamma_{a_r, a_{t_e}} = \frac{\int_{a_r, a_{t_e} - \varepsilon}^{a_r, a_{t_e} + \varepsilon} (B_{a_r, a_{t_e}})^{\alpha-1} (1 - B_{a_r, a_{t_e}})^{\beta-1} dB_{a_r, a_{t_e}}}{\int_0^1 U^{\alpha-1} (1-U)^{\beta-1} dU} \quad (2.24)$$

2.4 协商

协商是 Agent 之间通过交互就某些问题达成相互有利的一致。如果协商的议题只有一个, 通常是价格, 那么就称为单议题协商; 如果协商的议题多于一个, 如价格、质量、付款方式等, 那么就称为多议题协商。多 Agent 协商一般应用在电子商务、电子贸易等方面, 也可以解决资源配置和化解危机等。单议题协商的研究已经比较成熟, 已有了大量的工作, 主要包括基于劝说的协商^[53]、基于辩论的协商^[38]、自适应协商^[46]和不完全信息协商^[19-21]。目前不完全信息条件下带有时间底限的多议题协商成为研究的热点。

Jennings 等人^[34]认为协商主要有三个方面研究内容: 协商协议、协商均衡目标、协商策略。协商协议给出协商的基本框架和 Agent 交互过程中共同遵守的规则; 协商均衡目标给出协商何时终止, 即协商达到均衡的目标; 协商策略是 Agent 为实现目标而采用的基本原则。协商采用的基本方法是基于对策论的方法。

Agent 协商一般有两种分类方法, 一种依据协商议题数目: 单议题协商和多议题协商, 另一种依据信息状态: 完全信息协商和不完全信息协商, 其中不完全信息多议题协商是较为复杂的并成为目前的研究热点。

在不完全信息条件下, 目前有几种 Agent 多议题协商方法^[19-21]: 一是同时 (并发) (simultaneous procedure) 的多议题单独协商, 该方法研究较少。一是顺序

(sequential procedure)的按照协商议程的多议题协商。王黎明等人采用相似的分阶段方法研究多议题协商^[128],但都不能保证 Pareto 最优。Fatima 等人^[21]总结了多议题协商方法,分别给出顺序协商、一揽子交易(package deal procedure)多议题协商和同时协商的 Pareto 最优解存在性、计算复杂性、达成一致的时间以及解的唯一性等特性。

目前研究最多的是—揽子交易多议题协商,将多个议题捆绑作为一个完整的包同时讨论,或者同意所有议题或者拒绝所有议题,并允许在不同议题之间进行折衷。一个核心问题是哪种方法能够取得最优的协商效果(Pareto 最优),Fatima 等人^[21]指出只有—揽子交易多议题协商才能保证 Pareto 最优解。

2.4.1 协商的基本模型

协商所用的基本理论是对策论,Fatima 等人^[19-21]在这方面做了大量工作,提出了一种基于对策论的单议题协商协议。

Agent s 在时刻 t 对出价 $p_{b \rightarrow s}^t$ 所做出的反应动作

$$A^s(t, p_{b \rightarrow s}^t) = \begin{cases} \text{Quit} & \text{if } t > T^s, \\ \text{Accept} & \text{if } U^s(p_{b \rightarrow s}^t) \geq U^s(p_{s \rightarrow b}^t), \\ \text{Offer } p_{s \rightarrow b}^t \text{ at } t' & \text{otherwise.} \end{cases} \quad (2.25)$$

其中, $p_{b \rightarrow s}^t$ 表示在时刻 t , Agent b 向 s 的出价, T^s 表示 Agent s 的时间底限, U^s 是 Agent s 的效用函数。

Agent a 的效用函数为

$$U^a(p, t) = U^a(p)U_i^a(t) \quad (2.26)$$

其中, p 是在时刻 t 的协商价格。 $U^a(p)$ 是价格效用函数, 表示为

$$U^a(p) = \begin{cases} RP^b - p & \text{for the buyer} \\ p - RP^s & \text{for the seller} \end{cases} \quad (2.27)$$

其中, RP^b 是买方的保留出价, RP^s 是卖方的保留出价。 $U^a(t)$ 是时间效用函数, 表示为

$$U_i^a(t) = (\delta^a)^t \quad (2.28)$$

其中, δ^a 表示时间折扣率, $\delta^a > 1$ 表示 Agent 是耐心型的, 随着时间的增加获得效用; 相反, $\delta^a < 1$ 表示 Agent 是非耐心型的, 随着时间的增加损失效用。

Fatima 等人^[20]还提出了多议题协商协议框架, 用对策论研究了完全信息和有时间底限的多议题协商, 提出了基于日程的顺序协商模型并给出了 Agent 信息状态和动作定义, 基于对策论和 Nash 均衡给出了带有时间折扣率的综合效用函数和 Agent 出价策略、接受出价策略。使用外因作用将议题分为 k 个阶段, 排序采用内因的方式, 分析表明, 尽管是不完全信息但依然存在均衡解。

Fatima 等人给出了多议题协商的基本模型^[20]。 b 表示买方, s 表示卖方, $[IP^a, RP^a]$ 表示 Agent a 的初始出价和保留出价, 其中 $a \in \{b, s\}$, \hat{a} 表示 a 的对手。 I^a 表示 Agent a 的信息状态, Agent a 不断的在时间序列 $T = \{0, 1, \dots\}$ 出价, 每个 Agent 都有一个时间底限, T^a 表示 Agent a 的时间底限, $T^a \in T$, 设 $P_{b \rightarrow s}^t$ 表示 Agent b 在时刻 t 向 Agent s 的出价, 考虑有两个议题(例如 A 和 B)的协商, Agent 信息状态定义为

$$I^a = \{RP_A^a, RP_B^a, T^a, \delta^a, U^a, RP_A^{\hat{a}}, RP_B^{\hat{a}}\} \quad (2.29)$$

$$I^{\hat{a}} = \{RP_A^{\hat{a}}, RP_B^{\hat{a}}, T^{\hat{a}}, \delta^{\hat{a}}, U^{\hat{a}}, RP_A^a, RP_B^a\} \quad (2.30)$$

协商过程中 Agent s 在时刻 t 的动作 $A^s(t, A_{b \rightarrow s}^t, B_{b \rightarrow s}^t)$ 为

$$A^s(t, A_{b \rightarrow s}^t, B_{b \rightarrow s}^t) = \begin{cases} \text{Quit} & \text{if } t > T^s \\ \text{Accept } A_{b \rightarrow s}^t & \text{if } A_{b \rightarrow s}^t \geq S_{oA}^s(t) \\ \text{Accept } B_{b \rightarrow s}^t & \text{if } B_{b \rightarrow s}^t \geq S_{oB}^s(t) \\ \text{offer } S_{oA}^s(t') \text{ at } t' & \text{if } A_{b \rightarrow s}^t \text{ not accepted} \\ \text{offer } S_{oB}^s(t') \text{ at } t' & \text{if } B_{b \rightarrow s}^t \text{ not accepted} \end{cases} \quad (2.31)$$

其中, $A_{b \rightarrow s}^t$ 表示在时刻 t , Agent b 向 Agent s 提出的关于议题 A 的出价, $B_{b \rightarrow s}^t$ 则是关于议题 B 的出价; $S_{oA}^s(t)$ 表示由 Agent s 的最优策略得到的关于议题 A 在时刻 t 的出价, 其余类似。

Fatima 等人给出的 Agent 效用函数为

$$U^a(pA, PB, t) = \begin{cases} (RP_A^b - pA)(\delta_A^b)^t + (RP_B^b - pB)(\delta_B^b)^t & \text{for } b \\ (pA - RP_A^s)(\delta_A^s)^t + (pB - RP_B^s)(\delta_B^s)^t & \text{for } s \end{cases} \quad (2.32)$$

该效用函数考虑了多个议题的组合效用, 而且考虑时间贴现率。但是该效用函数将两个议题同等对待, 效用的加权是相同的。这显然是不符合实际情况的, 对每

个 Agent 来说,对两个议题的偏好程度是有所区别的,称为偏序关系。自然的,两个议题的总效用也就应该是两个议题效用的加权求和。

Fatima 等人^[21]指出只有一揽子交易协商才能保证取得 Pareto 最优的多议题协商结果,其采用的基本方法是多议题之间的折衷。

国内学者王立春等人^[127]利用协商-协商过程-协商线程的概念建立了一个多边多问题协商模型 MMN。郭庆等人^[92]给出了一个多议题整合效用评估机制,在保证协商参与者整体协商效用的前提下动态放宽某个议题的保留值,促使协商双方避免协商僵局,快速达成一致的协定。

很多工作将协商研究应用于实际的案例中,取得了很好的效果。如 Ramchurn 等人^[55]研究了反复交互情形下,用 Rewards 实现的劝说式协商机制,实验表明这可以提高囚犯两难问题解的效用。Saha 等人^[59]研究了多个不可分割资源的协商,给出了一种有效的协议。Chevaleyre 等人^[12]研究了无嫉妒的货物配置的分布式协商方法。Kraus 等人^[39]研究通过自动协商方法化解危机,实验显示加拿大和西班牙之间的渔业冲突问题协商中,Agent 可以至少和(在某种程度上好于)西班牙一方的人类谈判者一样好。

2.4.2 协商的策略与学习

多议题协商策略对于协商的结果和协商的难度有着直接的影响,合理的策略选择是协商成功和个体效用最大化的关键。对协商策略的基本要求是保证 Agent 是理性的,至少是 Pareto 最优,而且 Agent 可以从多次协商中学习较优的协商策略,即 Agent 具有学习性。但是在时间限制和不完全信息条件下最优协商策略的取得需要借助于智能。因此很多工作集中于 Agent 协商中的学习和优化^[13,22,42-43,46]。在不完全信息和时间底限条件下,Agent 尽可能地从对手的历史出价中学习,这不仅可以有效地采取最优出价还可以在协商取得优势地位并得到较好的结果。其中贝叶斯再励学习方法是一个比较合适的选择,被对手接受的出价作为正反馈,被对手拒绝的出价作为负反馈。而遗传算法能够发现有效的协商策略,将 Agent 视作染色体,协商参数作为基因,通过进化这些 Agent,协商策略的优劣一目了然,从而得到最优的协商策略。

学习是 Agent 协商过程中的重要问题。Agent 可以从对手以往的出价中学习其议题偏好,以尽可能多的掌握对手的信息。这样不仅能够采取合理的协商策略,而且还可以在多议题的折衷处理中占得主动,最大化个体利益。Coehoorn 等人^[13]提出了一种统计学的核密度估计方法来学习对手的偏好,基本思想是使用了相似

度函数来表示和计算议题之间的关系。设 $x_j, y_j \in D$ 是议题 j 的两个值, h 函数

$$D \rightarrow [0,1]$$

将其映射为 $[0,1]$ 中的某个值, 使用如下公式计算出对手关于各个议题的相似度。

$$sim_j(x_j, y_j) = 1 - |h(x_j) - h(y_j)| \quad (2.33)$$

$$sim = \sum_{j \in J} \omega_j^b sim_j(x_j, y_j) \quad (2.34)$$

其中, ω_j^b 是关于议题 j 的对手的权重。

Luo 等人^[43]提出一种基于事例学习算法的对手偏好学习方法, 首先将某些议题的属性变坏, 然后让对手调整其它属性以达到平衡, 最后达到让对手分配各个议题满意度的目的。Luo 等人^[42]设计了一种多议题协商的模糊限制模型, 针对买方的需求一般都是模糊的限制条件这一事实, 采用预定义的模糊限制来表示协商多议题可能值, 并指出应该做出多大妥协。与单点限制协商相比, 该协商模型能更大程度的覆盖协商可能交易空间, 基本原理是采用了扩充协商 Agent 模型, 卖方 Agent 模型

$$(G, A, P, \varpi, \Theta)$$

其中, G 是商品的模型, A 是动作集, P 是行为协议, ϖ 是通信端口。

$$\Theta = (\text{constraint-set}, \text{solution}, \text{last-solution}, \text{previous-solution})$$

是工作存储器, 其中, Constraint-set 存储了卖方 Agent 回价的模糊限制条件, solution 是卖方 Agent 根据模糊限制条件 constraint-set 能够发现的当前解决方案; 买方 Agent 同样有工作存储器。

$$\Theta = (\text{submitted-constraints}, \text{cut-level})$$

其中, submitted-constraints 存储了买方 Agent 出价的模糊限制, cut-level 则表明了模糊限制的放松程度。

2.4.3 协商策略的改进

Faratin 等人^[22]提出用相似度方法处理多议题之间的折衷, 其给出的算法是满足自身效用不变前提下寻找使对手效用最优的出价; Fatima 等人^[21]则用了与之相似的策略, 其算法是满足对手效用不变前提下寻找使自身效用最优化的出价, 这

两种方法都是单目标约束满足问题。Lin 等人^[41]设计的一种软件 Agent 则可以使自身效用和社会福利（双方效用之和）都扩大化。

王黎明等人^[128]提出了一个以理性 Agent 为基础的基于多阶段的多问题协商框架。王黎明等人^[129]又提出了基于信任和 K 臂赌博机问题选择多问题协商对象的协商算法。杨佩等人^[131]提出一种劝说式多 Agent 多议题协商方法，考虑对手对协商议题的偏好，并根据一种基于辩论的信念修正方法调整自身的偏好，这样就能够使协商 Agent 对变化的协商环境具备适应性。

高坚等人^[91]给出了一个 Agent 多议题协商模型，提出了一种双边多议题协商的加速混沌进化算法（ACEA），既克服了进化计算过早收敛到局部 Nash 平衡点的缺点，又解决了多议题协商时繁杂的计算和引入混沌后带来的收敛速度慢的问题。童向荣等人^[126]提出了不完全信息条件下基于案例和对策论的 Agent 多议题 Pareto 最优协商模型，算法的计算复杂度为多项式级，且一般情况下低于 Fatima 工作的计算复杂度。

2.4.4 拍卖

拍卖作为协商的一种重要形式引起了广大学者的关注。MAS 试图引入拍卖机制来解决任务分配、资源配置等问题，来优化系统性能。近年来，拍卖已成为人工智能和分布式系统中的研究热点。拍卖的概念被扩充，拍卖的物品已不限于艺术品、收藏品、房地产等，在电子商务、网络带宽分配、电力调度、网上零售、电子图书馆等大型分布式系统中也引入拍卖机制。

一个卖方和多个买方的交易构成正拍卖。一个买方和多个卖方的交易构成反拍卖。多个买方和多个卖方的交易就构成双边拍卖。反拍卖与正拍卖是对称的，因此一般只需讨论正拍卖。按叫价变化的方向，拍卖可分为递增叫价拍卖和递减叫价拍卖。按照叫价的方式，拍卖可分为暗标叫价拍卖和明标叫价拍卖。

效用是拍卖中的首要问题。对于一个拍卖结果来说，未能成交的 Agent 效用为 0，若物品以价格 p 成交，则买方 b 的效用为

$$V_b(g) - p$$

卖方 s 的效用为

$$p - V_s(g)$$

根据效用理性的假设，当 $V_b(g) < p$ 时不应成交。当 $V_b(g) > p$ 时，成交后的

总共效用为

$$(V_b(g) - p) + (p - V_s(g)) = V_b(g) - V_s(g)$$

与成交价格无关，仅与买方的评价价值 $V_b(g)$ 有关。衡量一个拍卖结果效用的主要指标有：总效用最大、Pareto 最优和卖方效用。

策略是拍卖中第二个重要的问题。衡量拍卖方法中 Agent 策略的主要指标有：占优策略存在性，即是否存在一种无论其它 Agent 如何行动都能保证最优的策略，若不存在占优策略，则希望存在 Bayes 最优策略，即存在概率意义下的最优策略；策略的时间复杂度；满足激励相容性，即将对物品的真实估价作为叫价是占优策略；独立理性。从策略角度，理想的拍卖方法应对 Agent 理性和计算能力的要求尽可能低，最好满足激励相容性。

常见的单个物品拍卖方法有英国式拍卖、荷兰式拍卖和第 k 高价拍卖。将所有待拍卖的物品同时进行拍卖就是组合拍卖。单物品单属性拍卖是最简单的拍卖方式，目前的研究已近完善，现在研究的主要是组合拍卖。

组合拍卖模型

$$M = \langle B, S, G, V, A \rangle$$

其中， B 为买方集合，包含 n 个买方， $B = \{1, \dots, n\}$ ， S 为拍卖中唯一的卖方。

$$G = \{x_1, \dots, x_m\}$$

为物品集合， $C \subseteq G$ 表示物品组合， G 中的单个物品也用 x, y, z 表示。

$$V = (V_1, \dots, V_n)$$

其中， $V_i: G \rightarrow R$ 为买方 i 的物品组合评价函数（ R 为实数集合），即买方 i 对物品组合 C 的评价价值为 $V_i(C)$ 。评价函数 V_i 应满足

$$V_i(\phi) = 0 \quad (\text{无物品则无价值})$$

$$\forall C_1 \subseteq C_2 \subseteq G: V_i(C_2) \geq V_i(C_1) \quad (\text{物品可无代价地自由丢弃})$$

物品分配方案表示为

$$A = (A_1, A_2, \dots, A_n)$$

其中， A_1, A_2, \dots, A_n 为 G 的一个划分，即

$$\bigcup_{i \in B} A_i = G, \text{ 且 } i \neq j \text{ 时 } A_i \cap A_j = \phi$$

定义物品分配方案 A 的效用

$$U(A) = \sum_{i=1}^n V_i(A_i)$$

最优分配 A^* , 有

$$U(A^*) = \max_A (U(A))$$

最简单的组合拍卖是一种暗标的最高叫价拍卖方法, 采用胜者决定算法。拍卖初始每个买方对所有物品组合进行暗标叫价, 当收集到所有买方的叫价后, 卖方利用胜者决定算法, 确定拍卖的结果。Sandholm 等人提出了一种最优物品分配算法 CABOB。实验说明 CABOB 在绝大多数情况下都明显优于 CPLEX 7.0 算法。后来, Sandholm 等人又提出了改进的胜者决定算法 BOB^[63]。为减小组合拍卖中的通信量, Sandholm 等人提出了一种部分报价方法, 利用自由处置假设得出的组合叫价间的关系, 买方先将少部分叫价传给卖方, 根据计算的需要, 再向买方询问必要的叫价^[62]。Candale 等人^[11]研究多维投标优化机制, 以在同时拍卖中得到多维捆绑的最优投标, 给定一个初始投标向量, 该机制能系统地对每一维进行优化。

Yokoo 等人提出当一个买方利用多个身份参与组合拍卖时有可能获得更多的效用, 卖方的利益则无法保证达到最大, 即通常的组合拍卖不具有抗假名叫价的性质^[81]。Yokoo 等人提出了一种将叫价分层的拍卖方法, 使得使用假名叫价的买方不能获得不正当利益。Yokoo 等人又将上一方法扩展到多数量组合拍卖中^[82], 并讨论了利用上一方法时如何分层^[83]。

能够使买卖双方在物品的各种属性上进行协商的拍卖方法称为多属性拍卖, 买方的物品属性估价值函数 (简称估价函数) 和卖方的物品属性成本函数 (简称成本函数) 的多样性使这一类问题复杂化。

多属性拍卖模型

$$M = \langle A, B, S, V, C, Res \rangle$$

其中, A 为属性空间。

$$A = A_1 \times \dots \times A_m$$

拍卖的物品包含 m 个属性 a_1, \dots, a_m , 取值范围分别为 A_1, \dots, A_m 。令 $a = (a_1, \dots, a_m)$ 为物品的属性向量, $a \in A$ 。 B 为拍卖中唯一的买方, B 需要购买一个物品。 S 为卖方集合,

$$S = \{1, \dots, n\}$$

每个卖方可以提供不同属性的物品。买方 B 的属性评价函数表示为

$$V: A \rightarrow R$$

其中, R 为实数集合, 即买方 B 对属性为 a 的物品评价值为 $V(a) \in R$ 。

$$C = \{C_1, C_2, \dots, C_n\}$$

其中, C_i 为卖方 i 的成本函数, 即卖方 i 对属性为 a 的物品的成本为 $C_i(a) \in R$ 。Res 为成交方案, 表示为

$$\text{Res} = (P, a)$$

其中, 成交价格 $P \in R$, 成交属性向量 $a \in A$ 。此时买方 B 的收益为

$$U = V(a) - P$$

卖方 S_i 的收益为

$$U_i = P - C_i(a)$$

金滓等人^[103]针对边际效用递减的组合拍卖提出了物品组合的 1-UNT 条件, 给出了一种基于 1-UNT 检查的近似算法, 证明了 1-UNT 算法的解的效用率不低于 0.5, 并通过实验说明 1-UNT 算法的效率高于贪心算法, 而运算时间比贪心算法稍长, 远低于 CABOB 算法。

金滓等人^[104,106]还提出了买方叫价活性级, 给出了一种多活性级的递增叫价组合拍卖方法 MALB, MALB 中采用的策略兼顾了时间复杂度和效用。选取适当的参数, 可使 MALB 总效用率、总效用最大率、卖方效用率和拍卖轮数等指标都优于 iBundle 拍卖方法。

金滓等人^[105]提出了一种暗标叫价多属性拍卖方法 MAV 和一种递增叫价多属性拍卖方法 MAE。MAV 和 MAE 分别是 Vickrey 拍卖和递增叫价拍卖从单属性到多属性的推广, MAV 和 MAE 中的买方和卖方都存在占优策略或 Bayes 最优策略, 在 Bayes 均衡下, 买卖双方总效用可达到或接近最大。MAV 可保证拍卖效率, 而 MAE 拍卖过程透明、公正, 卖方策略直观。

在 IJCAI'07 会议中, 拍卖又一次成为 Agent 合作求解的研究热点, 而且关于拍卖的安全性和反欺骗机制的研究越来越成为关注的重点。Brandt 等人^[10]研究了密封投标拍卖中的恶意投标, 得出了恶意 Agent 在第 1 价格和第 2 价格密封投标拍卖中的对称贝叶斯 Nash 均衡。第 1 价格投标中, 恶意 Agent 变得更诚实; 而在第二价格投标中, 均衡策略并不依赖于参与者的数量。Holland 等人^[30]介绍了期望的单调性概念, 给出了关于诚实机制的不可能结果, 该结果可以扩展到组合拍卖,

也显示该结果对最大化期望社会福利有好的作用。Gerding 等人^[27]考虑两个有相似货物的卖方在并发在线拍卖中的竞争问题，显示两个具有不对称产品成本的卖方存在完美的 Nash 均衡，双方都将保留价格设为产品成本之上。而且，一个卖方使用“托”，能更有效的提高效用。

2.5 小结

本章回顾和综述了多 Agent 系统的合作求解研究进展，介绍了目前联盟、信任、协商等研究的基本概念、基本方法和改进方法等。合作求解是多 Agent 系统的核心内容之一，目前也是多 Agent 系统研究的热点和难点，在国内外杂志和国际会议上有大量的文献出现，引起了广大研究者的兴趣。当前这些工作仍然有一些值得改进的地方，本论文将就存在的问题展开研究，改进目前工作。

第三章 一种 Agent 动态交互信任预测与异常行为检测模型

在多 Agent 系统理论中,信任计算是一个有意义的研究方向。信任是“agency”和“on behalf of”的基础,也是开放网络环境下多 Agent 交互的基础,对联盟、团队、组织、合作、协商起关键作用。目前有大量的工作致力于 Agent 信任的研究,也有很多信任评估模型为大家所接受。然而目前的 Agent 信任研究一般以平均交互成功率来计算,很少考虑信任的动态变化。事实上,Agent 信任与上下文、时间、背景等有关,是动态变化的,其变化率一般来说不宜过大。如果变化率过大则有可能是异常行为,这在股市的异常交易行为检测等方面有重要作用。在这一章中我们主要介绍 Agent 的动态交互信任预测与异常行为检测。

本章以概率论为工具,按时间将交互历史分段,给出 Agent 动态交互信任计算模型 CMAIT;结合信任的变化率,给出信任计算的置信度和异常行为检测机制。实验结果显示该计算模型的预测误差是 TRAVOS 模型的 0.5 倍,计算量也较少。该模型既可用于对手历史行为的异常检测,防止被欺骗,又可用于对手未来行为的预测。改进了 Jennings 等人关于 Agent 信任的工作。

3.1 引言

信任一般为信任者对被信任者能力和可靠性的一种判断性信念,是对被信任者完成任务或承诺的期望。信任关系是 Agent 选择合作伙伴的重要依据。信任包括很多内涵,如从历史、实力、环境和条件的角度,从主观偏好的角度等都能产生信任,且信任关系随时间、上下文、环境等诸多因素动态变化。目前主要有交互信任和声誉(也称为推荐信任),本章主要研究由交互历史产生的交互信任。

本章使用目前普遍采用的概率论方法。目前国际上有很多主流的信任评估模型,如 Beth 等人^[7]提出的信任度评估模型采用均匀分布、Jøsang 等人^[35-36]提出的采用二项分布的基于主观逻辑的信任管理模型、Teacy 等人^[69-70]提出的 TRAVOS 模型采用 Beta 分布。然而,此前的信任计算模型都没有将交互历史分段计算,而是取平均成功率,这导致交互信任的表示过于简单,预测精度较低,无法合理选择伙伴。另外这些模型处理噪声信息和避免被欺骗的能力较弱。

针对以上问题,按时间分段研究 Agent 交互信任。给定一系列交互历史,对其进行分析,得到更多有助于 Agent 选择合作伙伴的信息,并提出 Agent 交互信任

计算模型 CMAIT (Computational Model of Agent Interaction Trust)。实验结果显示 CMAIT 的计算误差比 TRAVOS 模型小, 计算量也比 TRAVOS 模型小; 且考虑了信任变化的趋势, 为未来交互提供预测; 该模型给出的置信度能滤除噪声、防止恶意的欺骗等。深化了之前 Jennings 等人关于 Agent 信任的研究工作。

多 Agent 系统中的信任研究可以应用于网上电子商务中, 也可以用于网上银行业务客户的信任评估、网格服务的合作伙伴选择等。实验选择 Agent 在淘宝网购买手机及附件过程中如何选择手机商家, 交互次数较多。实验结果表明, 该模型能够合理地计算 Agent 信任, 反映实际情况, 并通过对比实验说明该模型的计算误差是 TRAVOS 模型的 0.5 倍, 且能有效检测对手的异常行为, 防止被欺骗。

3.2 相关工作分析

本章主要关注于信任计算模型, 国际上流行的主要有以下几种信任计算模型: Beth 信任度评估模型^[7]、Jøsang 信任度评估模型^[35-37]、FIRE 模型^[30-31]、TRAVOS 模型^[69-70]。国内也有一些学者提出自己的计算模型如窦文^[90]、李景涛^[108]、彭冬生^[116]等人提出的模型。表 3.1 给出目前已有的几种典型的直接交互信任计算模型, 通过对比分析, 我们将选择一种信任评估模型作为我们动态信任的基础模型。

表 3.1 几种典型的交互信任计算模型 (r 是成功交互次数, s 是失败交互次数)

Table 3.1 Some typical interaction trust computational model

模型	Beth	Jøsang	TRAVOS	FIRE	窦文	彭冬生	李景涛
公式	$v(r) = 1 - \alpha^r$	$\frac{r+1}{r+s+2}$	同 Jøsang	无	$\frac{r}{r+s}$	$\frac{r}{r+sN_{pnsh}}$	$\frac{\max(r-s, 0)}{\sum \max(r-s, 0)}$
参数	α 是伙伴完成任务可能性的期望	二项分布	服从 Beta 分布	合成		N_{pnsh} 是惩罚系数	

表 3.2 Agent i 与卖家 wfs19830419, huagong1979ka 和 kennethlee 的交互历史纪录 (取自淘宝网 2008.03.28)

Table 3.2 Seller Agent wfs19830419, huagong1979ka and kennethlee interaction history (derived from www.taobao.com on 28.03.2008)

卖家信用度 (174) — wfs19830419	信任模型的计算值
---------------------------	----------

交互历史	最近1周	最近1个月	最近6个月	6个月前	总计	李景涛(174/ σ)
好评	38	122	144	30	174	窦文 $\alpha=1(1)$
中评	0	0	0	0	0	彭冬生 $N_{pnh}=1(1)$
差评	0	0	0	0	0	TRAVOS(0.9943)
总计	38	122	144	30	174	Jøsang(0.9943)

卖家信用度 (196529) —kennethlee						信任模型的计算值
交互历史	最近1周	最近1个月	最近6个月	6个月前	总计	李景涛(196691/ σ)
好评	7927	26568	104014	92677	196691	窦文 $\alpha=1(0.992892)$
中评	7	55	628	618	1246	彭冬生 $N_{pnh}=1(0.99289)$
差评	0	13	105	57	162	TRAVOS(0.992882)
总计	7934	26636	104747	93352	198099	Jøsang(0.992882)

卖家信用度 (2636) —huagong1979ka						信任模型的计算值
交互历史	最近1周	最近1个月	最近6个月	6个月前	总计	李景涛(2636/ σ)
好评	23	71	2636	0	2636	窦文 $\alpha=1(1)$
中评	0	0	0	0	0	彭冬生 $N_{pnh}=1(1)$
差评	0	0	0	0	0	TRAVOS(0.99962)
总计	23	71	2636	0	2636	Jøsang(0.99962)

例 3.1 淘宝网上几个卖家与 Agent 的交互历史纪录见表 3.2，试用此例分析各个信任计算模型。

计算模型应该能够区分各种情况。如例 3.1, 考察各种交互历史序列, 对于三个商家 wfs19830419, kennethlee, huagong1979ka, 计算模型应该给出不同的信任值。然而, 我们可以看到对于 wfs19830419 和 huagong1979ka, 窦文^[90]和彭冬生^[116]的计算模型都无法给出区别, 这是不理想的, 李景涛^[108]的计算模型虽然能给出区别, 但是这种模型用到了整个系统的交互历史, 需要完全信息, 这也不容易实现。只有 TRAVOS 模型能够给出各种区别, 因而本章的计算模型借用 TRAVOS 模型的计算方法。

由于大量交互的不确定性, 因而概率论工具较成功地应用于信任计算中, 本章不考虑推荐信任, 只考虑基于概率论的交互信任计算, 研究交互信任的动态性。

3.3 问题描述

已知 MAS 系统中有 n 个 Agent 与 Agent b 存在直接交互历史, Agent b 的信息状态 IS , 形成 Agent 交互关系如图 3.1 所示。小圆圈表示 Agent, 小圆圈之间的连线表示 Agent 之间的交互, 计算 Agent b 对其它 Agent 的信任, 以供 Agent b 选择伙伴。

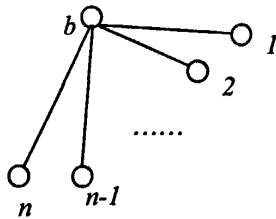


图3.1 Agent交互示意图
Figure 3.1 Agent interactions

定义 3.1 Agent 交互信任

Agent i 和 Agent j 之间在时刻 t 由交互历史结果形成的信任, 称为 Agent 交互信任 $it'_{i,j}$ 。

定义3.2 Agent b 的信息状态

$$IS = \langle M, T, K \rangle$$

其中, 向量

$$\mathbf{T} = [T_1, T_2, \dots, T_m]$$

是时间序列, Agent之间的交互历史序列可以划分为 m 个阶段, $T_i (i=1, 2, \dots, m)$ 表示第 i 个阶段的周期。向量

$$\mathbf{K} = [\tau_I, \tau_D]$$

分别表示Agent计算信息量置信度的容忍值, 计算偏离置信度的容忍值, 可用来检测噪声和恶意欺骗。矩阵

$$\mathbf{M} = \langle \mathbf{M}_s, \mathbf{M}_f \rangle, \mathbf{M}_s = \begin{bmatrix} s_1^1 & s_1^2 & \dots & s_1^m \\ s_2^1 & s_2^2 & \dots & s_2^m \\ \vdots & \vdots & \ddots & \vdots \\ s_n^1 & s_n^2 & \dots & s_n^m \end{bmatrix}, \mathbf{M}_f = \begin{bmatrix} f_1^1 & f_1^2 & \dots & f_1^m \\ f_2^1 & f_2^2 & \dots & f_2^m \\ \vdots & \vdots & \ddots & \vdots \\ f_n^1 & f_n^2 & \dots & f_n^m \end{bmatrix}$$

是 Agent b 存储的与其它 Agent 的按时间顺序排列的 m 个阶段的交互历史。 \mathbf{M}_s 是各个阶段的成功交互次数, \mathbf{M}_f 是各个阶段的失败或平局交互次数。成功交互即 Agent b 能够获得收益; 失败交互即 Agent b 没有获得收益或受到了损失。行向量是 Agent b 的各个阶段交互历史, 列向量是 Agent b 在每个阶段的与各个 Agent 交互历史。在每个阶段内 Agent b 对于交互历史的排序没有偏好, 即可以认为是时间等效的。矩阵中的元素 s_j^k 表示 Agent b 与 Agent j 在第 k 个阶段的成功交互次数, f_j^k 表示 Agent b 与 Agent j 在第 k 个阶段的失败交互次数。

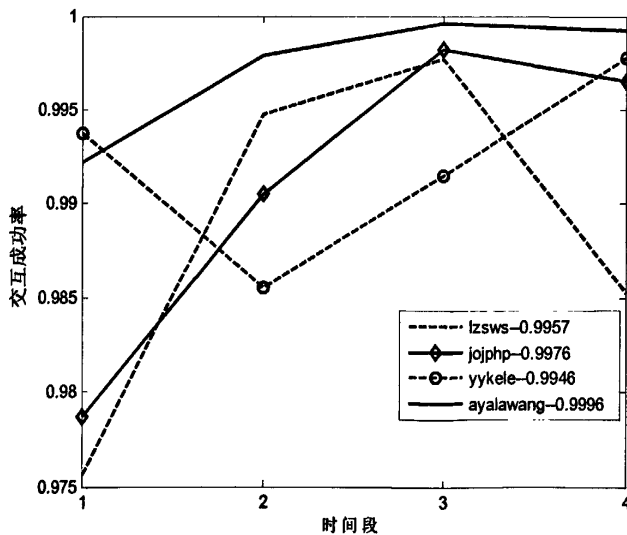


图 3.2 Agent 交互信任分段分布图

Figure 3.2 Agent interaction Trust divided by time

根据时间将交互历史分段, 我们得到例 3.1 的分段信任值, 如图 3.2 所示。由

图可知，信任在不同阶段有不同值，是不断变化的，因此计算动态信任，并考虑信任变化率是有必要的。

3.4 Agent 交互信任计算

3.4.1 计算原理

信任总是随时间动态变化的，在不同的时刻有不同的交互成功率。因为过长的历史把更重要的当前交互成功率和趋势给稀释了，容易造成判断上的错误。因此，采用平均交互成功率的方式来计算信任并不可靠。

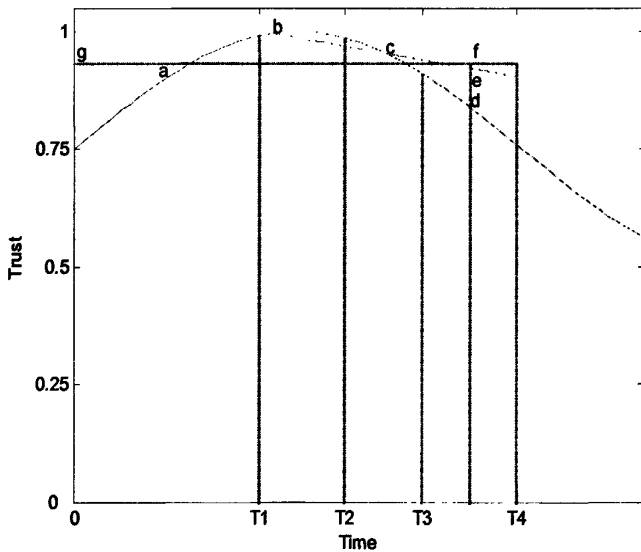


图 3.3 Agent 交互信任分段计算示意图

Figure 3.3 Agent interaction Trust computation

本章将交互历史序列分段计算，每个阶段的交互成功率采用 TRAVOS 模型的基本方法，然后考虑各个阶段的动态变化率，预测下一个阶段的信任值。如图 3.3 所示，Agent 之间的交互可以分为 4 个阶段，已知前 3 个阶段的交互历史数据，可以求在第 4 个阶段的交互信任。线段 gf 的纵坐标表示前 3 个阶段的历史交互平均成功率， a, b, c, d 点的纵坐标分别表示在第 1、2、3 和 4 个阶段的实际交互成功率。

由图可知第 4 个阶段的真值应该是 d 点的纵坐标的值。如果由历史交互平均成功率来求，得到 f 点的纵坐标的值，误差为 df ；如果采用 CMAIT 方法来求，得到 e 点的纵坐标的值，误差为 de ，显然 $de < df$ 。可见，CMAIT 方法较好的考虑了信任变化的趋势，能较好地预测未来交互，与历史平均交互成功率相比更适用。

3.4.2 动态交互信任计算

由 TRAVOS 模型^[69-70], 在每一个阶段 k Agent i 对 Agent j 的信任值为

$$E_{i,j}^k = \frac{s_j^k + 1}{s_j^k + f_j^k + 2} \quad (3.1)$$

其中, $E_{i,j}^k \in [0, 1]$ 是由 Beta 概率分布确定的 Agent i 和 Agent j 之间的 TRAVOS 模型的交互信任。

第 $k-1$ 阶段到第 k 阶段的变化率为

$$d(k-1, k) = \frac{E_{i,j}^k - E_{i,j}^{k-1}}{T_k} \quad (3.2)$$

其中, $d(k-1, k)$ 为从第 $k-1$ 阶段到第 k 阶段的信任变化率, $E_{i,j}^k - E_{i,j}^{k-1}$ 是从第 $k-1$ 阶段到第 k 阶段的信任值之差, $\frac{1}{T_k}$ 是第 k 阶段的周期的倒数, 是该周期的权重。

$\frac{1}{T_k}$ 是第 k 阶段的周期的倒数, 是该周期的权重。

t 时刻的信任 $it'_{i,j}$ 为

$$it'_{i,j} = \min \left\{ 1, \max \left\{ E_{i,j}^k + \frac{E_{i,j}^k - E_{i,j}^{k-1}}{T_k} (t - t_k), 0 \right\} \right\} \quad (3.3)$$

其中, t_k 为第 k 阶段的结束时的时间, 则 $T_k = t_k - t_{k-1}$ 。

函数 \max 和函数 \min 是为了让 $it'_{i,j}$ 有界, 函数 \max 为了避免 $it'_{i,j}$ 出现负值的情况, 函数 \min 为了避免 $it'_{i,j}$ 出现大于 1 的情况。由此, $it'_{i,j} \in [0, 1]$ 。

另外, 给出根据 TRAVOS 模型用所有阶段的数据求得的平均交互成功率 it_i^j 。

$$it_i^j = \frac{1 + \sum_{k=1}^m s_j^k}{2 + \sum_{k=1}^m s_j^k + \sum_{k=1}^m f_j^k} \quad (3.4)$$

由于 CMAIT 采用的是与 TRAVOS 相同的计算方法, 而且其采用的数据量只是 TRAVOS 的一部分, 因而其计算量比 TRAVOS 小。 it_i^j 满足以下性质。

性质 3.1 历史平均交互成功率是 CMAIT 的特例: $m=1, d(k-1, k)=0$ 。

证明: 当 $d(k-1, k)=0$ 时,

$$it_{i,j}^t = E_{i,j}^k,$$

这是 CMAIT 的特例, 即变化率为零的情形, 此时未来交互的成功率就等于最近一个阶段交互的成功率。而当 $m=1$ 时,

$$it_{i,j}^t = E_{i,j}^k (m=1) = it_i^j。$$

即此时为 TRAVOS 的计算模型, 取历史交互平均成功率。证毕。

性质 3.2 每一个阶段 k Agent i 对 Agent j 的信任值 $E_{i,j}^k \in (0, 1)$, $it_{i,j}^t \in [0, 1]$ 。

证明: 考虑两种极限情况, 一是 $s_j^k = 0$, $f_j^k \rightarrow \infty$, 即所有交互都失败了, 此时信任应该最低,

$$E_{i,j}^k = \frac{0+1}{0+f_j^k+2} \rightarrow \frac{1}{\infty} \rightarrow 0;$$

二是 $s_j^k \rightarrow \infty$, $f_j^k = 0$, 即所有交互都成功了, 此时信任应该最高,

$$E_{i,j}^k = \frac{s_j^k+1}{0+s_j^k+2} \rightarrow \frac{\infty}{\infty} \rightarrow 1。$$

故 $E_{i,j}^k \in (0, 1)$ 。 $it_{i,j}^t \in [0, 1]$ 可由函数 \max 和函数 \min 决定, 参见式 3.3。证毕。

性质 3.3 $t=t_k, it_{i,j}^t = E_{i,j}^k$; $t=t_{k-1}, it_{i,j}^t = E_{i,j}^{k-1}$, 此时计算误差为 0。

证明: 由性质 3.2 知: $E_{i,j}^k \in (0, 1)$ 。若 $t=t_k$, 得

$$\begin{aligned} it_{i,j}^t &= \min\{1, \max\{E_{i,j}^k + \frac{E_{i,j}^k - E_{i,j}^{k-1}}{T_k}(t_k - t_k), 0\}\} = \min\{1, \max\{E_{i,j}^k + \frac{E_{i,j}^k - E_{i,j}^{k-1}}{T_k} \cdot 0, 0\}\}, \\ &= \min\{1, \max\{E_{i,j}^k, 0\}\} = \min\{1, E_{i,j}^k\} = E_{i,j}^k \end{aligned}$$

若 $t=t_{k-1}$, 得

$$\begin{aligned}
it'_{i,j} &= \min\{1, \max\{E_{i,j}^k + \frac{E_{i,j}^k - E_{i,j}^{k-1}}{T_k}(t_{k-1} - t_k), 0\}\} = \min\{1, \max\{E_{i,j}^k + \frac{E_{i,j}^k - E_{i,j}^{k-1}}{T_k}(-T_k), 0\}\} \\
&= \min\{1, \max\{E_{i,j}^k + E_{i,j}^{k-1} - E_{i,j}^k, 0\}\} = \min\{1, \max\{E_{i,j}^{k-1}, 0\}\} = \min\{1, E_{i,j}^{k-1}\} = E_{i,j}^{k-1}
\end{aligned}$$

证毕。

3.4.3 置信度计算

置信度可以分为两个方面：计算信息量置信度、计算偏离置信度。直观上，采用的证据越多、信任值变化率越小，则置信度越高。计算信息量置信度考察采用不同的数据量得到相同信任值情况下，其置信度有不同值。如最近两个阶段的交互次数越多，则信任值的置信度越高。计算偏离置信度考察各个阶段信任值与平均值接近的程度，即计算结果越稳定，计算变化率越小，则置信度越高。一个有意思的例子：如果一个 Agent 的信任值很低但是保持稳定，无大的变化，则其置信度很高，而另一个 Agent 的信任大多数情况下很低，但是偶尔会高很多，则其置信度反而会较低。这是因为该 Agent 并不稳定，或者含有噪音，或者可能想通过偶尔高姿态来欺骗对手，造成假象。当然信任值一直高，偶尔低，其置信度也不会高。

计算信息量置信度为

$$\gamma_i^j = \frac{s_j^{k-1} + s_j^k + f_j^{k-1} + f_j^k}{\sum_{k=1}^m s_j^k + \sum_{k=1}^m f_j^k} \quad (3.5)$$

其中， $\gamma_i^j \in [0, 1]$ 是 Agent i 对 Agent j 交互信任的计算信息量置信度。

性质 3.4 最近两个阶段交互次数越多，则计算信息量置信度越高。

证明：由式 3.5 可知，当

$$\sum_{k=1}^m s_j^k + \sum_{k=1}^m f_j^k = C,$$

即交互总次数为常数时， $s_j^{k-1} + s_j^k + f_j^{k-1} + f_j^k$ 越大，即最近两个阶段交互的次数越多，则 γ_i^j 越大，即计算信息量置信度越高。证毕。

计算偏离置信度为

$$\gamma_D^j = 1 - \sum_{k=2}^m (it_{i,j}^k - it_i^j)^2 \quad (3.6)$$

其中, $\gamma_D^j \in [0, 1]$ 是 Agent i 对 Agent j 交互信任的计算偏离置信度, $it_{i,j}^k$ 是第 k 个阶段的交互信任值。

性质 3.5 每个阶段的信任值变化越缓慢, 即信任值越稳定, 则计算偏离置信度越高。

证明: 信任值越稳定, 则 $it_{i,j}^k - it_i^j$ 越小,

$$\sum_{k=2}^m (it_{i,j}^k - it_i^j)^2$$

也越小,

$$1 - \sum_{k=2}^m (it_{i,j}^k - it_i^j)^2$$

越大, 则 γ_D^j 越大, 即计算偏离置信度越高。证毕。

总体置信度为

$$\gamma_i^j = \gamma_i^j \cdot \gamma_D^j \quad (3.7)$$

其中, γ_i^j 是 Agent i 对 Agent j 交互信任的总体置信度。

利用计算偏离置信度和动态变化率, 可以对对手进行异常行为检测。具体方法可以设定偏离容许值 τ_D , 若在某个阶段, 信任值偏离过大, 则认为是异常行为。

3.4.4 编程

我们用 MATLAB 进行编程实验。

算法 3.1 动态信任计算算法

1. 给定 $n, m, t, M, \tau_I, \tau_D$
2. for $i=1:m$
3. 用 beta 分布求取各个分段信任值 $e(i):=(s(i)+1)/(s(i)+f(i)+2)$;
4. 计算信任变化率, 并求动态信任值 $j:=i+1; d(1):=0; d(j):=e(j)-e(i); it(j):=e(i)+d*(t-ti)$;
5. 求计算量置信度 $gi(j):=(s(i)+s(j)+f(i)+f(j))/(sum(s)+sum(f))$

6. 求计算偏离置信度 gd
7. end for
8. if $gd < 1 - \tau_D$
9. 显示该 Agent 不可信
10. end if

3.5 实验结果及分析

3.5.1 实验背景和参数

在网上电子商务应用中，多 Agent 系统可以代理客户完成交易，体现客户的意愿，实现客户收益的最大化。在互联网的开放环境中，由于信息的不可靠和不精确，如何选择合作伙伴就成为了一个重要问题。合作伙伴的选择主要依据是看能否取得成功的收益，但是也要选择信任值较高的 Agent。

本章以网上电子商务中的淘宝网的手机及其配件购买为背景，取 2008 年 3 月 28 日网站保存的对商家好评等级记录。本实验目的是比较 CMAIT 与其它模型的预测误差，验证其异常行为检测机制，从而验证该计算模型的合理性。

S1,S2,S3,S4 分别表示第一个阶段（半年之前）、第二个阶段（半年之内）、第三个阶段（一个月之内）和第四个阶段（一周之内）的成功交互次数，T1,T2,T3,T4 各列分别表示四个阶段的交互总数。

表 3.3 源数据：每个阶段和总体成功交互次数

Table 3.3 Data resource: interaction number of each stage and as a whole

卖家	S1	S2	S3	S4	S 总数	T1	T2	T3	T4	T 总数
lzsws	334	1306	382	79	1640	338	1308	383	80	1646
jojphp	1732	568	104	45	2300	1737	568	104	45	2305
yykele	7622	5112	1161	317	12734	7638	5155	1177	318	12793
ayalawang	3825	2707	478	127	6532	3827	2707	478	127	6534
jovi448750	798	41	37	4	880	839	41	37	4	859
kitty810720	6935	5406	1297	362	12341	6935	5407	1298	363	12342
wfs19830419	30	144	122	38	174	30	144	122	38	174
huagong1979ka	0	2636	71	23	2636	0	2636	71	23	2636
chenkai_ck	14638	7344	1826	363	21982	14646	7354	1834	365	22000
claxy888	8742	5642	1558	432	14384	8785	5645	1560	433	14430
apr_pud	362	39	13	3	401	386	42	13	3	428
Netoocn	22609	10467	2051	675	33076	22609	10467	2051	675	33076

3.5.2 信任计算实验

给定 Agent 之间的交互参数, 则可以求得 it_i^j , 实验结果如表 3.4 和图 3.4 所示。

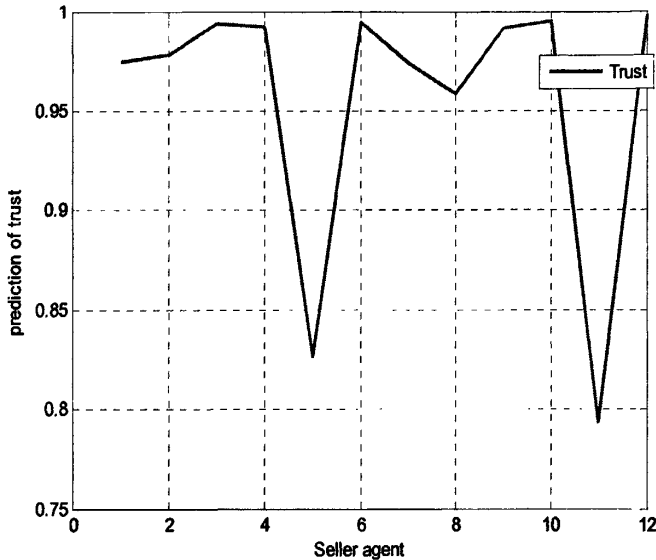


图 3.4 CMAIT 模型计算的 Agent 交互信任图
Figure 3.4 Agent interaction Trust using CMAIT

表 3.4 每个阶段的信任值和平均信任值
Table 3.4 Trust of each stage and average

卖家	S1	S2	S3	S4	CMAIT	TRAVOS
lzsws	0.98529	0.99771	0.99481	0.97561	0.97467	0.99575
jojph	0.99655	0.99825	0.99057	0.97872	0.97814	0.9974
yykele	0.99777	0.99147	0.98558	0.99375	0.99415	0.99531
ayalawang	0.99922	0.99963	0.99792	0.99225	0.99197	0.99954
jovi448750	0.97558	0.97674	0.97436	<u>0.83333</u>	0.82643	0.97561
kitty810720	0.99986	0.99963	0.99846	0.99452	0.99433	0.99984
wfs19830419	0.96875	0.99315	0.99194	0.975	0.97417	0.99432
huagong1979ka	<u>0.5</u>	0.99962	0.9863	0.96	0.95871	0.99962
chenkai_ck	0.99939	0.9985	0.9951	0.99183	0.99167	0.99914
claxy888	0.99499	0.99929	0.99808	0.9954	0.99527	0.99674
apr_pud	0.93557	0.90909	0.93333	<u>0.8</u>	0.79347	0.93488
Netoocn	0.99996	0.9999	0.99951	0.99852	0.99847	0.99997

在每一个阶段分别将真实值和 CMAIT、TRAVOS 求得值进行比较, 得到第三个、第四个阶段的计算误差、预测误差和总体计算误差, 见表 3.5、图 3.5 和 3.6。

表 3.5 计算误差

Table 3.5 Computational error

卖家	第 3 阶段		第 4 阶段		预测下一个信任值	
	CMAIT	TRAVOS	CMAIT	TRAVOS	CMAIT	TRAVOS
lzsws	1.35E-05	5.46E-06	0.00015371	0.00014439	4.42E-07	0.000203
jojphp	3.86E-05	2.33E-05	2.77E-05	0.00013443	1.68E-07	0.000174
yykele	1.54E-06	4.09E-05	6.66E-05	2.29E-06	8.00E-08	1.22E-06
ayalawang	1.97E-06	1.14E-06	1.10E-05	2.23E-05	3.85E-08	2.66E-05
jovi448750	4.95E-06	1.63E-06	0.0097526	0.010114	2.39E-05	0.010121
kitty810720	5.21E-07	8.21E-07	5.35E-06	1.15E-05	1.86E-08	1.41E-05
wfs19830419	3.25E-05	6.03E-05	0.00013187	4.62E-05	3.44E-07	0.000187
huagong1979ka	9.38E-05	0.027964	0.00017449	0.0086276	8.30E-07	0.000785
chenkai_ck	4.00E-06	7.40E-06	8.76E-07	1.70E-05	1.28E-08	2.67E-05
claxy888	1.84E-06	4.39E-07	1.97E-06	2.11E-06	8.60E-09	8.99E-07
apr_pud	0.00086459	6.05E-05	0.010834	0.0079376	2.13E-05	0.009097
Netoocn	6.41E-08	8.70E-08	2.93E-07	8.04E-07	1.18E-09	1.05E-06

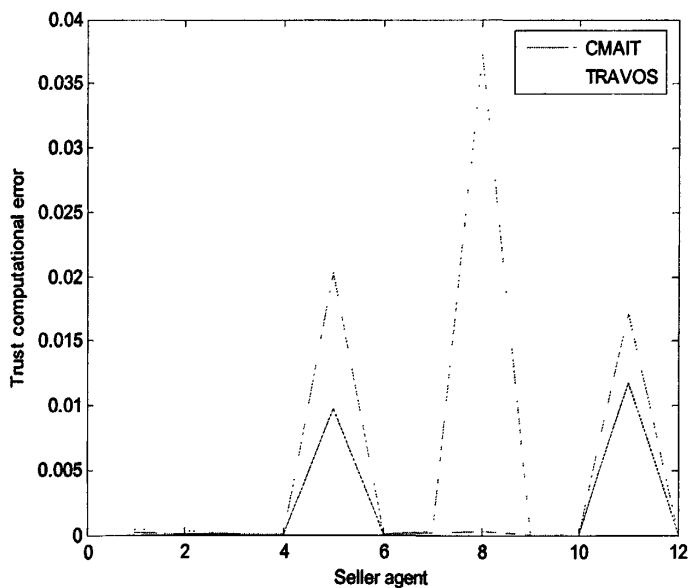


图 3.5 CMAIT 和 TRAVOS 模型总体计算误差比较

Figure 3.5 Total computational error comparison between CMAIT and TRAVOS

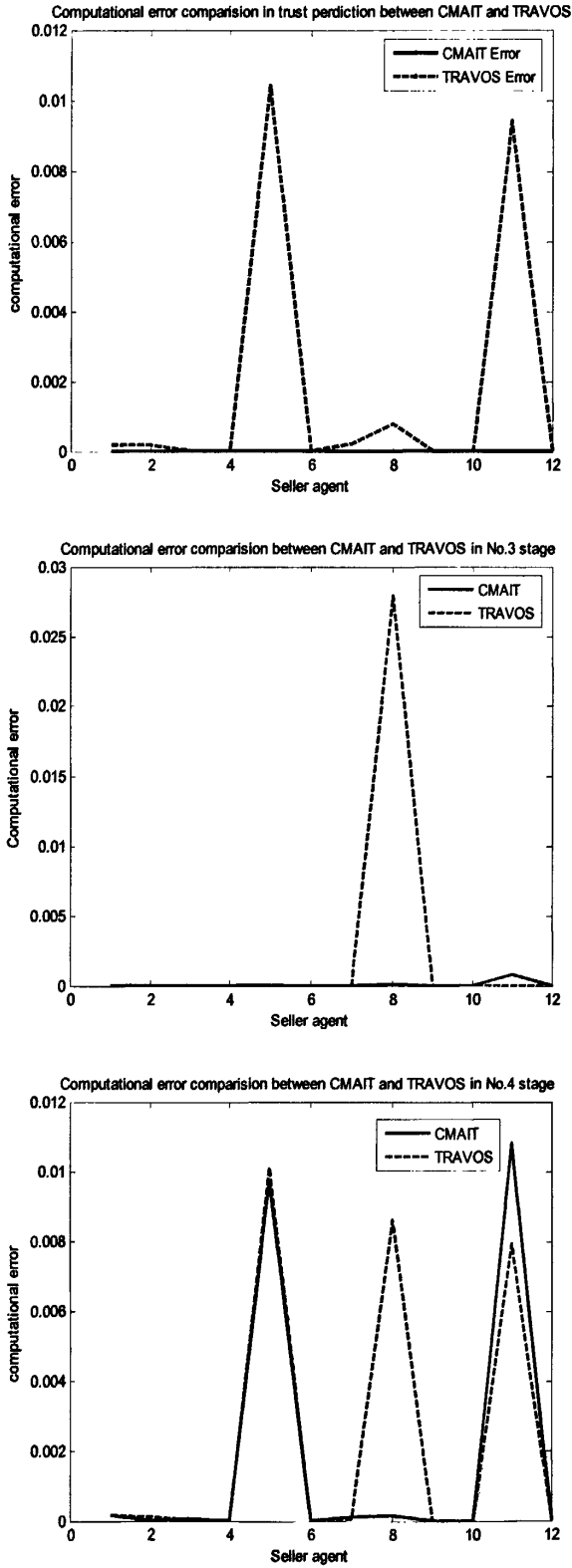


图 3.6 CMAIT 和 TRAVOS 的各阶段计算误差比较

Figure 3.6 Computational error comparison between CMAIT and TRAVOS

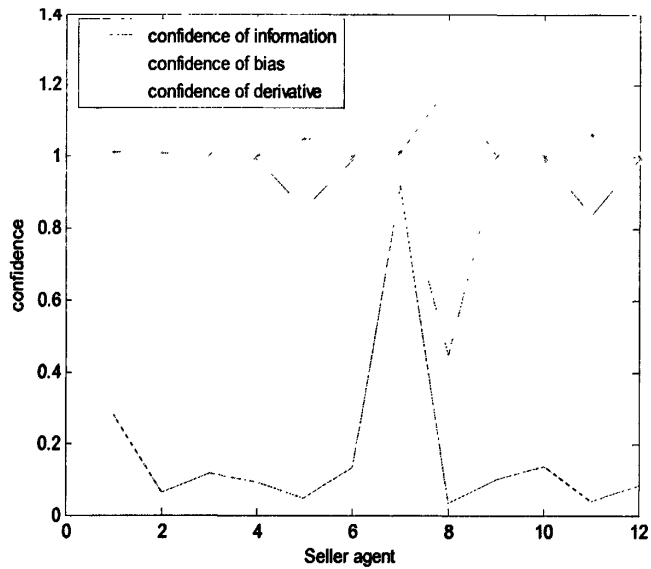


图 3.7 卖家 Agent 的计算置信度

Figure 3.7 Three confidence of seller Agent using CMAIT

总体误差分别为 $E_{CMAIT}=0.0222, E_{TRAVOS}=0.0552$ 。

从实验结果看，CMAIT 模型的计算误差是 TRAVOS 模型的 0.5 倍，能够进行合理的预测，较好符合实际情况，是正确的、可行的。

另外，我们还可以得到置信度计算结果，如表 3.6 所示。通过置信度计算结果，我们可以对 Agent 进行异常行为检测。

表 3.6 置信度和变化率计算

Table 3.6 Confidence and derivative computation

卖家	信息量 置信度	偏离 置信度	总体 置信度	1-2 阶段变 化率	2-3 阶段变 化率	3-4 阶段变 化率
lzsws	0.21954	0.96653	0.21464	-0.019195	-0.0029047	0.012416
jojphp	0.060717	0.97248	0.059465	-0.011843	-0.0076796	0.0016959
yykele	0.10463	0.9838	0.10364	0.008169	-0.0058869	-0.006307
ayalawang	0.084746	0.99059	0.084167	-0.0056686	-0.0017142	0.00041435
jovi448750	0.045606	<u>0.84881</u>	0.040622	<u>-0.14103</u>	-0.0023852	0.0011642
kitty810720	0.11862	0.99333	0.11804	-0.003941	-0.0011687	-0.0002256
wfs19830419	0.47904	0.94074	0.45709	-0.016935	-0.0012152	0.024401
huagong1979ka	0.034432	<u>0.44739</u>	0.018438	-0.026301	-0.01332	<u>0.49962</u>
chenkai_ck	0.090872	0.98844	0.090048	-0.0032724	-0.0034066	-0.00088096
claxy888	0.12135	0.99302	0.12084	-0.0026771	-0.0012123	0.0042991
apr_pud	0.036036	<u>0.82911</u>	0.031768	<u>-0.13333</u>	0.024242	-0.026476
Netoocn	0.076141	0.99801	0.076026	-0.00099001	-0.00039157	-5.13E-05

由表 3.6 和图 3.7 可知，信息量置信度整体较低，这是由于只采用了最近两个阶段的数据，因而导致这种情况出现。另外由表中加下划线的数据可知，这些点

的数据与其它数据相比有明显的区别，这可为异常行为检测提供依据。

实验可知，卖家 huagong1979ka，jovi448750，apr_pud 不可信，可能存在异常行为，其中卖家 huagong1979ka 最不可信。

3.6 小结

目前 Agent 信任已逐渐成为多 Agent 系统的研究热点之一，它能为 Agent 选择合作伙伴和提高收益提供有益的帮助。但是就直接交互信任研究而言，仍有很多可以改进的地方，本章试图从交互信任中得到更多的对未来交互有用的信息。结合信任的动态变化率，本章给出了 Agent 分段动态信任计算模型 CMAIT 及其置信度，实验结果表明 CMAIT 模型的计算误差是 TRAVOS 模型的 0.5 倍，计算量也较少，而且能够进行异常行为检测。

本章未考虑信任计算的二阶和更高阶导数，如果考虑，也许会进一步减少计算误差，但是计算量会显著增加；另外，本章的信任计算采用的是概率论的方法，没有采用逻辑的方法，因而未将 BDI 模型作为基础。将来的工作可以关注于如何使 Agent 信任计算更可靠，如何减少 Agent 信任计算误差，提高预测的准确性；还需研究 Agent 信任与 Agent 决策的逻辑关系。

第四章 一种 Agent 长期联盟信誉模型

在上一章我们介绍了个体 Agent 之间信任的动态性以及异常行为检测方法。现实生活中,很多情况下个体 Agent 需要和长期联盟(如某些电子商务网站)进行反复交互,那么在交互过程中,Agent 需要依据效用和信任选择一个最优的联盟。然而,之前很少有联盟信任方面的研究工作。

为此,本章提出了一种长期联盟信誉模型 LCCM(Long-term Coalition Credit Model),而且给出了联盟信誉和联盟收益之间的关系。在本章中,信誉被定义为内部信任和外部声誉的加权和,LCCM 包括基于 Agent 直接交互的内部信任和基于 Agent 直接观察的外部声誉。本章通过和合作和竞争环境的实验验证了 LCCM 可适用于不同的情况。实验结果显示 LCCM 能够有效的计算联盟信誉,并能反映不同参数对联盟信誉的影响。

4.1 引言

Agent 联盟是复杂任务求解的重要方式之一,通过联盟形成,个体 Agent 和 Agent 联盟都能取得收益最大化。Agent 联盟通常都是面向目标和短期的,为了一个求解目标而形成联盟,当任务完成时联盟就解散。稳定的联盟要求 Agent 满足超加性,形成联盟核^[61],然而判定核非空的计算复杂度是 NP 完全的^[14]。随着联盟研究的不断深入,短期联盟的高耗性和不稳定性等弊端日益显现,近年来很多工作基于信任提出了长期联盟。

在某些领域里,Agent 经常与一些长期联盟多次交互,如淘宝网等,Agent 通常都保存着大量的交互历史,并能从交互历史中获取经验。个体 Agent 一般选择信誉值较高的联盟作为合作的伙伴,因为较高的信誉能够降低交互失败率,避免被联盟利用,并能提高交互收益。由此,联盟信誉的研究越来越重要。

联盟收益是多 Agent 系统联盟研究中的一个关键问题。为了保证个体 Agent 和联盟均能取得最大收益,在联盟形成之前需要计算联盟收益。之前基于对策论讨论的联盟都只关注于有形收益,如 Sandholm 等人^[60-61]提出的联盟核和 Shehory 等人^[67-68]提出的内核联盟。然而,对长期联盟而言,无形收益不应该被忽略,因为联盟信誉将带来大量的收益。如商标作为一种信誉能够长期给企业带来巨大的效益,百年老字号的招牌蕴含着巨大的无形资产,并能给加盟连锁店带来长期稳

定的收益。

有许多关于个体 Agent 的信任和声誉研究工作^[7,31-32,35-37,45,54,58,69-70], 如 Huynh 等人^[31-32]提出的个体 Agent 信任和声誉 FIRE 模型, 集成了交互信任、角色信任、观察声誉和认证声誉等。其问题是公共可信第三方 Agent 难于管理和维护, 易造成通信瓶颈。Sabater 等人^[58]提出的 REGRET 模型和 Teacy 等人^[69-70]提出的 TRAVOS 模型用概率论研究个体 Agent 之间信任关系, 用 *Beta* 分布的数学期望表示信任关系, 给出了基于概率论的 Agent 个体信任计算模型, 并有一定的滤除噪声能力。

以上研究都是对个体 Agent 的信任和声誉研究, 没有涉及 Agent 联盟信任。而个体 Agent 对 Agent 联盟的信任是常见的情况, 如消费者购买某公司的各种产品从而对该公司产生了整体评价。另一方面, 个体 Agent 大多是以组织成员的形式存在, 因而联盟信任对个体信任有直接影响, 如银行放贷会考虑客户所处的行业或职业的可信度。然而, 只有很少的工作如 Blankenburg、Vassileva 等人^[8,28,75]涉及 Agent 联盟信誉问题, Vassileva 等人^[75]基于经验提出了长期联盟信任模型, 指出 Agent 可以基于以往的经验 and 当前的信任值形成新的信任值, Griffiths 等人^[28]基于动机和信任提出了 Agent 党派, Blankenburg 等人^[8]认为 Agent *i* 对 Agent *j* 的信任值取决于 Agent *i* 对 Agent *j* 的成功概率评估和联盟中其他 Agent 对 Agent *j* 的成功概率评估。这些工作都没有给出一个可行的计算方法, 而且这些已知的模型在过滤不精确和撒谎信息方面都有所欠缺, 而这又是信任研究领域一个很重要的基本问题。因此, 有必要对这方面的工作做出有效的改进, 并能给出个体信任和联盟信誉之间的关系。

长期联盟信誉的形成是一个基于 Agent 交互的历史过程, 本章考虑一个有限交互历史和不完全信息的长期联盟, Agent 有自己的主观偏好, 该偏好可被视为在交互历史较少情况下信誉的先验概率或基于角色的信任。交互历史越少, 主观偏好的作用越大。基于以上假设, 本章提出一种长期联盟信誉模型 LCCM, 将内部信任和外部声誉加权平均集成为信誉。该模型可以在多项式级时间范围内计算出联盟信誉。

Agent 联盟信誉可以应用于多个领域, 如电子商务、虚拟组织等。LCCM 既能应用于合作环境又能应用于竞争环境。首先将实验设置在网上购物的淘宝网站上购买手机及配件。淘宝网上有多家手机供应商, 买家与卖家有许多的交互历史。本章中将这些交互视为一个买家 Agent, 这样, 在我们的实验环境中, 好像这些交互都是属于一个买家, 即一个买家和一个长期联盟(淘宝网)进行交互。其次, 将实验应用于中超联赛, 计算某一球队对整个联盟的“信任”(取胜概率)。如果该队赢了对手, 则认为是成功的交互, 如果他打平或输给对手, 则认为是失败的

交互。基于以上假设，该球队对整个联盟的信任值可以求得。实验结果显示 LCCM 在有限交互历史条件下能有效地计算联盟信誉，且能反映不同参数，如进出联盟 Agent 数量、成功交互概率及主观偏好等对联盟信誉的作用。

本章在 Agent 联盟和 Agent 信誉方面做出了有益的探索。建立起了联盟信誉和联盟收益的关系，提供一个计算联盟收益的工具。其次，LCCM 能处理交互历史较少的情况。一般情况下，由于缺少证据导致信任的计算置信度过低，然而，通过使用主观偏好，LCCM 能得到相对好的估计。再次，LCCM 能有效地滤除不精确和撒谎信息，从而避免被欺骗。因为所有的信息资源都是 Agent 的直接交互或直接观察，不包括中介信息，从而能提升计算置信度，并能滤除不精确信息，减少计算错误。

4.2 问题描述

假设系统中总共有 n 个 Agent，其中，联盟内部有 m 个 Agent， $m \leq n$ ，每一个 Agent i 都有自己的信息状态 I_i ，经过一段时间相互交互之后，Agent 之间形成了如图 4.1 所示的关系。其中，大圆圈表示 Agent 联盟，小圆圈表示个体 Agent，小圆圈之间的连线表示个体 Agent 之间的交互关系。从联盟中离开的 Agent 用带向外箭头的小圆圈表示，进入联盟的 Agent 用带向内箭头的小圆圈表示。

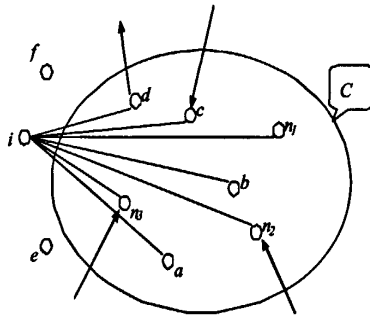


图4.1 Agent交互示意图

Figure 4.1 An example of Agent interaction

长期联盟信誉包括两个方面：联盟内部信任，表示为 T ；联盟外部声誉，表示为 R 。 T 基于 Agent 之间的交互历史，计算联盟内部信任为 Agent i 与联盟中 Agent 的交互信任的加权平均。 R 通过 Agent 的直接观察得到，如果有外部的 Agent 进入联盟，则联盟声誉提升；如果有内部的 Agent 离开联盟，则联盟声誉降低。

定义 4.1 个体 Agent 信息状态

$$I_i = \langle I_i^1, I_i^2, \dots, I_i^m \rangle$$

其中,

$$I_i^j = [v_{i,j}^1, v_{i,j}^2, \dots, v_{i,j}^{s_i^j}],$$

其中, I_i^j 是一个向量, 保存着 Agent i 和 Agent j 之间的以时间排序的交互历史, s_i^j 表示 Agent i 和 Agent j 之间的交互次数, 显然, $s_i^j = s_j^i$, $v_{i,j}^l$ 表示 Agent i 和 Agent j 之间的第 l 次交互结果, $l = 1, 2, \dots, s_i^j$ 。

$$v_{i,j}^l = \begin{cases} 1 & \text{若交互成功} \\ 0 & \text{其它} \end{cases} \quad (4.1)$$

交互成功即 Agent i 实现了自己的目的, 反之, Agent i 没有达到目的, 则称为失败的交互。

假设 4.1 联盟内的 Agent 数量越多, 则联盟信任在联盟信誉中的比重越大。

依据假设 4.1

定义 4.2 长期联盟信誉

$$\Phi_c = \frac{1}{n} [(m + n_i - n_o)T + (n - m - n_i + n_o)R] \quad (4.2)$$

其中, Φ_c 表示 Agent 长期联盟信誉, n_i 是在观察周期内进入联盟的 Agent 的数量, n_o 是在观察周期内离开联盟的 Agent 的数量。

$T \in [0, 1]$ 是基于 Agent 直接交互的联盟内部信任。如果 Agent i 完全信任联盟内的每一个 Agent, 则 $T = 1$; 如果 Agent i 完全不信任联盟内的每一个 Agent, 则 $T = 0$; 否则, $T \in (0, 1)$ 。

$R \in [-1, 1]$ 是基于 Agent 直接观察的外部声誉, 若所有的联盟外的 Agent 在最后都进入了联盟, 则说明该联盟有很强的吸引力, $R = 1$; 若所有的联盟内的 Agent 在最后都离开了联盟, 则说明该联盟声誉很差, $R = -1$; 否则, $R \in (-1, 1)$ 。

显然, 定义 4.2 满足假设 4.1。

联盟无形收益受很多因素的影响, 如信誉、历史、角色等。在本章中, 忽略其它因素的作用, 假设联盟无形收益决定于联盟信誉, 给出了联盟信誉和联盟收益之间的关系函数。根据微观经济学的边际效用递减原理, 该关系饱和和非线性。

定义 4.3 长期联盟收益和信誉之间的关系

$$V_c = \frac{1}{\sqrt{2\pi}\sigma} e^{-(\Phi_c - 1)^2 / 2\sigma^2} V \quad (4.3)$$

其中， V 是未包括无形收益的联盟有形收益， V_C 是包括无形收益的联盟有形收益和无形收益之和， σ 是最大系数因子，表征无形收益对有形收益的放大作用， σ 值依赖于领域知识和环境。

性质 4.1 联盟收益的最小值在联盟信誉最小值时取得；联盟收益的最大值在联盟信誉最大值时取得。

$$\text{证明: } \lim_{\Phi_C \rightarrow -1} V_C = \frac{1}{\sqrt{2\pi}\sigma} e^{-(-1-1)^2/2\sigma^2} V = \frac{1}{\sqrt{2\pi}\sigma} e^{-2/2\sigma^2} V = \frac{1}{\sqrt{2\pi}\sigma} e^{-1/\sigma^2} V,$$

$$\text{即 } V_C \text{ 的最小值是 } \frac{1}{\sqrt{2\pi}\sigma} e^{-1/\sigma^2} V.$$

$$\lim_{\Phi_C \rightarrow 1} V_C = \frac{1}{\sqrt{2\pi}\sigma} e^{-(1-1)^2/2\sigma^2} V = \frac{1}{\sqrt{2\pi}\sigma} e^{-0/2\sigma^2} V = \frac{1}{\sqrt{2\pi}\sigma} V,$$

$$\text{即 } V_C \text{ 的最大值是 } \frac{1}{\sqrt{2\pi}\sigma} V. \text{ 证毕.}$$

V_C 和 Φ_C 之间的关系如图 4.2 所示。

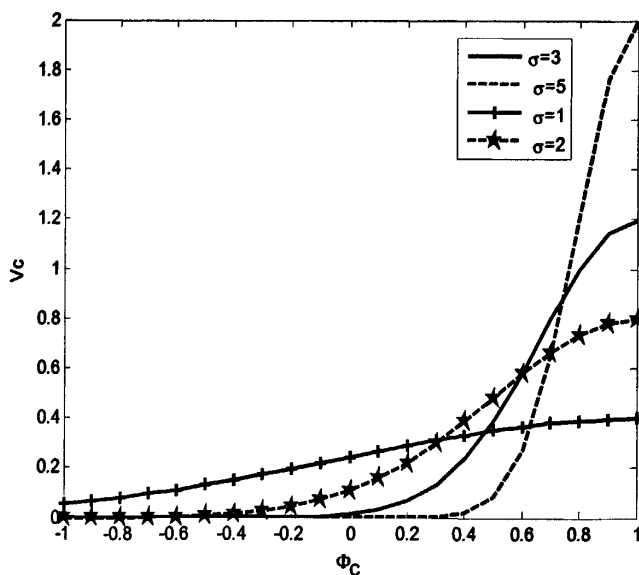


图4.2 V_C 和 Φ_C 之间的关系

Figure 4.2 Relationship between V_C and Φ_C

联盟收益随联盟信誉的动态变化如表 4.1 所示。随着 σ 值从 1 变化到 5，联盟信誉变得越重要，表 4.1 的数据能够清楚说明 V_C 和 Φ_C 的数量关系。一般来说二者之间的关系是一种饱和非线性的关系，在联盟信誉值小于某一个较小的阈值时，联盟信誉对联盟收益没有作用；当联盟信誉值超过该阈值时，二者的关系近似于线性关系；当联盟信誉值达到一定值时，其对联盟收益的影响就明显变小，甚至当联盟信誉达到某一较大的阈值时，联盟收益达到饱和，联盟信誉的增长对联盟

收益就不会产生影响了。

表 4.1 V_C 和 Φ_C 之间的关系
Table 4.1 Relationship between V_C and Φ_C

Φ_C	V_C			
	$\sigma=1$	$\sigma=2$	$\sigma=3$	$\sigma=5$
-1.0000	0.0540	0.0003	0.0000	0.0000
-0.9000	0.0656	0.0006	0.0000	0.0000
-0.8000	0.0790	0.0012	0.0000	0.0000
-0.7000	0.0940	0.0025	0.0000	0.0000
-0.6000	0.1109	0.0048	0.0000	0.0000
-0.5000	0.1295	0.0089	0.0000	0.0000
-0.4000	0.1497	0.0158	0.0002	0.0000
-0.3000	0.1714	0.0272	0.0006	0.0000
-0.2000	0.1942	0.0448	0.0018	0.0000
-0.1000	0.2179	0.0709	0.0052	0.0000
0.0000	0.2420	0.1080	0.0133	0.0000
0.1000	0.2661	0.1579	0.0313	0.0001
0.2000	0.2897	0.2218	0.0672	0.0007
0.3000	0.3123	0.2995	0.1320	0.0044
0.4000	0.3332	0.3884	0.2369	0.0222
0.5000	0.3521	0.4839	0.3886	0.0876
0.6000	0.3683	0.5794	0.5826	0.2700
0.7000	0.3814	0.6664	0.7983	0.6476
0.8000	0.3910	0.7365	0.9997	1.2099
0.9000	0.3970	0.7821	1.1442	1.7603
1.0000	0.3989	0.7979	1.1968	1.9947

4.3 长期联盟信誉模型

长期联盟信誉的计算流程图如图 4.3 所示。首先根据 TRAVOS 模型用 Beta 分布计算各个分段的信任值，结合主观偏好、客观系数、主观系数得到个体信任值，接下来利用权重系数得到联盟信任 T 。其次，根据进出联盟的 Agent 数量得到联盟声誉 R ，最后把联盟信任和联盟声誉加权平均，得到联盟信誉。

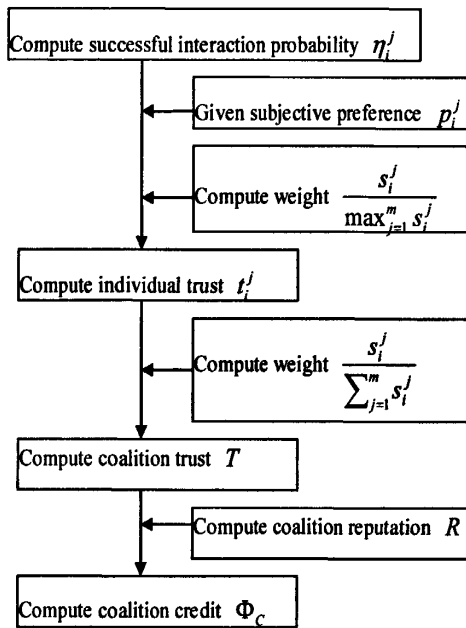


图4.3 计算流程图

Figure 4.3 flow chart of LCCM computation

4.3.1 长期联盟信任计算

本章不仅考虑 Agent 之间的交互历史,还考虑到自主 Agent 的一个重要性质——自主偏好。直观上,如果缺少足够的交互历史来计算信任,则 Agent 不应该盲目的相信来自其他 Agent 的信息资源,除非该 Agent 有足够的证据证明他的信息是可靠的。在证据不足的情况下,Agent 应该相信自己的主观偏好,抛弃不可靠的间接信息。在这种情况下,该偏好也可被认为是对对手信任的信念的先验概率,则信任形成过程可被看做是一种信念修正过程。

Agent 自主偏好的另一个优点是可以滤除不精确信息。例如不精确或撒谎信息对拥有不同自主偏好的人作用是不同的,有强烈的主观偏好的人(如某种信仰)就能够不为所动,从而滤除这些信息,而完全没有主见的人则容易被欺骗。实验表明如果系统中有某些不精确或撒谎信息,则 LCCM 能有效地处理这些信息,其计算误差要远远小于 BRS 系统。

假设 4.2 当 Agent 之间的交互历史比较少时,交互成功概率的置信度较低,此时 Agent 的主观偏好在信任中占较大比重。

假设 4.3 联盟信任应该与拥有较大交互次数的 Agent 的信任保持一致,即拥有更多交互次数的主流 Agent 可以反映联盟信任。

计算联盟信任 T 需要两步, 首先计算 Agent 的个体信任 t_i^j , 即 Agent i 对 Agent j 的个体信任, 然后, 根据权重计算整个联盟信任 T 。

实际上, 信任是包含了 Agent 自身信念和知识的主观标准。然而, 之前的研究大多忽略了 Agent 的偏好和信念。本章将考虑 Agent 的主观偏好, 将其视为初始化给定的常数。

因此, Agent i 对 Agent j 的个体信任

$$t_i^j = \frac{s_i^j}{\max_{j=1}^m s_i^j} \eta_i^j + (1 - \frac{s_i^j}{\max_{j=1}^m s_i^j}) p_i^j \quad (4.4)$$

其中, $t_i^j \in [0, 1]$ 是 Agent i 对 Agent j 的信任值, $p_i^j \in [0, 1]$ 是 Agent i 对 Agent j 的主观偏好, 由 Agent i 独立给出。 $\eta_i^j \in [0, 1]$ 表示 Agent i 和 Agent j 之间的平均交互成功率, 引用 TRAVOS 模型的定义^[70]。

$$\eta_i^j = \frac{1 + \sum_{l=1}^{s_i^j} v_{i,j}^l}{2 + s_i^j} \quad (4.5)$$

客观参数

$$\frac{s_i^j}{\max_{j=1}^m s_i^j}$$

由交互次数决定, 即该值越大, Agent 越客观。若 Agent i 有最大的交互次数, 则它是完全客观的。

Agent i 对整个联盟的信任值与该 Agent 对联盟内所有 Agent 的个体信任有关。

$$T = \sum_{j=1}^m w_i^j t_i^j \quad (4.6)$$

其中, T 是联盟的整体信任值, w_i^j 是 Agent j 的权重参数, 也可以理解为置信度标准。Agent i 和 Agent j 之间的交互次数越多, 该权重就越大。因此, Agent 之间的交互次数可影响信任计算的置信度, 即

$$w_i^j = \frac{s_i^j}{\sum_{j=1}^m s_i^j} \quad (4.7)$$

显然,

$$\sum_{j=1}^m w_i^j = 1。$$

因此长期联盟信任计算的统一公式为

$$T = \sum_{j=1}^m \frac{s_i^j}{\sum_{i=1}^m s_i^j} \left[\frac{s_i^j}{\max_{j=1}^m s_i^j} \eta_i^j + \left(1 - \frac{s_i^j}{\max_{j=1}^m s_i^j} \right) p_i^j \right] \quad (4.8)$$

其中， T 是联盟的整体信任值。

性质 4.2 T 的最小值是 0，最大值是 1。

证明：从公式 4.4 可知， $t_i^j \in [0, 1]$ ，不论 $w_i^j \in [0, 1]$ 是多少， $T \in [0, 1]$ 。证毕。

性质 4.3 LCCM 满足假设 4.2 和 4.3。

证明：根据公式 4.4，若 s_i^j 很小，则 $s_i^j / \max_{j=1}^m s_i^j$ 也很小，则 η_i^j 占 t_i^j 得比重很小，而 p_i^j 占 t_i^j 得比重很大。

根据公式 4.6 和 4.7， w_i^j 越大， t_i^j 越重要，即 T 与 w_i^j 一致；而 w_i^j 与 s_i^j 一致。根据传递性， T 与 s_i^j 一致。而且，根据假设 4.2， s_i^j 越大， η_i^j 的权重越大，即 s_i^j 与 η_i^j 一致。所以 T 由拥有最大交互次数 s_i^j 的 Agent 决定。证毕。

4.3.2 长期联盟声誉计算

如果环境是封闭的，系统中所有的 Agent 都在联盟内，并且没有 Agent 进出联盟，即 $(n-m) = 0$ 且 $(n_i - n_o) = 0$ ，则长期联盟声誉无需考虑，即

$$(n - m + n_o - n_i) = 0 \Rightarrow (n - m + n_o - n_i) * R = 0 \Rightarrow \Phi_C = \frac{m}{n} T = T。$$

如果系统中有某些 Agent 初始化时不在联盟内，即 $(n-m) > 0$ ，则他们可以通过进出联盟影响联盟的声誉。

假设 4.4 进入联盟的 Agent 数量越多，则 R 越大。

假设 4.5 联盟声誉 R 的最小值 -1 在所有 Agent 都离开联盟时得到，即 $n_i - n_o = -m$ ；而最大值在所有 Agent 都加入联盟时得到，即 $n_i - n_o = n - m$ 。

根据以上假设，给出 R

$$R = \begin{cases} \frac{n_i - n_o}{m} & (n_i - n_o) \leq 0 \\ 1 - e^{-\beta \frac{n_i - n_o}{(n-m-n_i+n_o)}} & (n_i - n_o) > 0 \end{cases} \quad (4.9)$$

其中, β 是常数, 表征 $(n_i - n_o)$ 对 R 的影响。

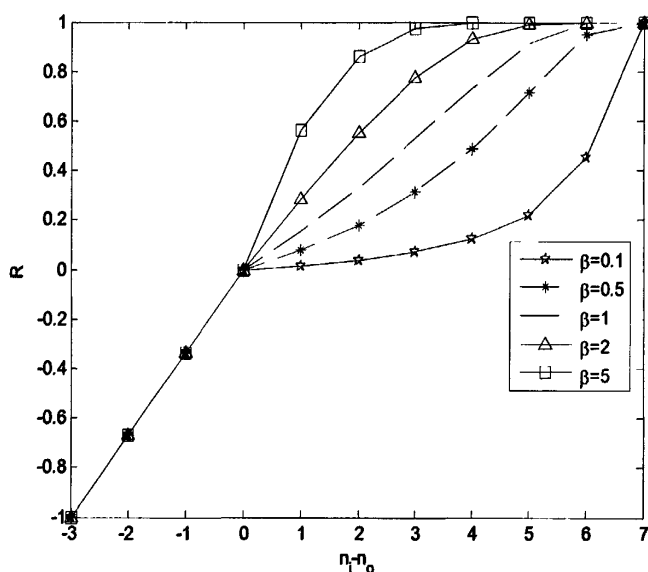


图4.4 联盟声誉示意图

Figure 4.4 Agent coalition reputation

如果 $n_i = n_o$, 则联盟是稳定的, $R = 0$; 如果 $n_i > n_o$, 则联盟扩张, 收益增加, $R > 0$; 如果 $n_i < n_o$, 则联盟缩小, 收益降低, $R < 0$ 。

性质 4.4 LCCM 满足假设 4.4 和 4.5。

证明: 由公式 4.9 可知, 当 $(n_i - n_o) = -m$ 时, 即所有的 Agent 离开该联盟, 得

$$R = \frac{n_i - n_o}{m} = \frac{-m}{m} = -1,$$

即 R 的最小值是 -1。

当 $(n_i - n_o) = (n - m)$ 时, 得

$$(n - m - n_i + n_o) = 0$$

$$\Rightarrow -\beta \frac{n_i - n_o}{(n - m - n_i + n_o)} \rightarrow -\infty$$

$$\Rightarrow e^{-\beta \frac{n_i - n_o}{(n - m - n_i + n_o)}} \rightarrow 0$$

$$\Rightarrow 1 - e^{-\beta \frac{n_i - n_o}{(n-m-n_i+n_o)}} \rightarrow 1$$

$$\Rightarrow R \rightarrow 1,$$

所以 R 的最大值是 1。证毕。

性质 4.5 长期联盟信誉 Φ_C 的最小值是 -1，其最大值是 1。

证明：由性质 4.2 知 T 的最小值是 0，由性质 4.4 知 R 的最小值是 -1，则当 $m + n_i - n_o = 0$ 时，即所有 Agent 离开联盟时， $R = -1$ ，且

$$\Phi_C = \frac{1}{n} [(m + n_i - n_o)T + (n - m + n_o - n_i)R] = R = -1,$$

所以 Φ_C 的最小值是 -1；因为 T 和 R 的最大值均为 1，则 Φ_C 的最大值是 1。证毕。

4.3.3 编程

本章给出由 MATLAB 编程实现的算法，变量 $sucl$ 表示成功交互次数， $sum1$ 表示总的交互次数， p 表示 Agent 的主观偏好， ita 依据 TRAVOS 用 Beta 分布函数来计算交互成功概率， t 是 Agent 之间的个体信任， $weight$ 通过交互次数得到， tr 和 r 分别表示 T 和 R ， $phai$ 表示 Agent 长期联盟信誉。

算法 4.1 长期联盟信誉算法

1. 给定向量 $sucl$, $sum1$, p
2. 设定 m , n , ni , no , $beta$
3. $sum2:=0$; $tr:=0$; $ns:=ni-no$
4. For $i=1:m$
5. $ita(i):=(sucl(i)+1)/(sum1(i)+2)$
6. $t(i):=(1-sum1(i)/max(sum1))*p(i) + (sum1(i)/max(sum1)) * ita(i)$
7. $sum2:=sum2+sum1(i)$
8. $weight(i):=sum1(i)/sum2$
9. $tr:=tr+weight(i)*t$
10. End For
11. If $ns>0$ $r:=1-exp(-beta*ns/(n-m-ns))$
12. Else $r:=ns/m$
13. End If
14. $phai:=((m+ns)*tr+(n-m-ns)*r)/n$

定理 4.1 算法 4.1 的计算复杂度是多项式级的。

证明: 首先, R 的计算复杂度是常数级的, 而 t_i^j 的计算复杂度是 $O(m)$, T 的计算复杂度是 $O(mm)$, 因此, Φ_c 的计算复杂度是多项式级。证毕。

4.4 实验

为了验证所提出的模型的有效性和算法的效率, 我们在两种实验背景下进行了实验, 本实验首先演示长期联盟信誉的计算过程, 并在不同参数条件下进行对比。实验背景考虑电子商务的重要部分--网上购物。除了收益之外, 参与网上购物的 Agent 还需考虑所购物网站的信誉, 以保证交易能够顺利完成。在实验 1 中, 我们选取在淘宝网 (www.taobao.com) 上购买手机及其配件, 某消费者 Agent 计划在网站上购买手机, 该网站上有众多的商家在出售手机, 消费者与商家有大量的交互历史, 商家被组织形成一个长期联盟, 因而消费者在购买之前需要知道该联盟的信誉。在实验 2 中, 我们选取中国足球超级联赛作为竞争性的环境, 且只考虑联盟信任的情况, 即联盟稳定的情况。在实验 3 中我们对 LCCM 与其他模型的噪声过滤能力进行了比较。

4.4.1 电子商务实验

假设有 10 个 Agent 参与了电子商务, 其中 Agent $n1, n2, n3, d, e$ 形成了联盟 C , 在 2008 年 3 月 28 日, 从淘宝网站得到交互历史数据, 如表 4.2 所示。买方 Agent 的主观偏好基于卖方 Agent 的所在地区, 即若卖方 Agent 处于北京, 则买方 Agent 对其偏好为 0.96, 在上海为 0.92, 在深圳为 0.88, 如表 4.3 所示。在整个观察周期内, 3 个 Agent 进入联盟, 1 个 Agent 退出联盟, β 根据经验取值为 2.6。Agent 个体信任计算结果如表 4.4 所示。最后, $T=0.9724$, $R=0.8233$, $\Phi_c=0.9277$ 。

表 4.2 Agent 的交互历史数据

Table 4.2 Interaction history of Agent

Provider Agent	S number	U number	Total number	City of provider
lzsws	2101	8	2109	Shenzh
jojphp	2449	5	2454	Shenzh
yykele	14212	76	14288	Shenzh

ayalawang	7137	2	7139	Shanghai
jovi448750	880	19	899	Shanghai
kitty810720	14000	3	14003	Beijing
wfs19830419	334	0	334	Shenzh
huagong1979ka	2730	0	2730	Shenzh

表 4.3 Agent 的偏好

Table 4.3 Preference of Agent

City of provider	Beijing	Shanghai	Shenzhen
Preference	0.96	0.92	0.88

表 4.4 Agent 的偏好和个体信任

Table 4.4 The preferences and individual trusts

Provider Agent	Preference	η_i^j	t_i^j	w_i^j
lzsws	0.88	0.9957	0.8971	0.0480
jojphp	0.88	0.9976	0.9002	0.0558
yykele	0.88	0.9946	0.9946	0.3251
ayalawang	0.92	0.9996	0.9598	0.1624
jovi448750	0.92	0.9778	0.9236	0.0205
kitty810720	0.96	0.9997	0.9989	0.3186
wfs19830419	0.88	0.9970	0.8827	0.0076
huagong1979ka	0.88	0.9996	0.9029	0.0621

我们注意到 lzsws, jojphp, wfs19830419 和 huagong1979ka 的 η_i^j 值很高且接近于 1，但是他们的 t_i^j 值却相对很低，原因在于他们的交互次数过少，所以 η_i^j 没有足够的置信度来修正 t_i^j ，此时， p_i^j 决定了 t_i^j 。

4.4.2 不同参数实验

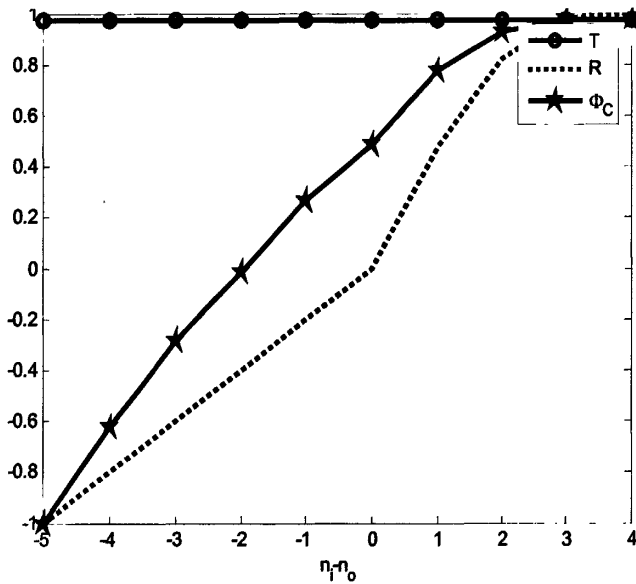


图4.5 $(n_i - n_o)$ 对 T , R 和 Φ_C 的影响

Figure 4.5 $(n_i - n_o)$ effect on T , R , and Φ_C

为了验证 LCCM 的合理性，我们设计了不同参数条件下的多个实验，首先为验证 $(n_i - n_o)$ 对 R 和 Φ_C 的影响，实验设置如下，将 $(n_i - n_o)$ 值从 -5 到 4 变化，然后计算 T , R 和 Φ_C ，实验结果如图 4.5 所示。

其次，为了验证 η_i^j 对 T 和 Φ_C 的影响，实验设置如下，使总交互次数每一步增加 100 次，然后计算 T , R 和 Φ_C ，实验结果如表 4.5 和图 4.6 所示。

表 4.5 η_i^j 对 T , R 和 Φ_C 的影响

Table 4.5 η_i^j effect on T , R , and Φ_C

$\eta_i^j (j=1zsws)$	$t_i^j (j=1zsws)$	T	R	Φ_C
0.9957	0.8971	0.9724	0.8233	0.9277
0.9507	0.8909	0.9659	0.8233	0.9231
0.9096	0.8847	0.9576	0.8233	0.9173
0.8718	0.8787	0.9504	0.8233	0.9122
0.8371	0.8727	0.9433	0.8233	0.9073
0.8051	0.8668	0.9363	0.8233	0.9024
0.7754	0.8610	0.9294	0.8233	0.8976

0.7478	0.8552	0.9227	0.8233	0.8929
0.7221	0.8496	0.9160	0.8233	0.8882
0.6981	0.8440	0.9095	0.8233	0.8836

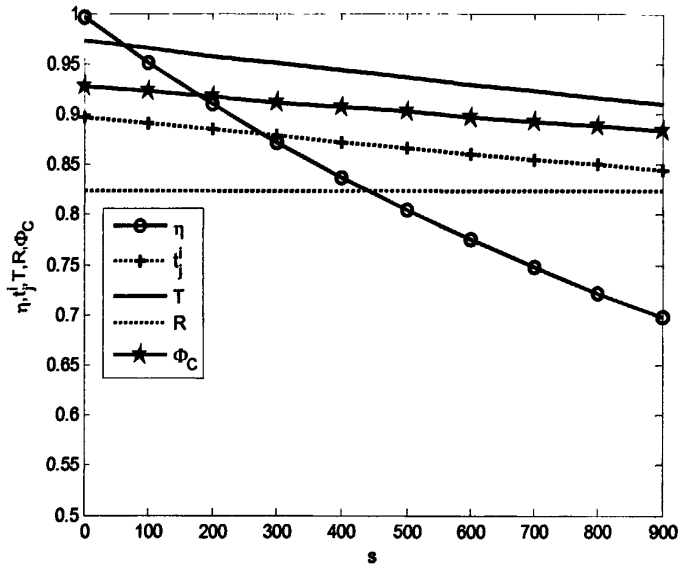


图4.6 η_i^j 对 T , R 和 Φ_C 的影响

Figure 4.6 η_i^j effects on T , R , and Φ_C

最后, 为了验证 p_i^j 对 T , R 和 Φ_C 的影响, 实验设置如下, 使深圳商家的 p_i^j 从 0.8 变化到 1, 每一步增加 0.02, 然后计算 T , R 和 Φ_C , 实验结果如表 4.6 和图 4.7 所示。

表 4.6 p_i^j 对 T , R 和 Φ_C 的影响

Table 4.6 p_i^j effect on T , R , and Φ_C

p_i^j ($j=1zsws$)	t_i^j ($j=1zsws$)	T	R	Φ_C
0.80	0.8289	0.9608	0.8233	0.9195
0.82	0.8459	0.9637	0.8233	0.9216
0.84	0.8630	0.9666	0.8233	0.9236
0.86	0.8800	0.9695	0.8233	0.9256
0.88	0.8971	0.9724	0.8233	0.9277
0.90	0.9141	0.9753	0.8233	0.9297
0.92	0.9312	0.9782	0.8233	0.9317
0.94	0.9482	0.9811	0.8233	0.9337

0.96	0.9653	0.9840	0.8233	0.9358
0.98	0.9823	0.9869	0.8233	0.9378
1	0.9994	0.9897	0.8233	0.9398

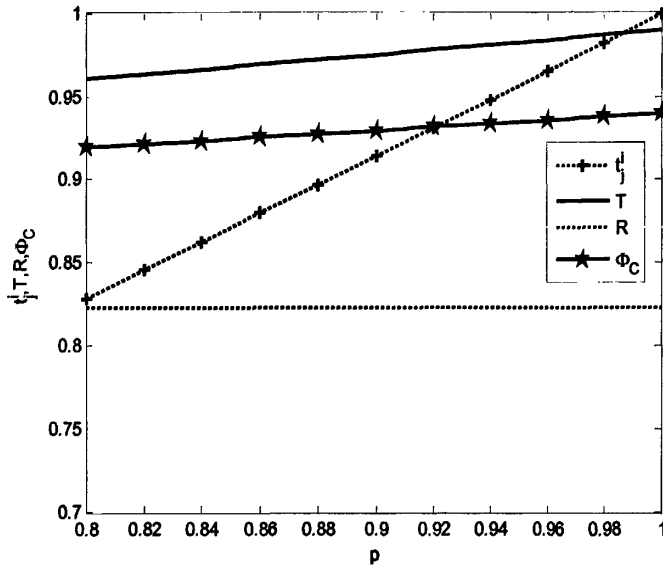


图4.7 p_i^j 对 T , R 和 Φ_C 的影响

Figure 4.7 p_i^j effects on T , R , and Φ_C

从图 4.5~4.7 可知, LCCM 能较好的反映参数 $(n_i - n_o)$, η_i^j , p_i^j 的变化对 T , R 和 Φ_C 的影响, 是合理的。

4.4.3 实验 2

在竞争性环境中应用 LCCM 同样有效, 我们选取一个稳定的联盟—中国足球超级联赛。实验目的是计算某一支队伍对整个联盟的“信任”, 此队伍的战绩越好, 则“信任”值越大。如果该队伍赢得了对手, 则 $v_i^j=1$, 如果该队伍与对手打平或输给对手, 则 $v_i^j=0$ 。此时, $n=m$, $n_i=n_o=0$, 因此, $R=0$, $\Phi_C=T$ 。我们计算北京、山东、大连、青岛四支队伍对整个联盟的“信任”。

表 4.7 t_i^j 和 T

Table 4.7 t_i^j and T

Team	ShanDong	QingDao	BeiJing	DaLian
1	0.1667	0.0769	0.2333	0.4333
2	0.5000	0.1538	0.5000	0.7692

3	0.5833	0.3636	0.5417	0.4167
4	0.5714	0.4286	0.4286	0.5000
5	0.2500	0.5000	0.5000	0.5000
6	0.4167	0.1818	0.4583	0.6250
7	0.3667	0.3077	0.3000	0.6000
8	0.3667	0.2692	0.3333	0.3000
9	0.3333	0.1667	0.3333	0.6667
10	0.5000	0.5000	0.5000	0.2500
11	0.6250	0.5000	0.5417	0.5417
12	0.5769	0.3333	0.3462	0.6923
13	0.6667	0.4167	0.5000	0.7500
14	0.5000	0.2222	0.5000	0.6667
Total	0.4544	0.2845	0.4074	0.5528

从表 4.7 和图 4.8 可知, LCCM 能够反映大连、山东、北京、青岛四支队伍的实际排名情况。如果只考虑一点的情况, 如第 10 个点, 大连是最小值, 这不符合实际情况, 而作为一个整体来考虑, 联盟信任就更为可信, 因此 LCCM 能给联盟信任提供一个好的估计。

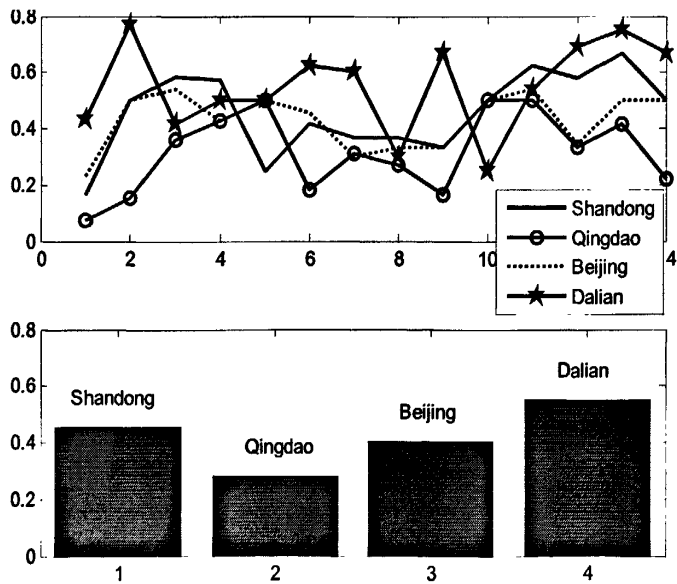


图4.8 t_i^j 和 T

Figure 4.8 t_i^j and T

4.4.4 实验 3

为了验证 LCCM 能有效地滤除噪声,我们将 LCCM 和 BRS 系统进行了对比。实验设置与 TRAVOS 相似,假设在实验 1 中添加了 100 条撒谎信息,即每一对 Agent 添加了 100 次不存在的交互历史,而成功的交互次数保持不变。实验结果如表 4.8 和图 4.9 所示。

表 4.8 BRS 和 LCCM 的计算误差比较

Table 4.8 Comparison of computational error between BRS and LCCM

Round	Lying information	Error of BRS	Error of LCCM(10^{-3})
1	100	0.0450	0.3781
2	200	0.0390	0.3783
3	300	0.0069	0.3810
4	400	0.0138	0.3962
5	500	0.0997	0.3945
6	600	0.0071	0.4134
7	700	0.2287	0.3762
8	800	0.0353	0.3784
9	900	0.0041	0.3833
10	1000	0.0060	0.3975
11	1100	0.0726	0.3777
12	1200	0.0028	0.4150
13	1300	0.0157	0.3960
14	1400	0.0004	0.4269
15	1500	0.1717	0.3766
16	1600	0.0078	0.3969
17	1700	0.0037	0.3838
18	1800	0.0042	0.4141
19	1900	0.0059	0.4136
20	2000	0.0488	0.3781

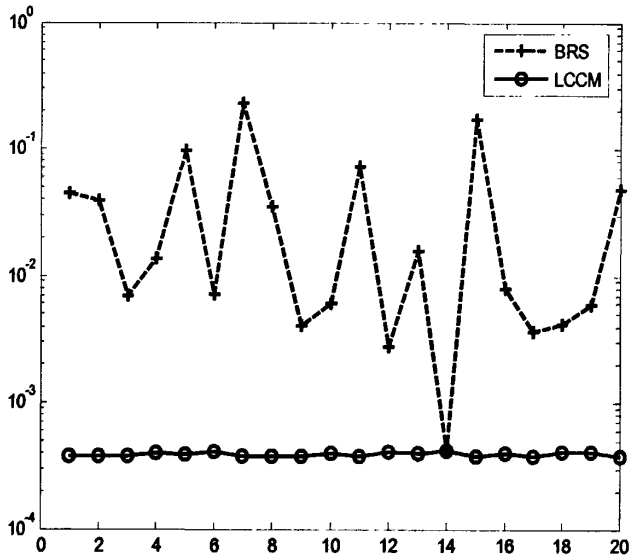


图4.9 t_i^j 的误差比较

Figure 4.9 t_i^j error contrast

结果显示，BRS 系统的 t_i^j 的平均误差是 0.0409，其联盟信任误差是 0.0045；而 LCCM 的 t_i^j 平均误差是 0.00039278，其联盟信任误差是 0.00062849。由此可知，LCCM 的误差至少比 BRS 系统少一个数量级。

如果我们改变不精确信息的设置，则得到了相似的结果，如图 4.10 所示。可知 LCCM 对不精确信息的鲁棒性比 BRS 系统更强。

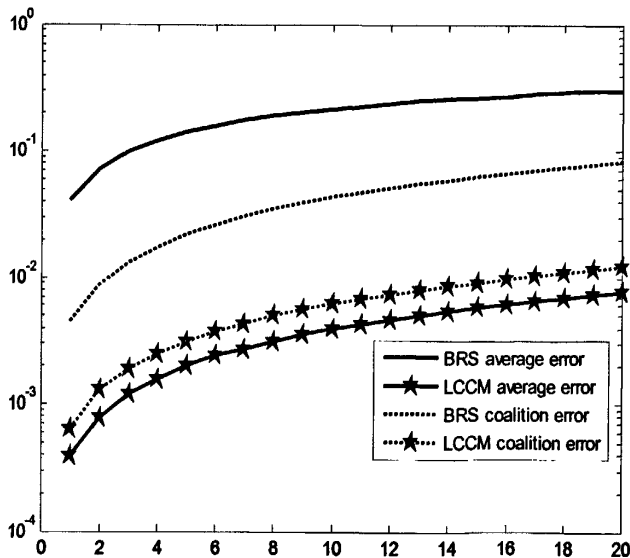


图 4.10 不同数量的不精确信息对信任的影响

Figure 4.10 Various inaccurate information effects on trust

4.5 小结

本章在现有的个体 Agent 信任和声誉研究的基础上提出了长期联盟信誉模型 LCCM, 给出了联盟信誉和联盟收益之间的关系。我们发现联盟信誉与交互成功率、个体 Agent 的偏好以及联盟的稳定性有关, 而且我们的联盟信誉计算模型可应用于竞争和合作环境中, 其对不精确信息的过滤能力也好于目前大多数的信任评估模型, 如 TRAVOS 模型。

当然, LCCM 模型中仍然有许多工作值得继续深入, 如联盟信任和联盟声誉的计算精度问题和计算可靠性等问题; 另外噪声的过滤能力需要进一步加强。

第五章 一种基于案例的 Agent 多议题协商模型

前面两章我们介绍了 Agent 的动态信任和长期联盟信誉,这在 Agent 合作求解中是重要的问题。另外,协商是合作求解必然要面临的一个基本问题,一般来说,Agent 通过协商和妥协达成一个相互有利的一致,取得双方都满意的结果,协商研究的重点和难点是不完全信息条件下多议题之间的折衷方案。

本章给出我们提出的一种基于案例的 Agent 多议题协商模型,并通过实验验证该模型优于目前的工作,包括 Fatima 等人的工作。

5.1 引言

不完全信息条件下 Agent 协商研究是一个有挑战意义的课题,目前有很多的工作已经就协商的协议、策略等展开了探讨,但是不完全信息条件下 Agent 协商研究的最优出价策略尚不能令人满意,因而学习就变得越来越重要。许多工作都是间接学习对手的偏好,然后再求得最优出价^[13,22,42-43,46],忽视了通过以往的协商经验和案例的直接学习出价方法。实际上,Agent 多议题协商研究大多假设 Agent 拥有不完全信息^[21],但是 Agent 仍然会有部分的领域共同知识和协商经验,但是很少有工作注意到这一点。不完全信息意味着 Agent 将从协商历史中学习,而基于案例的学习方法正是这一问题的合适解法,即成功的协商结束之后被保存起来,以备将来求解相似的问题。

基于案例的推理(CBR)从源案例库中选择一个与目标最相似的案例并作为求解目标的参考^[1]。通常假定与目标最相似的案例就是对目标最有用的,则相似度评估方法是核心和关键问题。不同的相似度评估方法将导致不同的检索结果、不同的自适应和不同的解。通常相似度被认为是解的效用、结构和外形等,而且相似度评估方法还要易于操作,其中 k -近邻 (k -NN) 算法最常用。然而基于 CBR 的协商却被大多数学者忽视了,只有小部分零星的工作^[85-86],这些工作还不够深入,也未见形式化的描述。

一个最优回价应该针对对手的出价,寻找一个与对手相似度最大而且自身效用最大的回价,这样不仅能够保证对手对出价的接受度最大,且能保证自身效用取得最优。针对以上问题,在 Agent 多议题协商中采用基于案例的方法产生最优回价,并得到更好的协商结果。为此,提出了基于案例和对策论的 Agent 多议题

协商模型。利用多目标优化方法中的线性功效系数法, Agent 能产生最优回价策略。

本章基于对策论和案例提出的 Agent 多议题协商模型, 改进了以往间接学习对手偏好的方式研究最优回价策略, 用直接学习 Agent 出价历史的方法得到双方 Pareto 最优出价。算法的计算复杂度是多项式级的, 当 $m < n\hat{\pi}r^3/(n+1)$ ($\gamma = \min(r, dl)$) 时, 该算法的计算复杂度低于 Fatima 等人^[21]的工作, 并通过实验验证该模型的效用和达成一致时间都优于人和 Lin 等人^[41]的方法。

5.2 问题描述

5.2.1 协商场景

定义 5.1 协商案例

$$C = \langle pb, sol_pb \rangle$$

其中, 问题(problem)向量

$$pb = [I_1, I_2, \dots, I_n],$$

是买方 Agent 提出的多议题需求向量, Agent 多议题协商不仅讨论价格, 还要考虑诸如质量、送货时间、罚金等议题。相应解(solution)向量

$$sol_pb = [I_1, I_2, \dots, I_n],$$

是卖方 Agent 提出的多议题回价向量, 是能够被买方 Agent 所接受的成功出价。源案例 C 是 Agent 以前的出价及其相应解, 而目标案例表示为

$$\langle tgt, sol_tgt \rangle,$$

其中, tgt 是当前的出价, sol_tgt 则是当前出价的回价。

定义 5.2 协商模型

$$N = \langle CB, s, b, Iss, vrg, v, u, time \rangle$$

其中, 卖方 Agent 的案例库

$$CB = \{C_1, C_2, \dots, C_m\}$$

包括所有能被买方 Agent 接受的出价。s 是卖方 Agent，b 是买方 Agent。问题向量

$$Iss = [I_1, I_2, \dots, I_n]$$

有 n 个属性（议题）需要协商。属性值域

$$vrg(I_i) = \{I_i^1, I_i^2, \dots, I_i^{k_i}\},$$

是问题的第 i 个属性（议题）的可能取值。其中， k_i 是问题的第 i 个属性的可能取值的个数。相似度值函数

$$v(I_i) = \{1, 2, \dots, k_i\},$$

计算第 i 个属性的相似度时的可能取值。效用函数

$$u(I_i) = \{u_i^1, u_i^2, \dots, u_i^{k_i}\},$$

计算问题的第 i 个属性的可能效用值。协商轮次

$$time = \{1, 2, \dots, dl\},$$

其中，dl 是协商的最后一轮。

5.2.2 Agent 信息状态

对策论一般作为不完全信息条件协商工具，在实际协商中，Agent 不仅有自身的完全信息，还包括对手的不完全信息，而获得对手信息的主要方法是通过协商历史来学习。本章以卖方 Agent 为主进行说明，买方 Agent 类似。假定不完全信息，即卖方 Agent 不能提前知道买方 Agent 的确定类型，但买方 Agent 的可能类型及其先验概率是已知的公共知识。

定义 5.3 卖方 Agent 的信息状态

$$\Theta = \langle EU^s, u_turn, u_dl, \omega^s, L^b, P^b \rangle$$

其中， EU^s 是卖方 Agent 的初始期望效用。 u_turn 是卖方 Agent 每一轮损失的

效用, 因为随着时间推移, 卖方 Agent 的效用递减。 u_dl 是卖方 Agent 在最后一轮的期望效用。 向量

$$\omega^s = [\omega^s_1, \omega^s_2, \dots, \omega^s_n]$$

表示卖方 Agent 的问题属性的权重。 r 维向量

$$L^b = [L^b_1, L^b_2, \dots, L^b_r]$$

表示卖方 Agent 拥有的买方 Agent 的不完全信息, 即买方 Agent 的可能 ω 和 τ 值。 其中, $1-\tau$ 是容许的相似度最低阈值, 即相似度计算值必须大于 $1-\tau$, 如果低于该值则 Agent 认为此报价不可接受。 向量

$$L^b_i = \langle \omega_b^i, \tau_b^i \rangle$$

表示卖方 Agent 关于买方 Agent 的第 i 种类型的信念。 其中, 向量

$$\omega_b^i = [\omega_b^i_1, \omega_b^i_2, \dots, \omega_b^i_n]$$

表示卖方 Agent 关于买方 Agent 的第 i 种类型的 n 个问题属性的 ω 值信念。 向量

$$P^b = [p^b_1, p^b_2, \dots, p^b_r]$$

表示概率分布, 其中, $p^b_i \in [0,1]$ 表示 L^b 的概率分布函数。 满足

$$\sum_{i=1}^r p^b_i = 1。$$

因此可以用概率论标准期望值方法得到以下结果。

$$\omega_i^b = \sum_{j=1}^r \omega_b^j p_j^b,$$

$$\omega^b = \{\omega_1^b, \omega_2^b, \dots, \omega_n^b\},$$

$$\tau^b = \sum_{i=1}^r \tau_b^i p_i^b。$$

5.2.3 相似度评估

基本的相似度评估算法是 CBR 中常用的 k -近邻算法, 可以通过权重计算得到目标案例和源案例之间的相似度。

定义 5.4 某属性的相似度

$$f(tgt_i, pb_i^j) = 1 - \frac{|v(tgt_i) - v(pb_i^j)|}{|tgt_i|} \quad (5.1)$$

其中, tgt 是目标问题, pb^j 是源数据库中的第 j 个案例。 $v(tgt_i) \in \{1, 2, \dots, k_i\}$ 是进行相似度计算时目标问题的第 i 个属性的值。 $v(pb_i^j) \in \{1, 2, \dots, k_i\}$ ($i=1, 2, \dots, n$, $j=1, 2, \dots, m$, n 是问题的属性数量, m 是数据库中的案例数量) 是进行相似度计算时 pb^j 的第 i 个属性的值。式中,

$$|tgt_i| = k_i$$

是目标问题第 i 个属性相似度的基。

由此我们可以得到整体相似度评估函数。

定义 5.5 Agent a 关于 tgt 和 pb 之间相似度评估函数

$$Sim(tgt, pb^j, a) = \sum_{i=1}^n f(tgt_i, pb_i^j) \omega_i^a \quad (5.2)$$

其中, $Sim(tgt, pb^j, a)$ 是 Agent a 关于 tgt 和 pb 之间相似度评估值。其中 $a = s|b$, 表示 Agent a 可以是卖方或买方 Agent。

定义 5.6 Agent a 的总效用函数

$$U(tgt, a) = \sum_{i=1}^n u(tgt_i) \omega_i^a \quad (5.3)$$

其中, $u(tgt_i)$ 是目标问题的第 i 个属性的效用值。

5.2.4 一个例子

为了演示该模型的操作并验证其可行性, 本章设计了一个网上交易场景的原

型系统。一个买家需要从一个卖家那里买一种商品或服务，有四个议题(属性)需要协商，即[价格，质量，送货时间，罚金]，假定不同的 Agent 对各个议题有不同的权重。首先从卖方 Agent 的信息状态 L^b ， P^b 入手，如表 5.1 所示。

表 5.1 买方 Agent 的权重向量 L^b 和概率分布 P^b

Table 5.1 Weight vectors L^b and P^b of buyer Agent

价格	质量	时间	罚金	容忍度	概率
0.5	0.3	0.05	0.15	0.09	0.3
0.2	0.6	0.15	0.05	0.11	0.5
0.25	0.55	0.05	0.15	0.09	0.2

由表 5.1 可知，在卖方 Agent 的信息状态中有 3 种类型的买方 Agent。卖方 Agent 对其的信念向量分别为

$$L_1^b = \langle 0.5, 0.3, 0.05, 0.15, 0.09 \rangle, L_2^b = \langle 0.2, 0.6, 0.15, 0.05, 0.11 \rangle, L_3^b = \langle 0.25, 0.55, 0.05, 0.15, 0.09 \rangle,$$

$$P_1^b = 0.3, P_2^b = 0.5, P_3^b = 0.2.$$

则 ω^b, τ^b 可以通过标准期望值方法得到。

$$\omega_1^b = 0.5 * 0.3 + 0.2 * 0.5 + 0.25 * 0.2 = 0.3,$$

同理可得

$$\omega_2^b = 0.5, \omega_3^b = 0.1, \omega_4^b = 0.1, \tau^b = 0.1.$$

因此，

$$\omega^b = [0.3, 0.5, 0.1, 0.1], \tau^b = 0.1.$$

假定目标问题（买方 Agent 提出的需求）：[300,很好,3 天,50]。解数据库与问题数据库相同，卖方 Agent 的问题数据库定义如表 5.2 所示，其相似度计算结果也列在表 5.2 中。

计算第 3 个源案例[310,很好,3 天,70]与目标问题的相似度，价格、送货时间和罚金的计算方法如定义 5.4 和 5.5 所示，如第 1 个属性（议题）价格。

$$|tgt_1| = 300, \nu(tgt_1) = 300, \nu(pb_1^3) = 310,$$

则

$$f(tgt_1, pb_1^3) = 1 - \frac{|v(tgt_1) - v(pb_1^3)|}{|tgt_1|} = 1 - \frac{|300 - 310|}{300} = 1 - \frac{1}{30} = \frac{29}{30} = 0.9667.$$

第 2 个议题质量共有 5 个选项：优秀、很好、好、一般、差，每差一级相似度差 0.2。

$$f(tgt_2, pb_2^3) = 1 - \frac{|v(tgt_2) - v(pb_2^3)|}{|tgt_2|} = 1 - \frac{|2 - 2|}{5} = 1 - 0 = 1.$$

同理可求得

$$f(tgt_3, pb_3^3) = 1, \quad f(tgt_4, pb_4^3) = 0.6.$$

因此，得到

$$Sim(tgt, pb^3, a) = \sum_{i=1}^n f(tgt_i, pb_i^3) \omega_i^a = 0.9667 * 0.3 + 1 * 0.5 + 1 * 0.1 + 0.6 * 0.1 = 0.95.$$

表 5.2 卖方 Agent 的数据库和计算结果

Table 5.2 Case base of seller Agent and computational results

价格 (\$)	质量	时间 (天)	罚金 (\$)	买方相似度
340	很好	3	60	0.94
320	一般	2	60	0.726667
310	很好	3	70	0.95
300	好	1	20	0.773333

由表 5.2 可知，案例[310, 很好, 3 天, 70, 0.95]被选中，因为它的相似度评估值最大， $Sim(tgt, pb^3, a) = 0.95$ 。所以可得 $pb = [310, \text{很好}, 3 \text{ 天}, 70]$ 。

5.3 协商协议

5.3.1 接受条件

一个出价是否被接受是基于对策论的。接受条件分为两种情况，首先来考虑最后一轮，如果对手出价的效用值大于 Agent 自身底限的效用，则卖方 Agent 就接受该出价，即

$$U(tgt, s) \geq u_dl \quad (5.4)$$

然后再考虑之前的轮次。如果对手出价的效用值大于 Agent 的期望效用，则卖方 Agent 就接受该出价。基于对策论还有一种情况需要考虑，如果卖方 Agent 拒绝了对手的出价，则协商将进入下一轮，由于存在时间折扣，则其期望效用也将相应降低。因此为了提前达成一致，避免时间和效用的浪费，卖方 Agent 应提前让步，其期望效用也相应降低，即

$$U(tgt, s) \geq EU^i - u_turn * (i+1) \quad (5.5)$$

其中， i 是协商的轮次。

总之，卖方 Agent 应该接受以上两种情况的出价。

5.3.2 最优回价的产生

一个最优回价应该针对对手的出价，寻找一个与对手的相似度最大而且自身效用最大的回价，这样不仅能够保证对手对回价的接受度最大，且能保证自身效用取得最优。

根据传统 CBR，卖方 Agent 将搜索具有最大相似度评估值的最相似案例，以便在下一个动作时重用。然而根据对策论，卖方 Agent 认为最相似的案例并不一定是买方 Agent 认为最相似的案例，即不一定能被买方 Agent 接受，因为买卖双方 Agent 有不同的权重向量，因此选择最优出价需要考虑对手的权重和容忍度。

其次，依据对策论，在完全信息条件下给出一个超出对手容忍度的回价是没有意义的，然而在不完全信息条件下，只能估计对手的偏好和容忍度，因而其值是不确定的，容忍度也就不再是确定的约束条件。为此本章改变买方容忍度，即将 τ^b 改变为 $\tau^b + \varepsilon$ ，其中， ε 为确定的合适值。总之，解案例是否被对手接受是协商成功的关键，因而最优回价是寻找解案例最优的情况。

对于 n 个议题，假设每个议题有 k 个可能值，则所有可能的议题组合是 k^n 情况，显然随着议题数量的增加议题组合呈指数增长，属于 NP Hard 问题，计算复杂度过高。然而，根据以上的分析，可行解应该是两个集合的交集：卖方可接受的最低效用议题组合的集合和买方可以容忍的相似度的议题组合的集合。只有这两个集合的交集才能保证出价可能被双方 Agent 接受，因而最优回价就是从交集中找到最优的一个，因此计算复杂度大大降低。

本章综合考虑对手权重向量和容忍度，给出卖方 Agent 的最优回价。在具体

协商过程中，首先考虑对手的容忍度，搜索所有满足买方 Agent 容忍要求的案例，组成一个可行域数据库。

$$CB^r = \{pb^j \mid \sum_{i=1}^n f(tgt_i, sol_pb^j_i) \omega_i^b \geq 1 - \tau^b - \varepsilon, sol_pb^j \in CB\} \quad (5.6)$$

可知， $CB^r \subseteq CB$ ，令 $|CB^r| = m^r$ ，则 $m^r \leq m$ ， m^r 的具体值与 ω^b, τ^b 有关。

求得 CB^r ，则可求得某些案例满足买方 Agent 相似度要求。依据 Agent 的自利性，卖方 Agent 的效用和买方 Agent 的接受度都要考虑，理想情况是二者都能取得最大值，显然这是一个多目标优化问题。本章依据多目标优化原理，采用线性功效系数法求解，即取卖方的功效为当前出价效用减去最后一轮的效用，而取买方的功效为买方的相似度，则最优选择为两者乘积最大的案例。

$$pb_{opt} = \arg \max_{sol_pb^j \in CB^r} (U(sol_pb^j, s) - u_dl) * Sim(tgt, sol_pb^j, b) \quad (5.7)$$

最后，卖方 Agent 给出 pb_{opt} 作为最优回价。

5.3.3 算法

算法 5.1 最优回价生成算法

1. $x(i,j)$ is selected by one people as an offer
2. For $i=1:m$ % m is the number of negotiation case base dimension
3. For $j=1:n$ % n is the number of negotiation issue
4. $u(i,j) = fun_intou(x(i,j));$ % let item translate into utility
5. $v(i,j) = fun_intov(x(i,j));$ % let item translate into the degree of similarity
6. End For
7. $u(i) = u(i,j) * w(j);$ % compute utility of an offer
8. $v(i) = Sim(tgt, v(i,j), b);$ % compute degree of similarity of an offer
9. End For
10. $Uv(i) = (u(i) - u_dl) * v(i);$ % the efficiency of utility
11. $[y, i] = max(Uv(i));$ % select the maximal product of utility and similarity
12. $cb_s = cb_sol(i, :);$ % give a counter-offer

性质 5.1 算法 5.1 的最优回价可实现双方 Agent 的 Pareto 最优。

证明：算法 5.1 是基于线性功效系数法求解的，其功效函数分别为卖方的效用

函数和买方的相似度函数，卖方的功效函数可以反映卖方的效用，而买方的相似度函数则可以反映买方的接受度。由多目标规划的基本定理可知，其解必为多目标优化的有效解，即卖方效用和买方接受度的最大化，即 Pareto 最优解。证毕。

性质 5.2 算法 5.1 的计算复杂度为 $O(T)$ ，其中， $T = \max(m(n+1), nr)$ ， m 为案例库中案例数量， n 为需要协商的议题数量， r 为对手类型个数。

证明：其计算由两部分组成，其一评估对手的 ω^b 和 τ^b ，其二求解最优回价。第一部分 ω^b 的计算复杂度为 $O(nr)$ ，求解 τ^b 的计算复杂度为 $O(r)$ ，其总复杂度为 $O(nr)$ 。第二部分求解每一个回价的效用所用计算次数为 $n+1$ ，其计算复杂度为 $O(n+1)$ ，求解对手相似度的计算复杂度为 $O(n+1)$ ；则计算一个回价和所有的案例 (m 个) 的相似度的计算复杂度为 $O(m(n+1))$ ，最后寻找最大值的计算复杂度为 $O(2m)$ ，则总计算机复杂度为 $O(m(n+1))$ 。两部分综合，算法 5.1 的计算复杂度为 $O(T)$ ，其中， $T = \max(m(n+1), nr)$ 。证毕。

算法 5.2 基于案例的 Agent 多议题协商算法

1. For $i=1:dl-1$ % dl is negotiation deadline turn
2. If any Agent quit negotiation, then Break, negotiation is over.
3. Else
4. If the utility of seller Agent is bigger than its expected utility then Seller Agent accepts the offer, break
5. Else seller Agent gives counter-offer
6. End If
7. If buyer Agent accepts the offer, then Break, negotiation is over.
8. Else seller Agent receives counter-offer from buyer Agent, $i++$
9. End If
10. End If
11. End For
12. If $i=dl$
13. If the utility of seller Agent is bigger than utility of deadline; then Negotiation is successful
14. Else then Negotiation failed
15. End If
16. End If

性质 5.3 算法 5.2 的计算复杂度是 $O(Tdl)$ ，其中， $T = \max(m(n+1), nr)$

证明：由性质 5.2 可知，算法 5.1 的计算复杂度为 $O(T)$ ，算法 5.2 最多总共需进行 dl 轮，每一轮次的计算复杂度为 1，则总的计算复杂度为 $O(Tdl)$ 。证毕。

5.3.4 与其它工作的比较

Faratin 等人^[22]的方法与本章相似。该工作提出基于对手出价来研究对手的偏好，基于对手出价和相似性，首先求出对自身而言与该出价有相同效用的出价集合，即对该 Agent 而言，这些出价是具有相同的效用的，是无差别的；然后从中选出该 Agent 认为的导致对手效用最高的出价，因为这样的出价被对手接受的可能性最大，并作为回价发给对手。然而该方法的缺陷之一是 Faratin 等人的方法只是学习 Agent 的前一个出价，并依据该出价给出自己的回价，并非学习一个被接受的出价。因而只是较小地增加了出价被接受的可能性，并没有从根本上解决对手接受度问题；其二完全没有顾及 Agent 自身的效用最大化，依靠对手的善心来实现自身效用最大化显然是不现实的。而且如果对手想欺骗该 Agent，则该 Agent 无能为力。而算法 5.2 能够实现对手的相似度最大值和自身效用最大值，改进了最优回价生成策略。

将算法 5.2 与 Fatima 等人的工作进行比较，可以发现两者都可以实现 Pareto 最优解，但是在案例库规模 $m < n\hat{\pi}\gamma r^3 / (n+1)$ ($\gamma = \min(r, dl)$) 情况下，算法 5.2 的计算复杂度更低。

Fatima 等人工作的计算复杂度为

$$O(n\hat{\pi}^3 t (dl - \frac{t}{2})),$$

其中， $t = \min(2r-1, dl)$ 。当 $dl \geq 2r-1$ 时， $t = 2r-1$ ，其计算复杂度为 $O(n\hat{\pi}^3 (2r-1)(dl-r+0.5))$ ，即 $O(n\hat{\pi}^4 dl)$ ；当 $dl < 2r-1$ 时， $t = dl$ ，其计算复杂度为 $O(n\hat{\pi}^3 dl^2)$ 。

分析比较可知：当 $nr > m(n+1)$ 时，算法 5.2 的计算复杂度为 $O(nrdl)$ ，明显低于 $O(n\hat{\pi}^3 dl^2)$ 和 $O(n\hat{\pi}^4 dl)$ 。

当 $m(n+1) \geq nr$ 时，算法 5.2 的计算复杂度为 $O(m(n+1)dl)$ ；则当 $m < n\hat{\pi}\gamma r^3 / (n+1)$ ($\gamma = \min(r, dl)$) 时，算法 5.2 的计算复杂度更低。

综上所述，当 $m < n\hat{\pi}\gamma r^3 / (n+1)$ ($\gamma = \min(r, dl)$) 时，算法 5.2 的计算复杂度更低。

5.4 实验

5.4.1 实验设置

实验的主要目的是验证该 Agent 能取得比人类更好的协商效用和更短的达成一致时间；其次是验证该 Agent 是否能取得更好的社会福利，即双方的效用总和更大；最后与 Lin 等人^[41]的实验效果进行对比。

本章采用 Lin 等人^[41]提出的实验环境，如表 5.3 所示。对手的可能类型是公共知识，但对手是哪种具体类型未知。协商最多持续 14 轮，如果一个协商没有在时间底线前达成一致，则得到一个冲突的结果。时间会对协商效用产生折扣效果，且每一个 Agent 都可以随时退出协商。具体协商环境是在成功面试后老板和工作候选者之间就某些具体待遇进行协商。卖方还是买方 Agent 都有三种类型：(i) 短期型，(ii) 长期型，(iii) 妥协型。

本章选择该场景理由有二：其一，当面试成功后，老板认为应聘者具备了完成任务的能力，决定录用该候选者，而应聘者也愿意接受这份工作，达成一致对双方都有利。但是就某些具体的工作条件（利益如何分配）双方需要进一步协商确定，也就是说双方有达成一致的共同利益，但是具体利益的分配还有进一步协商的必要，因而这就是一个典型的 Agent 双边多议题协商环境。其二，本章提出的协商协议完全适用于该环境，协商双方就某些具体条件的协商（如薪水\$7000，\$12000，\$20000 等）作为协商模型的 $vrg()$ ，其效用（如 3，6，8 等）作为 $u()$ ；其相似度（分别描述为 1，2，3 等）用 $v()$ 表示。

此实验背景与例 1 的区别是例 1 的价格、罚金两个议题相似度值是连续值，而此实验中是离散值，不过，这在计算时没有太大的影响。另外，例 1 的各个议题效用没有明确的给出，实际上价格为 310，并非指卖方或买方的效用就是 310，卖方的效用应该为 310 减去卖方的保留价格（卖方的保留价格意味着卖方以此价格成交没有效用可言），买方的效用等于买方的保留价格减去 310（买方的保留价格意味着买方以此价格成交则没有效用可言）。在议题的每一个选项上，买卖双方的总效用保持大体相等，因为一方效用增加一般意味着另一方效用的降低。

表 5.3 领域效用函数

Table 5.3 Domain Score Functions

输出	工作候选者	老板
	权重/效用	权重/效用

	i	ii	iii	i	ii	iii
薪水	20%	30%	15%	20%	15%	10%
\$7000	3	2	3	8	7	7
\$12000	6	6	5	6	6	6
\$20000	8	9	6	3	3	4
工作描述	15%	25%	20%	20%	30%	20%
QA	2	-2	2	4	2	3
Programmer	4	3	4	6	6	6
Team manager	5	6	6	4	3	4
Project manager	6	8	8	2	1	3
租用车	20%	5%	10%	10%	10%	10%
No car	-5	-5	-2	3	4	5
A car	5	5	2	-2	2	4
No agreement	0	0	0	0	0	0
基金	10%	5%	10%	10%	10%	10%
0%	-2	-2	-2	3	6	6
10%	3	4	3	4	4	4
20%	5	6	5	3	3	3
No agreement	0	0	0	0	0	0
升迁机会	5%	25%	35%	10%	20%	20%
Slow track	4	1	-2	3	8	6
Fast track	5	5	5	3	5	4
No agreement	0	0	0	0	0	0
工作时间	30%	10%	10%	30%	15%	30%
10 hours	3	3	4	8	8	9
9 hours	5	4	5	6	6	6
8 hours	7	5	6	3	4	3
时间效用	-8	-8	-8	-6	-6	-6
冲突效用	160	135	70	240	306	306
退出效用	150	75	80	210	150	215

5.4.2 实验方法

实验方法也与 Lin 等人^[41]的工作相似。每一个 Agent 通过选择不同可选值来构成出价，并用英语的方式传送。实验中每一个人都分别与 Agent 和其他人进行协商，但不能提前知道具体和谁进行协商，即不知道对手的具体类型。每一个人都要基于其效用函数进行协商，并尽可能地取得最好的协商结果。

实验中卖方 Agent 可接受的出价效用设为大于等于 510，即案例库中的案例效用均大于此，在具体的实验过程中由于存在时间折扣，所以最终卖方 Agent 所接受的出价可能会低于 510。由于某些议题的不同选项有相同的效用，所以导致可接受的案例数目增多。实验中卖方 Agent 的案例库有案例 192 个，即 $m=192$ ，包含了各种卖方 Agent 所能接受的案例，另外，聘用者的容忍度 $1-\tau^b-\varepsilon$ 设为 0.5。实验过程中除了主界面编程外，还分别编写了议题选项转化为效用函数，议题选项显示函数，议题选项相似度计算函数，分别进行议题效用计算，议题显示和议题相似度计算。总编程大约 1200 行。

烟台大学计算机学院大三本科生于 2008 年 5 月 28 日在计算机虚拟实验室进行了 30 组实验，每一个人的类型在实验之前临时指定，并相互保密，传送信息通过局域网和 QQ 方式。实验总用时大约 3 个小时，分别进行了人与人之间的协商，Agent（老板）与人之间的协商，以进行效果比较。图 5.1 是实验界面，每一个议题的选项在主界面的左侧，实验者从此选项选择合适的议题选项，然后按“OK”键。Agent 的回价在主界面的右侧，实验者如果同意 Agent 的回价则可以按下方的“Accept”键，中间柱体显示每一次选择时 Agent 的效用。

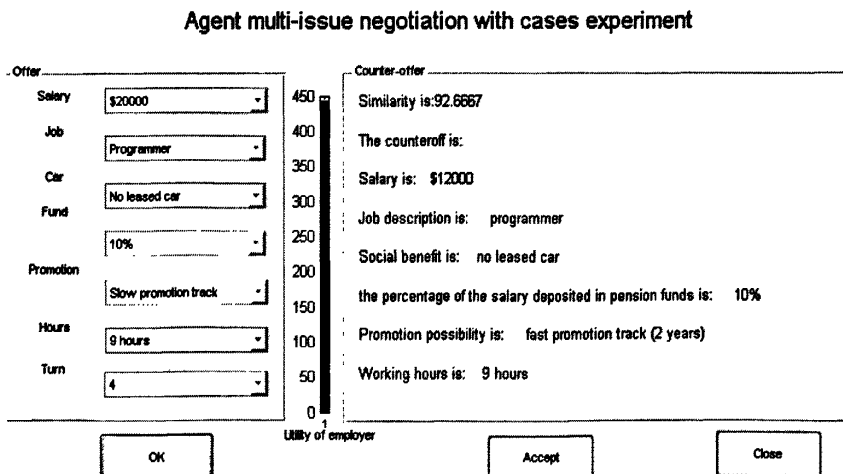


图 5.1 协商界面

Figure 5.1. Interface of negotiation

5.4.3 实验结果

从实验过程看，人与人之间的协商用时显然大于人机之间的协商。分析原因在于：其一，不同的人之间能力水平和生活背景不同，掌握协商要领的程度不同，因而会造成小误会、小失误等，浪费了较多的时间；其次，人的计算能力要远远低于计算机的计算能力，因而在相似度计算、效用计算和最优回价生成等方面，用时较多；最后，人机达成一致所用的协商轮次也要低于人人达成一致所用的协商轮次。

图 5.2 和图 5.3 显示了实验的具体数据。图 5.2 是 H_{Can} , H_{Emp} 和 Q_{Emp} 的效用比较，图 5.3 是 H_{Can} , H_{Emp} 和 Q_{Emp} 的达成一致时间比较。

表 5.4 协商结果比较

Table 5.4 Comparison of negotiation results

实验		U_Can	U_Emp	U_sum	Time
R. Lin	H_{Can} vs. H_{Emp}	309.7	410.6	720.3	
	H_{Can} vs. Q_{Emp}	342.45	437.3	779.7	
	优化	32.75	26.7	59.4	
本章	H_{Can} vs. H_{Emp}	400.4	434.9	835.3	3.4
	H_{Can} vs. Q_{Emp}	437.6	466.6	904.2	3
	优化	37.3	31.7	68.9	-0.4

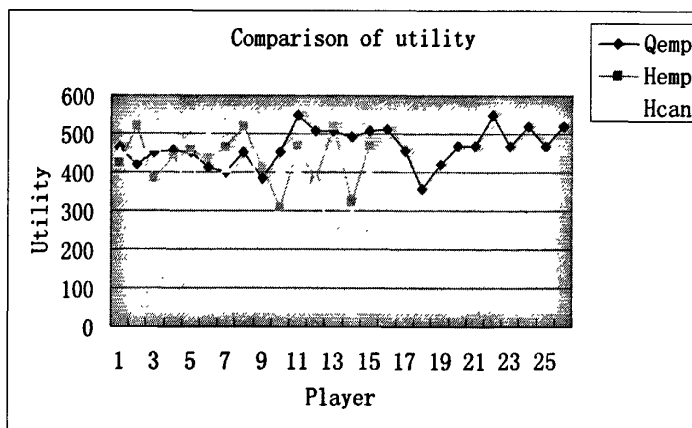


图 5.2 H_{Can} , H_{Emp} 和 Q_{Emp} 的效用比较

Figure 5.2 Comparison of experiment utility among H_{Can} , H_{Emp} and Q_{Emp}

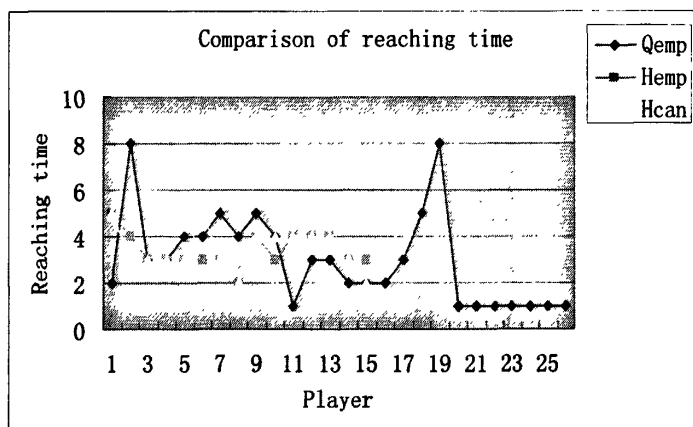
图 5.3 H_{Can} , H_{Emp} 和 Q_{Emp} 的达成一致时间比较Figure 5.3 Comparison of experiment reaching time among H_{Can} , H_{Emp} and Q_{Emp}

表 5.4 总结了实验的平均效用和平均达成一致时间, 并给出 Lin 等人的实验结果以便进行比较。 H_{Can} vs. H_{Emp} 表示工作候选者和老板都是人类的实验情况, H_{Can} vs. Q_{Emp} 表示工作候选者是人类而老板是 Agent 的实验情况, U_{Can} 表示工作候选者的平均效用, U_{Emp} 表示老板的平均效用, U_{sum} 表示工作候选者和老板的平均效用总和, $Time$ 表示协商达成一致所用的平均协商轮次。

由表 5.4 可知, 实验结果验证了我们的实验目的。具体地说, 使用本章的 Agent 能提高协商效率, 即可以取得比人类明显好的效用, 达成一致的时间明显更快。还可以得到本章的 Agent 比 Lin 等人的 Agent 更优, 因为其优化性能分别为 37.3: 32.75; 31.7: 26.7; 68.9: 59.4。

分析原因在于: 由于卖方 Agent 事先占有了大量的案例, 拥有了大量的经验和知识, 可以从中选出 Pareto 最优的出价, 这将保证卖方 Agent 的平均效用增加, 又因为考虑到对手的接受度, 所以对手更容易接受, 协商能够更提前地达成一致, 因而这将大大加快协商达成一致的进程, 并能够显著的增加双方的协商效用总合。

如果增加协商轮次和增大案例库的大小, 由性质 5.3 可知, 只是增加多项式级时间, 影响不大, 在实验过程中, 未见时间显著增加。

5.4.4 协商过程例子

(i) 工作候选者 (人) 出价:

<Salary: \$20000, Job: Team manager, social benefit: A leased car, 0% fund, Promotion possibility: slow track, Working hours: 8 hours, turn=1.>

老板 (Agent) 回应:

The counter-offer is:

<Salary: \$20000, Job: Programmer, social benefit: No leased car, 20% fund,
Promotion possibility: fast track, Working hours: 10 hours, turn=1.>

(ii) 工作候选者 (人) 出价:

<Salary: \$20000, Job: Programmer, social benefit: A leased car, 20% fund,
Promotion possibility: fast track, Working hours: 9 hours, turn=2.>

老板 (Agent) 回应:

The counter-offer is:

<Salary: \$12000, Job: Programmer, social benefit: No leased car, 20% fund,
Promotion possibility: fast track, Working hours: 9 hours, turn=2.>

(iii) 工作候选者 (人) 出价:

<Salary: \$12000, Job: Programmer, social benefit: A leased car, 20% fund,
Promotion possibility: fast track, Working hours: 9 hours, turn=3.>

老板 (Agent) 回应:

The counter-offer is:

<Salary: \$12000, Job: Programmer, social benefit: No leased car, 20% fund,
Promotion possibility: fast track, Working hours: 9 hours, turn=3.>

(iv) 工作候选者 (人) 出价:

<Salary: \$12000, Job: Programmer, social benefit: No leased car, 20% fund,
Promotion possibility: fast track, Working hours: 9 hours, turn=4.>

老板 (Agent) 回应:

I accept the offer. Negotiation is successful. The result is as follows:

<Salary: \$12000, Job: Programmer, social benefit: No leased car, 20% fund,
Promotion possibility: fast track, Working hours: 9 hours, turn=4. Utility of employer is
498.>

5.5 小结

本章提出了基于案例的 Agent 多议题协商模型, 可以基于对策论和案例产生 Pareto 最优出价。Agent 拥有的协商经验和背景知识越多, 越能取得更好的协商结果。本章的工作关注于以往大多被忽视的不完全信息条件下 Pareto 最优出价生成策略和方法, 实验结果验证了在不完全信息条件下该方法的协商效率。

基于案例的多议题协商仍然有许多工作可以继续开展。如何选择最合适的案例是其面临的一个重要课题，因此，相似度函数的选择和改进有必要进一步深化。另外，案例的重用也是一个有意义的工作，如何使过去的案例适用于现在的情况是一个值得仔细研究的内容。总体来说，未来有两个方向可进一步深入，一是可以构建贝叶斯分类算法来应对不完全信息，其次可把该工作扩展到更大规模数据库和应用环境。

第六章 一种非线性效用函数的 Agent 多资源配置协商模型

近年来, 不完全信息条件下的多资源配置一直是多 Agent 系统合作求解研究的重要问题。协商是基本的方法之一, Fatima 等人对此提出了各种协商模型和策略, 在 IJCAI'07 会议上有多篇关于资源配置的论文, 探讨了各种情况的多资源配置协商, 引起了学者的广泛关注。然而在这些研究之中, 效用函数的选择一直没有得到应有的重视, 很多学者没有明确给出效用函数, 也有的学者采用了线性效用函数, 这就导致了在计算协商效用时不够清晰和明确, 涵盖范围较小。

本章主要讨论在不完全信息条件下多 Agent 进行可分割的多资源配置协商, 将线性效用函数扩展为非线性, 达到 Pareto 最优效果。

6.1 引言

Saha^[59]、Chevaleyre^[12]、Fatima^[19-20]等人提出了 Pareto 最优的、无嫉妒的或/和社会福利(双方总效用)最优的多资源配置协商模型。然而, 目前不完全信息条件下 Agent 协商模型或没有明确给出效用函数与资源的关系, 如 Saha 等人^[59]、Chevaleyre 等人^[12]研究不可分割资源分配没有给出具体的效用函数; 或用简单的线性关系, 如 Fatima 等人^[19-20]直接将 Agent 得到的某种资源作为其效用使用。

实际上, 很多情况不符合效用函数线性的假设。效用是 Agent 的一种主观信念, 资源是客观存在, 资源对效用有激发作用, 一般情况下, 二者之间的关系是饱和和非线性的。Wooldridge^[77]指出金钱(资源)是效用的很好类似而不是效用, 并给出了效用和金钱(资源)之间的典型关系, 效用的增量呈现前快后慢的特点, 即符合边际效用递减原理。例如当某 Agent 身无分文时, 100 元对其的效用是非常大的, 而当其坐拥亿元资产时, 100 元对其的效用就接近于零。

为此, 本章将效用函数的选择由线性扩展为非线性。受神经网络的人工神经元激活函数的启发, 给出一种改进的 Sigmoid 函数表示资源和效用之间的关系^[48], 该函数符合边际效用递减原理。在此基础上, 本章提出了一种两阶段的多资源配置协商模型。在第一个阶段将每一种资源作为不可分割的对象用严格轮番法达成初步一致, 同时获得对手的偏好和激发速率信息, 然后, 在第二个阶段将资源分割并利用贪心算法达到 Pareto 最优解。

本章提出的 Agent 多资源配置协商模型不仅能够保证协商结果的 Pareto 最优,

且不需要 Agent 的历史信息；所给算法的计算复杂度为多项式级，实验结果显示算法 6.1 的 Pareto 优化效果高于 Fatima 等人的工作。本章工作可应用于许多类似于多资源分配的领域，如资源分配协商、电子商务中商品买卖协商，日常谈判如工作条件和危机冲突解决等。

6.2 问题描述

本章对 Fatima 等人^[21]的协商模型进行必要的修正和扩充。Agent a 和 b 就 $m \geq 2$ 个资源（议题）进行协商。假定这些资源是可分割的，并且这些资源之间没有相互依赖关系。本章以 Agent a 为例说明，则 Agent b 为协商对手。

定义 6.1 协商场景

$$N = \langle a, b, R, m, n, \rho \rangle$$

其中， a, b 是参与协商的两个 Agent，资源向量

$$R = [r_1, r_2, \dots, r_m]$$

表示 m 种待分配的资源，即 m 个议题集合， m 是待分配的资源数量， n 是协商的最大轮次，即 Agent 最多只能协商 n 个轮次；向量

$$\rho = [\rho_1, \rho_2, \dots, \rho_m]$$

表示议题的时间折扣率，满足 $\rho_i \in (0, 1]$ 。

定义 6.2 Agent 信息状态

$$I_a = \langle k_a, x_a, u_a, \lambda_a, C_b, P_b \rangle$$

其中， I_a 是 Agent a 的信息状态，偏好向量

$$k_a = [k_a^1, k_a^2, \dots, k_a^m]$$

表示 Agent a 对各个议题的偏好，其中，

$$k_a^i \in [0, 1], \quad \sum_{i=1}^m k_a^i = 1;$$

资源分配向量

$$x_a = [x_a^1, x_a^2, \dots, x_a^m]$$

表示 m 种资源分配给 Agent a 的份额, 其中, $x_a^i \in [0, 1]$ 表示 Agent a 得到资源 r_i 的份额。 u_a 是 Agent a 的总效用, 表示为

$$u_a = \sum_{i=1}^m u_a^i (\rho_i)^{t-1},$$

其中, t 是当前的协商轮次, u_a^i 是 Agent a 的资源 r_i 的效用, 如定义 6.3 所示, 向量

$$C_b = [C_b^1, C_b^2, \dots, C_b^l],$$

表示 Agent a 已知 Agent b 有 l 种类型, 概率向量

$$P_b = [P_b^1, P_b^2, \dots, P_b^l],$$

其中, P_b^i 表示 Agent b 是 C_b^i 的概率。

本章对人工神经元的激发 sigmoid 函数进行了改进, 给出效用和资源之间的关系如图 6.1 所示。

定义 6.3 改进的效用函数

$$u_a^i = k_a^i (1 - e^{-\lambda_a x_a^i}) \tag{6.1}$$

其中, $\lambda_a \in (0, \infty)$ 是 Agent a 的激发速率, 表示资源对效用的影响程度, 本章假定每一个 Agent 的各种资源的激发速率是相同的常量。 λ_a 越大, u_a^i 跃变的速度越快, 当 $\lambda_a \rightarrow \infty$ 时, u_a^i 是一个阶跃函数。

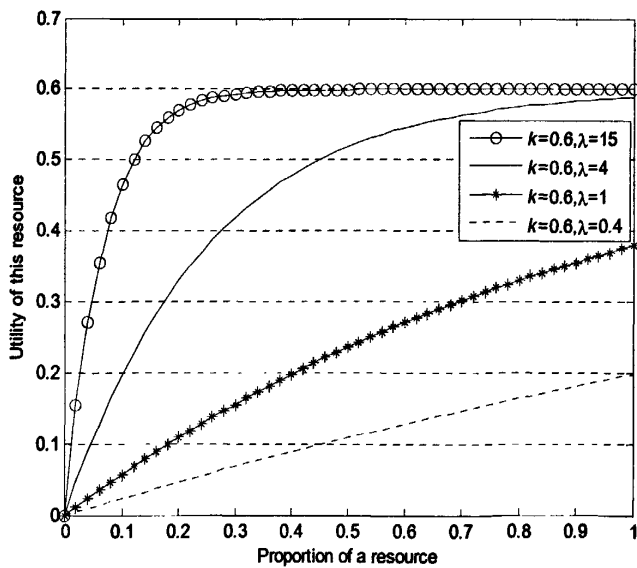


图 6.1 效用和资源的关系曲线
Figure 6.1 Relationship between utility and resource

该效用函数的选择与以前不同, 如 Fatima 等人^[21]的选择只是效用函数呈线性时的情况。实际上, 当 $\lambda_a < 1$ 时, 该效用函数 u_a^i 基本上是线性的, 因而之前 Fatima 的选择只是效用函数呈线性时的一种特例。本章的效用函数更为一般, 泛化能力也更强。另外, 注意到当 $\lambda_a < 4$ 时, 即使将资源 r_i 全部分配给 Agent a , 其效用也达不到 k_a^i , 这说明激发速率过小, 已有的资源并不能满足 Agent a 的需求。

Arrow-Pratt 度量是效用函数的一个重要特性, 定义一个效用函数 $u(x)$, Arrow-Pratt 的绝对风险厌恶度量定义为

$$\gamma(x) = -\frac{u''(x)}{u'(x)}.$$

本章所提出的效用函数遵循微观经济学的如下重要性质, 即有界、边际效用递减和恒定的绝对风险厌恶特性^[48]。

性质 6.1 $u_a^i \in [0, k_a^i]$ 是有界的。

证明: 已知 $\lambda_a > 0$, $x_a^i \geq 0$, 则当 $x_a^i = 0$ 时, u_a^i 取得最小值, 此时,

$$u_a^i = k_a^i(1 - e^0) = 0;$$

当 $x_a^i = 1$ 时, u_a^i 取得最大值, 若此时 $\lambda_a \rightarrow \inf$, 则

$$e^{-\lambda_a x_a^i} = e^{-\lambda_a} \rightarrow e^{-\inf} \rightarrow 0,$$

$$\Rightarrow u_a^i \rightarrow k_a^i.$$

证毕。

性质 6.2 u_a^i 符合边际效用递减原理。

证明: 由效用函数定义知

$$(u_a^i)' = (k_a^i(1 - e^{-\lambda_a x_a^i}))' = k_a^i \lambda_a e^{-\lambda_a x_a^i} \geq 0,$$

说明 u_a^i 是单调非递减的, 若 $k_a^i > 0$ 且 $\lambda_a > 0$, 则 u_a^i 单调递增的; 而

$$(u_a^i)'' = (k_a^i \lambda_a e^{-\lambda_a x_a^i})' = -k_a^i (\lambda_a)^2 e^{-\lambda_a x_a^i} \leq 0,$$

说明 u_a^i 的增量是单调非递增的, 若 $k_a^i > 0$ 且 $\lambda_a > 0$, 则其增量是单调递减的, 符合边际效用递减原理。证毕。

性质 6.3 u_a^i 具有恒定的绝对风险厌恶特性。

证明: 由 Arrow-Pratt 度量定义知

$$\gamma(x_a^i) = -(u_a^i)'' / (u_a^i)' = k_a^i (\lambda_a)^2 e^{-\lambda_a x_a^i} / k_a^i \lambda_a e^{-\lambda_a x_a^i} = \lambda_a \geq 0,$$

因为 γ 不随 x_a^i 的变化而变化, 是恒定的, 因而 u_a^i 具有恒定的绝对风险厌恶特性。

证毕。

6.3 两阶段的协商模型

我们将协商过程分为两个阶段，第一个阶段有两个目的，将每一种资源看成不可分割的整体的情况下，采用严格轮番法由两个 Agent 依次选择各种资源，以达成初步一致；同时，可通过两个规则推理获得对手的偏好信息；然后在第二个阶段将各种资源分割并进行折衷交换，实现 Pareto 最优。

6.3.1 第一阶段

本章在第一个阶段采用严格轮番法^[59](strict Alteration)，即由两个 Agent 轮番选择多种资源，每一种被选资源全部分配给该 Agent，则每一个 Agent 基于自利性将选择当前资源中自身偏好最高的资源。

严格轮番法^[59]：

1. 任选一个 Agent 标记为 s ，标记待协商的资源集为 G ，初始条件下 $G = R$ 。
2. s 从剩余的资源集中选择资源 $r \in G$ ， r 被配置给 s 。
3. 将另外一个 Agent 标记为 s ，并将 G 更新为 $G - \{r\}$ ，如果 $|G| \geq 1$ ，则转到 2，否则停止。

由严格轮番法，若 Agent a 首先选择，设 m 为偶数，则将形成如下的序列

$$r_a^1, r_a^2, \dots, r_a^{m/2},$$

$$r_b^1, r_b^2, \dots, r_b^{m/2}。$$

若 m 为奇数，再多一个 $r_a^{(m+1)/2}$ 。由此，两个 Agent 可以在第一阶段达成初步的一致，通过对各种资源进行划分来实现初步的资源配置。同时，这种初步的一致还是第二阶段的优化算法的基础。

通过对上述序列的分析，我们还可以得到关于 Agent b 的偏好的规则，有利于 Agent a 推理得到 Agent b 的类型。

Agent b 的类型推理规则

- (i) r_a^i 可能是 Agent b 的第 i 高的偏好到第 m 高偏好的资源；
- (ii) r_b^i 可能是 Agent b 的第 i 高的偏好到第 $2i$ 高偏好的资源。

对规则 i，因为 Agent b 的第 $i-1$ 次选择总是在 Agent a 的第 i 次选择之前，所以 Agent b 总会提前选择第 i 高偏好的资源，即 Agent a 不可能得到比 Agent b 的第 i 更

高偏好的资源。而在最坏情况下, r_a^i 可能是 Agent b 的第 m 高偏好。

例如: Agent a 在第 1 轮可以选择第 1 到第 m 高 Agent b 的偏好的资源, Agent b 则选择剩余资源中的排序最高的。因此, 在第 1 轮中 Agent b 的第 1 高的资源必然被选中了, 即使 Agent a 没有选择第 1 高 Agent b 的偏好的资源, 那么 Agent b 必然会选择它。在第 2 轮, Agent a 只能选择第 2 高到第 m 高 Agent b 的偏好的资源。

对规则 ii, 因为 Agent a 总是在 Agent b 的选择之前, 则 Agent a 是否选择了 Agent b 所期望的资源是关键问题。在最好情况下, 只要 Agent a 只要没有提前选择了该资源, 则 Agent b 总能选择到期望的资源, 即 r_b^i 可能是 Agent b 的第 i 高的偏好资源, 在这种情况下, Agent a 选择了第 $m/2$ 到 m 高偏好的资源; 在最坏情况下, Agent a 总是提前选择了该资源, 则 Agent b 总不能选择到期望的资源, 即 r_b^i 可能是 Agent b 的第 $2i$ 高偏好资源。

例 6.1 设 Agent a 和 b 之间有 4 种需分配的资源

$$R = \{r_1, r_2, r_3, r_4\},$$

Agent b 有 3 种可能的类型, 其偏好和 λ_b 分别是

$$k_b^{C_1} = \langle 0.2, 0.3, 0.15, 0.35 \rangle, \lambda_b^{C_1} = 2.5, k_b^{C_2} = \langle 0.4, 0.25, 0.15, 0.2 \rangle, \lambda_b^{C_2} = 2,$$

$$k_b^{C_3} = \langle 0.1, 0.4, 0.3, 0.2 \rangle, \lambda_b^{C_3} = 3, P_b = \langle 0.5, 0.3, 0.2 \rangle,$$

其中, $k_b^{C_i}$ 表示当 Agent b 为 C_i 时的偏好向量, $\lambda_b^{C_i}$ 表示当 Agent b 为 C_i 时的激发速率。Agent a 为类型 2, 即

$$k_a = \langle 0.4, 0.25, 0.15, 0.2 \rangle, \lambda_a = 2.$$

若用严格轮番法得到的序列为

$$r_a^1 = r_1, r_b^1 = r_4, r_a^2 = r_2, r_b^2 = r_3,$$

可知 r_4 排序第 1 或第 2 高的只有 $k_b^{C_1}$, 因而 Agent b 必然为 C_1 类型, Agent b 得到资源 r_4, r_3 。此推理方法在 Agent b 的可能类型不多的情况下是可行的。

如果 Agent b 先选择资源, 得到的序列为

$$r_b^1 = r_4, r_a^1 = r_1, r_b^2 = r_2, r_a^2 = r_3,$$

仍然可知 r_4 排序第 1 或第 2 高的只有 $k_b^{C_1}$, 因而 Agent b 必然为 C_1 类型, Agent b 得到资源 r_4, r_2 。

总之, 第一阶段不仅实现了资源初步分配, 而且也显示了 Agent 的偏好信息和激发速率。如果 Agent b 的类型只有一种可能的情况, 根据已知的不完全信息利

用排除法进行推理，则 Agent b 必然是该类型；如果 Agent b 的类型仍然有多种可能，可取其中的先验概率最高的类型。

6.3.2 第二阶段

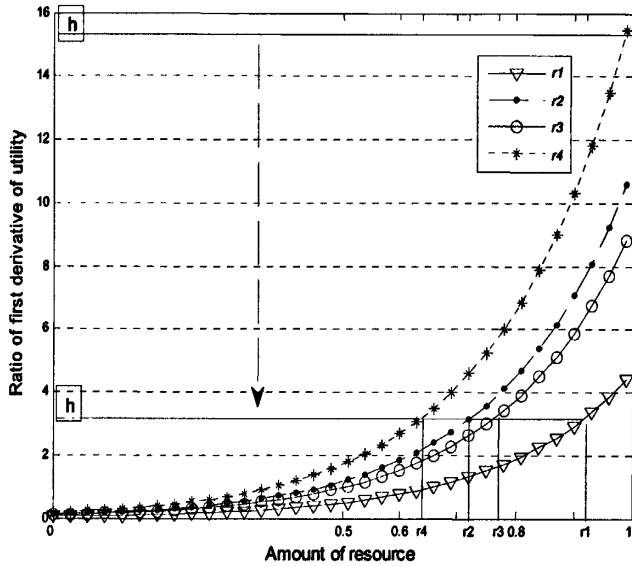


图 6.2 Agent b 和 a 关于效用函数的一阶导数的比值曲线
Figure 6.2 Ratio of first derivative of Agent b 's utility to that of Agent a

经过第一阶段之后，该模型可以得到一个初步的一致，并能推理得到完全信息。在第二阶段，该模型将利用贪心算法得到 Pareto 最优解。

令 κ_i 表示 Agent b 和 Agent a 的效用函数的一阶导数之比。

$$\kappa_i = \frac{(u_b^i)'}{(u_a^i)'} \quad (6.2)$$

定理 6.1 线性效用函数的偏好排序即为 κ_i 的排序。

证明：Fatima 等人^[21]利用了贪心算法实现 Pareto 最优。类似于背包问题，贪心算法使用 k_b^i/k_a^i 的排序，实际上就是 κ_i 的排序，因为

$$(u_a^i)' = (k_a^i x_a^i)' = k_a^i,$$

则

$$\kappa_i = \frac{(u_b^i)'}{(u_a^i)'} = k_b^i/k_a^i.$$

证毕。

由于已知对手的类型，则对手的偏好的激发速率都是已知的，因而可以用式 6.3 所示的优化方法求得 Pareto 最优解。

$$\begin{aligned} \max \quad & \sum_{i=1}^m u_a^i \\ \text{s.t.} \quad & \sum_{i=1}^m u_b^i = U_b^i \end{aligned} \quad (6.3)$$

其中， U_b^i 是 Agent b 在优化之前的初始总效用。

对 Agent a 的约束条件是其必须先满足 Agent b 的效用需求，则 Agent a 必然先选择对手效用增长率与自身效用增长率的比值最大的资源给对手，即 κ_i 最大的资源，直至满足对手的效用要求，然后将剩余的资源全部据为己有，实现效用最大化。这样既能最快满足 Agent b 的效用需求又能保留最大的效用给 Agent a 。

假设 6.1 对于资源 r_i ，若 Agent a 所分得的数量是 x_a^i ，则 Agent b 所分得的数量是 $(1-x_a^i)$ 。Agent b 在 $(1-x_a^i)$ 点和 Agent a 在 x_a^i 点的一阶导数的比值越大，则 Agent a 越早的将在 $(1-x_a^i)$ 点的资源 r_i 分配给 Agent b ，以尽快地满足 Agent b 的效用需求，然后将剩余的资源全部留给 Agent a 自己。

本章提出的效用函数的 κ_i 为

$$\kappa_i = \frac{(k_b^i(1-e^{-\lambda_b x_b^i}))'}{(k_a^i(1-e^{-\lambda_a x_a^i}))'} = \frac{k_b^i \lambda_b e^{-\lambda_b x_b^i}}{k_a^i \lambda_a e^{-\lambda_a x_a^i}} = \frac{k_b^i \lambda_b e^{-\lambda_b(1-x_a^i)}}{k_a^i \lambda_a e^{-\lambda_a x_a^i}} = \frac{k_b^i \lambda_b}{k_a^i \lambda_a e^{\lambda_b}} e^{(\lambda_a + \lambda_b)x_a^i} \quad (6.4)$$

依据贪心算法，Agent a 应该始终选择在点 x_a^i 处有最大 κ_i 的资源，而不管 k_b^i/k_a^i 是多少。实际上，该资源的选择依赖于点 x_a^i ，同时也依赖于 $k_a^i, k_b^i, \lambda_a, \lambda_b$ 等，总之决定性的因素是 κ_i 。

基于以上解释，具体方法是在图 6.2 中从上向下画水平直线 h ， h 与各条资源曲线的交点的横坐标 x_a^i 即为分配给 Agent a 的该资源的比例，剩余的资源 $1-x_a^i$ 分配给 Agent b ，其纵坐标即为 κ_i 。该方法将从 κ_i 的最大值开始，不断地向下移动，直到分配给 Agent b 的资源的效用总和等于 Agent b 的初始效用，将剩余的资源全部留给 Agent a ，这样就实现了 Pareto 最优。

给出的算法 6.1 如下所示，其中， ω 是每一种资源被分割的份数， χ 是 κ_i 的初始最大值， i 是 κ_i 递减的步长， δ 是 κ_i 误差的容许限， $u_a^p(i)$ 是第 i 种资源分配给 Agent a 后的优化效用， $u_b^p(i)$ 是第 i 种资源分配给 Agent b 后的优化效用。

算法 6.1 最优回价生成算法

1. $u_b^P(i) := 0, (i=1,2,\dots,m)$;
2. While $\sum_{i=1}^m u_b^P(i) < u_b^I$ //满足 Agent b 的初始效用要求
3. For $i=1:m$ // m 种资源
4. $j := \omega$; //每一种资源被分成 ω 份
5. While $(abs(\kappa_i(j) - \chi) > \delta)$ // δ 为 κ_i 的计算误差限
6. $j := j-1$;
7. End While
8. $x_a(i) := j/\omega$; $u_a^P(i) := u_a^I(j)$; $u_b^P(i) := u_b^I(j)$;
9. End For
10. $\chi := \chi - \iota$;
11. End While
12. $U_a^P := \sum_{i=1}^m u_a^P(i)$; $U_b^P := \sum_{i=1}^m u_b^P(i)$;

定理 6.2 算法 6.1 可得到 Pareto 最优解。

证明：由于算法 6.1 采用了贪心算法，因而在不降低 Agent b 的效用前提下，Agent a 的效用是最优的，因而算法 6.1 实现了 Pareto 最优。

定理 6.3 算法 6.1 在最坏情况下的计算复杂度是 $O(\omega m \eta)$ ，其中， $\eta = \chi/\iota$ ， χ 是 κ_i 的设定初始值， ι 是其步长。

证明：在最坏情况下，将所有的资源全部分配给 Agent b ，内嵌套的 While 语句将执行 ω 次，for 语句将执行 m 次，外嵌套的 While 语句将执行 η 次，则总计算复杂度是 $O(\omega m \eta)$ 。证毕。

算法 6.2 多资源配置协商算法

1. 把第一阶段的结果作为优化前的初始条件，任意选择其中的一个 Agent 作为 a ，另外一个 Agent 作为 b 。
2. If 第一个阶段的结果能够满足 b 的效用要求，则 b 将该结果作为首次出价发送给 a 。
3. Else b 依据自己的偏好和效用函数给出能够满足 b 效用要求的出价，End If。
4. a 收到 b 的出价，利用算法 6.1 计算 a 的效用。
5. If 算法 6.1 的计算结果能够满足 a 的效用要求，则 a 将其作为回价发送给 b ， b 应该接受该回价，因为它能够满足 b 的效用要求，协商成功结束；Else a 依据自己的偏好和效用函数给出能够满足自身效用要求的出价，End If。
6. If 该出价能满足 b 的效用要求，则 b 接受该回价，协商成功结束；Else 回到 3，End If。

6.4 实验

6.4.1 实验结果与分析

回顾例 6.1, 由严格轮番法可知 Agent b 在第一阶段分配得到资源 r_4 和 r_3 的全部, 其初始总效用为 0.45896, Agent a 得到资源 r_1 和 r_2 的全部, 其初始总效用为 0.56203。经过第二阶段的优化, 优化后 Agent a 得到资源 r_1 的 92%, 资源 r_2 的 72%, 资源 r_3 的 77%, 资源 r_4 的 64%, 如图 6.2 所示。则优化后, Agent b 的总效用为 0.46057, Agent a 的总效用为 0.78948。可以看出, Agent a 的总效用得到了明显的优化, 且 Agent b 的总效用也有较小的优化。Agent a 的总效用优化效果为 40.5% (0.78948: 0.56203), Agent b 的总效用优化效果为 0.35% (0.46057: 0.45896)。其结果如表 6.1 所示。

表 6.1 Pareto 最优解计算结果-- a 先出价

	Agent a 's utility	Agent b 's utility	x_a
Initial sum	0.56203	0.45896	1,1,0,0
r_1	0.33647	0.036254	0.92
r_2	0.19077	0.15102	0.72
r_3	0.11784	0.065594	0.77
r_4	0.14439	0.2077	0.64
Optimal sum	0.78948	0.46057	

如果换作 Agent b 先选择资源, 则其结果如表 6.2 所示。可以看到 Agent a 的总效用优化效果为 53.8% (0.73155: 0.47557), Agent b 的总效用优化效果为 0.9% (0.60193: 0.59664)。

表 6.2 Pareto 最优解计算结果-- b 先出价

	Agent a 's utility	Agent b 's utility	x_a
Initial sum	0.47557	0.59664	1,0,1,0
r_1	0.31924	0.078694	0.8
r_2	0.1747	0.18964	0.6
r_3	0.10829	0.089015	0.64
r_4	0.12931	0.24458	0.52
Optimal sum	0.73155	0.60193	

另外, 发现 δ 的大小对结果没有影响, 即当 $\delta = 0.001, 0.005, 0.01, 0.02, 0.05, 0.1$ 时, 实验结果无变化。

6.4.2 不同类型Agent的实验

当 Agent a 和 Agent b 为不同类型时,其实验结果如表 6.3 和图 6.3 所示。其中, $a=2,b=1(a)$ 表示 Agent a 为类型 2, Agent b 为类型 1,且 Agent a 先选择资源, Agent b 后选择资源的情况。 U_a^i 表示 Agent a 的优化前初始效用总和, U_b^i 表示 Agent b 的优化前初始效用总和, U_a^o 表示 Agent a 的优化后效用总和, U_b^o 表示 Agent b 的优化后效用总和。

表 6.3 不同类型 Agent 的影响
Table 6.3 Computational results of optimal solution

Utility	$a=2,b=1(a)$	$a=2,b=1(b)$	$a=2,b=2(a)$	$a=2,b=2(b)$	$a=2,b=3(a)$	$a=2,b=3(b)$
U_a^i	0.56203	0.47557	0.5188	0.34587	0.5188	0.5188
U_a^o	0.78948	0.73155	0.78986	0.71635	0.75095	0.75095
U_b^i	0.45896	0.59664	0.34587	0.5188	0.66515	0.66515
U_b^o	0.46057	0.60193	0.35596	0.52289	0.66642	0.66642
Utility	$a=1,b=1(a)$	$a=1,b=1(b)$	$a=1,b=2(a)$	$a=1,b=2(b)$	$a=1,b=3(a)$	$a=1,b=3(b)$
U_a^i	0.50485	0.41306	0.59664	0.45896	0.50485	0.50485
U_a^o	0.85773	0.83052	0.82686	0.78318	0.80764	0.80764
U_b^i	0.41306	0.50485	0.47557	0.56203	0.66515	0.66515
U_b^o	0.42305	0.51568	0.47744	0.56352	0.66737	0.66737
Utility	$a=3,b=1(a)$	$a=3,b=1(b)$	$a=3,b=2(a)$	$a=3,b=2(b)$	$a=3,b=3(a)$	$a=3,b=3(b)$
U_a^i	0.66515	0.66515	0.66515	0.66515	0.57013	0.38009
U_a^o	0.89248	0.89248	0.88881	0.88881	0.91954	0.88116
U_b^i	0.50485	0.50485	0.55075	0.55075	0.38009	0.57013
U_b^o	0.50699	0.50699	0.55348	0.55348	0.38122	0.58105

我们共进行了 18 种情况的实验。实验结果表明, Agent a 的效用优化率平均为 59.56%, Agent b 的效用优化率平均为 0.85%, 两个 Agent 的效用优化率平均为 28.95%。

还可以注意到, Agent a 和 Agent b 的效用总和一般都大于 1。这是由于参数 λ 足够大, 导致激发率足够高, 因而能够通过折衷实现双赢。

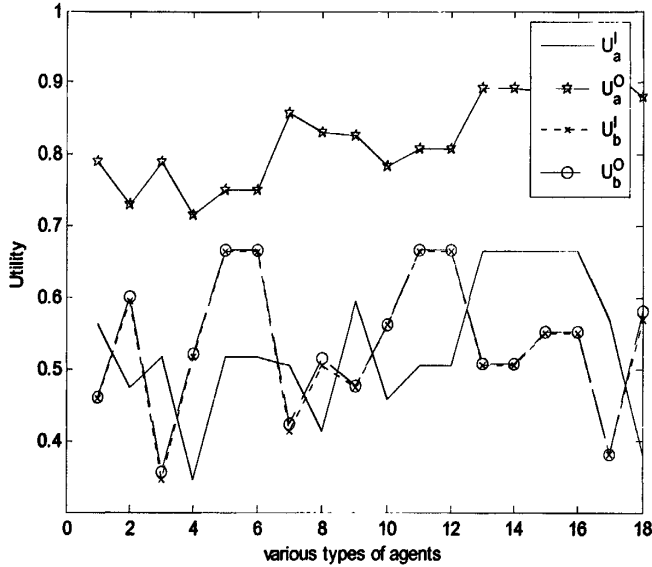


图 6.3 18 种情况的效用优化率

Figure 6.3 Optimal performance of utility in 18 cases

6.4.3 与Fatima等人工作的比较

本章与 Fatima 等人在贪心算法的使用上不同。Fatima 等人给出的效用函数是线性的，所给的贪心算法^[21]是 Agent a 选择对手与自身偏好的比值最大的资源分配给 Agent b 。而本章中给出的效用函数是非线性的，Agent a 则选择对手单位效用增量与自身单位效用增量的最大的资源给对手，直至满足对手的初始效用要求，然后将剩余的资源全部据为己有，实现效用最大化。Fatima 等人的贪心算法的前提是 Agent a 选择 k_b^i/k_a^i 排序，由定理 6.1 可知实际上是选择效用函数的一阶导数比值，而本章中选择的是 κ_i 排序，则选择资源时更复杂，因为不同的资源数量导致 κ_i 不同。实际上，本章的选择是从根本上的选择，因为增量的比值要比偏好的比值更能反映效用的作用。

在 18 种情况下，该协商模型的 Agent a 的效用优化率平均能够达到 59.56% (0.822555:0.529661)，而 Fatima 等人的工作中其优化率平均只能达到 0.55% (0.583611:0.580556)。详细比较结果如图 6.4 所示，其中， U_a^I 表示采用本模型 Agent a 的优化前初始效用总和， U_a^O 表示采用本模型 Agent a 的优化后效用总和， U_{af}^I 表示 Fatima 等人工作中的 Agent a 的优化前初始效用总和， U_{af}^O 表示 Fatima 等人工作中 Agent a 的优化后效用总和。

优化率的差别在于：Fatima 等人工作中，采用了线性的效用函数，这在实际中并不是普遍现象，也不能够很好的进行折衷，因而难以达到效用的最优化。

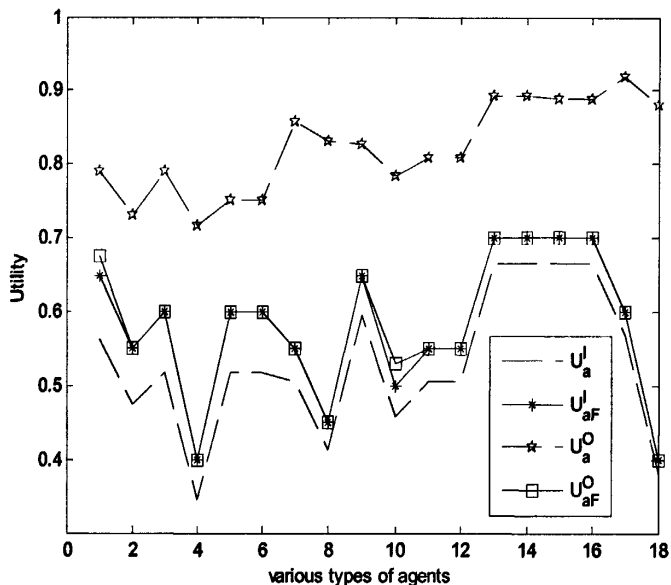


图 6.4 与 Fatima 工作的优化率的比较

Figure 6.4 Optimal performance comparison with the work of Fatima et al.

6.5 小结

本章提出的两阶段协商模型能够在第一阶段达成初步一致，并能利用两条规则由不完全信息推理获得完全信息。在应用贪心算法做多资源折衷交换时，Agent 总会将对手的效用增长率比自身效用增长率最大的资源分配给对手，直至满足对手的初始需求，然后将剩余资源据为己有。Fatima 等人的工作中效用和资源之间是线性关系，因而资源的效用增长率即为偏好，该比值是恒定的常数。而在本章中由于效用函数是非线性的，则该比值在资源数量不同时取值不同，是一个随资源数量变化的变量，因而会引起较高的计算复杂度。

本章给出各个议题的折衷方法，增加双方 Agent 的效用，实现了 Pareto 最优解，该方法的优化率高于 Fatima 等人的贪心算法。也可以推广到其它一般情况。

将来的工作可以关注于如何在时间限制条件下进行效率更高的协商，协商的策略将有何变化，以及协商的总效用如何计算等。

第七章 结束语

合作求解是 MAS 研究的核心问题之一,信任是合作求解的前提和基础,协商是合作求解的基本方法。目前 Agent 信任和协商都有大量的工作,是多 Agent 系统研究的热点。多 Agent 系统由不同的个体进行设计和实现,各个 Agent 具有自身的目标,因此他们之间可能无法形成共同的目标。多 Agent 系统中 Agent 之间的相遇与软件工程的对象不同,Agent 之间非被动地被利用,更类似于博弈,因而 Agent 一般会采用对策论的方式来选择动作。

从 Agent 技术提出开始,有学者就认为“智能 Agent 是 99%的计算机科学加 1%的人工智能”。然而到目前为止,大多数人仍然认同将多 Agent 系统归类为人工智能。其原因在于 Agent 本身仍然需要智能!除了合理的组织结构、优化的模型和适当的策略之外,智能必不可少,也显而易见。因此从人工智能尤其是机器学习和数据挖掘中吸取成熟的技术为多 Agent 系统所用,可能是一条可行的捷径。

本论文研究了多 Agent 合作求解中的信任和协商问题。提出了一种利用信任变化率计算的动态交互信任预测和异常行为检测模型,将静态信任研究引向动态。提出了一种 Agent 长期联盟信誉模型,将长期联盟的收益与联盟信誉紧密联系起来。提出了一种基于案例的 Agent 多议题协商模型,该模型利用以往协商的成功案例,快速地实现了 Pareto 最优。提出了一种不完全信息条件下的基于非线性效用函数的 Agent 多资源配置协商模型,可以优化回价生成策略,大大提高协商的效率。

7.1 本文工作的总结

(1) 在分析和研究了现有的信任评估模型的基础上,为了克服目前的平均成功率方法普遍存在的交互信任表示过于简单、预测精度较低的固有缺陷,按时间分段研究 Agent 交互信任,提出了 Agent 动态交互信任预测模型。该模型能够有效地预测 Agent 信任的变化趋势,并能对异常行为进行检测。实验结果表明该模型的计算量小于 TRAVOS 模型,计算误差也小。由于该模型考虑了信任变化的趋势,能为未来交互提供预测,其合理的置信度计算模型则能够滤除噪声、防止恶意欺骗。

(2) 目前联盟收益的工作多基于有形收益的概念,未涉及无形收益。实际上长

期联盟收益应包括有形收益和无形收益两部分。本论文关注了长期联盟形成中的无形收益,并给出了长期联盟的收益与联盟信誉之间的关系,并作为 Agent 选择联盟的依据;建立了长期联盟信誉模型 LCCM 并以此评估联盟信誉,将联盟信誉分为 Agent 对联盟的信任值和对该联盟的评价值,实验结果验证了该方法的可行性和正确性。

(3) 在多议题协商中,不完全信息条件下 Agent 多议题协商研究的最优回价策略尚不能令人满意,许多工作都是间接学习对手的偏好然后再求得最优回价,忽视了通过以往的协商经验和案例的直接学习方法。Agent 一般拥有或多或少的协商经验和领域知识,这将帮助它们获得更好的协商结果。这启发我们用基于案例的方法直接得到最优回价,为此提出了不完全信息条件下基于案例和对策论的 Agent 多议题 Pareto 最优协商模型。算法的计算复杂度为多项式级,且一般情况下低于 Fatima 工作的计算复杂度。改进了 Fatima 等人的工作。

(4) 不完全信息条件下的多议题协商是一个热点课题。目前大多数工作使用了线性的效用函数。针对该问题,本文将效用函数由线性扩展为非线性,受神经网络的人工神经元函数启发,提出了一种改进的 Sigmoid 函数表示效用和资源之间的关系。在此基础上,提出了一种两阶段的多资源配置协商模型。该模型不仅能处理不完全信息,而且能够得到 Pareto 最优解。实验结果表明,该协商模型能大大提高多议题协商的效率。

7.2 进一步研究的展望

本论文研究了多 Agent 合作求解中的信任和协商问题,就动态信任、长期联盟信任、基于案例的协商和非线性效用函数的协商进行了深入的研究。下一步的工作是就信任的推理问题开展深入的研究,以认知逻辑和模态逻辑为工具进行信任的形式化推理,对多议题协商问题需要结合机器学习和数据挖掘等方法提供一些有效、快速、最优的协商模型和策略,提高协商的效率。同时为信任和协商在电子商务中的应用开辟一条新的道路。具体地说,未来我们准备进行以下工作,提高 Agent 信任和协商技术的应用范围。

(1) 信任的逻辑推理。目前的研究很多都采用概率论的方法,缺少严谨的逻辑推理。针对这些问题,我们将尝试用认知逻辑和模态逻辑等方法研究信任、信念、承诺、诚实等之间的关系,找到彼此之间的联系。信任不是一个孤立的变量,Agent 的承诺和行为的结果对信任有很大的影响,另外,Agent 诚实度也会影响信任关系,这几者之间的关系应该可以用逻辑的方式进行表达和推导。

(2) 更精确的动态信任计算。尽管本文提出了基于信任变化率的动态信任预测模型，但是其预测精度仍然可以继续提高。最直接的方法是采用二阶或更高阶的导数方法来计算动态信任，但这必然导致计算量增加的问题，如何得到一个可接受的折衷方案是将来研究的方向。另外，也可以尝试采用其它不同的方法来计算动态信任，例如递推或者其它方法。

(3) 基于案例的多议题协商的改进。目前提出的基于案例的协商方法从原理上看比较简单，该方法也比较成熟，但缺乏足够的新意。将来的工作可以关注于针对多议题协商的相似度函数的重新定义、案例重复利用的新方法等。多议题协商的案例的相似度度量方法与其他 CBR 应该有所区别，因为其案例的定义就不同，因此一个新的相似度度量方法就有必要进行详细研究。案例的重复利用也是协商的重要内容，根据协商的场景、协商的对手、协商的策略等等因素，案例的重复利用必然要自适应的修改。

(4) 最优协商策略的研究。目前，所有的协商模型都只能实现 Pareto 最优的协商结果，很多学者也认为只要实现了此结果就认为完美了。实际上，Pareto 最优的结果并不能保证较好的效率，尤其是对双方而言，此结果距离多目标最优还有差距。在下一步的工作中，将致力于如何有效地提高双方的协商效率，实现协商双方都能得到最优的结果。

参 考 文 献

- [1] Aamodt A., Plaza E.. Case-based Reasoning: Foundational Issues, Methodological Variations, and System Approaches[J]. AICOM, 1994, 7 (1): 39-59.
- [2] Agotnes T., Hoek W. v. d., Wooldridge M.. On the Logic of Coalitional Games[C]. (AAMAS'06), Hakodate, Japan, May 2006.
- [3] Agotnes T., Hoek W. v. d., Wooldridge M.. Temporal Qualitative Coalitional Games[C]. (AAMAS'06), Hakodate, Japan, May 2006.
- [4] Agotnes T., Hoek W. v. d., Wooldridge M.. Quantified Coalition Logic[C]. IJCAI'07, Hyderabad, India, January 2007.
- [5] An B., Miao C., Shen Z., Market Based Resource Allocation with Incomplete Information [C], IJCAI'07, India, 2007, pp. 1193-1198.
- [6] Ashri R., Ramchurn S. D., Sabater J., Luck M., Jennings N. R., Trust Evaluation Through Relationship Analysis[C], AAMAS'05, Utrecht, Netherlands, July, 2005.
- [7] Beth T., Borchering M., Klein B.. Valuation of trust in open network[C]. In: Gollmann D, ed. Proceedings of the European Symposium on Research in Security (ESORICS). Brighton: Springer-Verlag, 1994. 3~18.
- [8] Blankenburg B., Dash R. K., Ramchurn S. D., Klusch M., Jennings N. R.. Trusted Kernel-Based Coalition Formation[C]. AAMAS'05, Utrecht, Netherlands, July 25-29, 2005.
- [9] Blaze M., Feigenbaum J., Lacy J.. Decentralized trust management[C]. In: Dale, J., Dinolt, G., eds. Proceedings of the 17th Symposium on Security and Privacy. Oakland, CA: IEEE Computer Society Press, 1996. 164~173.
- [10] Brandt F., Sandholm T., Shoham Y., Spiteful Bidding in Sealed-Bid Auctions[C]. IJCAI'07, India, 2007, p. 1207-1214.
- [11] Candale T., Sen S.. Multi-Dimensional Bid Improvement Algorithm for Simultaneous Auctions[C], IJCAI'07, India, 2007, p. 1215-1220.
- [12] Chevaleyre Y., Reaching Envy-free States in Distributed Negotiation Settings[C], IJCAI'07, India, 2007, p. 1239-1244.
- [13] Coehoorn R. M., Jennings N. R.. Learning an opponent's preferences to make effective multi-issue negotiation tradeoffs [C], Proc. 6th Int Conf. on E-Commerce, Delft, The Netherlands, 59-68. 2004.
- [14] Conitzer V., Sandholm T.. Complexity of constructing solutions in the core based on synergies among coalitions[J], Artificial Intelligence, 2006, 170: 607~619.
- [15] Dang V. D., Jennings N. R.. Generating coalition structures with finite bound from the optimal guarantees[C], in: Proceedings of the Third International Joint Conference on Autonomous Agents and MultiAgent Systems (AAMAS'04), 2004, pp. 564~571.
- [16] Dang V. D., Dash R. K., Rogers A. Jennings N. R.. Overlapping Coalition Formation for Efficient Data Fusion in Multi-Sensor Networks[C], (AAAI'06), Boston, MA, July 2006.
- [17] Dang V. D., Jennings N. R.. Coalition Structure Generation in Task-Based Settings[C]. ECAI06.
- [18] Etzioni O.. Moving up the information food chain: deploying softbots on the World Wide Web[C]. In proceedings of the 13th National Conference on Artificial Intelligence (AAAI'96), Portlan, OR, 1996, pp. 4-8.
- [19] Fatima S. S., Wooldridge M., Jennings N. R.. Optimal agenda for Multi-issue Negotiation[C], AAMAS'03, Melbourne, Australia , 2003, p. 129-136.
- [20] Fatima S. S., Wooldridge M., Jennings N. R.. An agenda-based framework for multi-issue negotiation[J]. Artificial Intelligence, 2004, 152(1): 1-45.
- [21] Fatima S. S., Wooldridge M., Jennings N. R.. Multi-Issue Negotiation with Deadlines[J], Journal of Artificial Intelligence Research (JAIR) , 2006, 27: 381-417.
- [22] Faratin P., Sierra C., Jennings N. R.. Using similarity criteria to make trade-offs in automated negotiations[J], Artificial Intelligence, 2002, 142 (2): 205-237.

- [23] Gao J., Zhang W.. Multi-Agent Negotiation Optimization Based on Accelerating Chaos Search Method[C], ICCSE'2006, 2006.8.
- [24] Gao J., Zhang W., Holon Based Self-Organization Evolution in MAS[C], DCABES'07, pp.768-771.
- [25] Gao J., Zhang W.. Agent-Based Multi-Dimensional Trust Model in Grid Environments[C]. ICCSE'07, pp.691-695.
- [26] Gatti N., Giunta F. D., Marino S.. Alternating-offers bargaining with one-sided uncertain deadlines: an efficient algorithm [J]. *Artificial Intelligence*, 2008,172:1119–1157.
- [27] Gerding H., Rogers A., Dash R. K., Jennings N. R.. Sellers Competing for Buyers in Online Markets: Reserve Prices, Shill Bids, and Auction Fees[C], IJCAI'07, India, 2007, p. 1287-1293.
- [28] Griffiths N., Luck M.. Coalition Formation through Motivation and Trust[C], AAMAS'03, July 14-18, 2003, Melbourne, Australia.
- [29] He L., Huang H., Zhang W., Zhao K.. ALRS: agent based literature recommendation system[C]. Accepted by 2009 international workshop on intelligent systems and applications (ISA'09), Wuhan, China, May 2009.
- [30] Holland A., Sullivan B. O., Truthful Risk-Managed Combinatorial Auctions[C], IJCAI'07, India, 2007, p. 1315-1320.
- [31] Huynh T. D., Jennings N. R., Shadbolt N. R.. Developing an Integrated Trust and Reputation Model for Open Multi-Agent Systems[C], AAMAS'04. New York, USA. July 19-23, 2004.
- [32] Huynh T. D., Jennings N. R., Shadbolt N. R.. An integrated trust and reputation model for open multi-agent systems[J]. *Journal of autonomous Agent multi-agent system*, 13, pp. 119–154, 2006.
- [33] Jennings N. R.. on agent-based software engineering[J]. *Artificial Intelligence*, 2000, 117(2):277-296.
- [34] Jennings N. R., Faratin P., Lomuscio A. R., et al., Automated negotiation: prospects, methods and challenges[J]. *International Journal of Group Decision and Negotiation*, 2001, 10(2): 199-215.
- [35] Jøsang A.. A model for trust in security systems[C]. In: *Proceedings of the 2nd Nordic Workshop on Secure Computer Systems*. 1997.
- [36] Jøsang A., Ismail R.. The beta reputation system[C]. In *Proceedings of the 15th Bled Conference on Electronic Commerce*, Bled, Slovenia, June 2002.
- [37] Jøsang A., Ismail R., Boyd C.. A Survey of Trust and Reputation Systems for Online Service Provision [J]. *Decision Support Systems*, 2007,43(2): 618-644.
- [38] Karunatillake N. C., Jennings N. R., Rahwan I., et al., Argument-based negotiation within a social context[C], *Proceedings of 2nd International Workshop on Argumentation in Multi-Agent Systems*, Utrecht, Netherlands, 74-88. 2005
- [39] Kraus S. et al., Resolving crises through automated bilateral negotiations[J], *Artificial Intelligence*, 2008, 172(1): 1-18.
- [40] Liao C.. Belief, information acquisition, and trust in multi-agent systems—A modal logic formulation[J], *Artificial Intelligence* 149, pp. 31–60, 2003.
- [41] Lin R., Kraus S., Wilkenfeld J., Barry J.. Negotiating with Bounded Rational Agents in Environments with Incomplete Information Using an Automated Agent[J], *Artificial Intelligence*, 2008, 172(6-7): 823-851.
- [42] Luo X., Jennings N. R., Shadbolt N., etc, A fuzzy constraint based model for bilateral multi-issue negotiations in semi-competitive environments[J]. *Artificial Intelligence*, 2003,148 (1-2): 53-102.
- [43] Luo X., Jennings N. R., Shadbolt N.. Acquiring user tradeoffs and preferences for negotiating agents: a default then adjust method, *Journal of Human Computer Studies*[J], 2006, 64: 304-321.
- [44] Manisterski E., Sarne D., Kraus S.. Enhancing MAS Cooperative Search Through Coalition Partitioning[C], IJCAI'07, India, 2007, p. 1415-1421.
- [45] Mui L.. Computational model for trust and reputation: Agent, evolutionary games, social networks, [D]. Massachusetts Institute of Technology, MA, USA, 2003.
- [46] Narayanan V., Jennings N. R.. Learning to Negotiate Optimally in Non-Stationary Environments[C]. CIA'06.
- [47] Neuman J. V., Morgenstern O.. *Theory of Games and Economics Behaviour*[M]. Princeton University Press, Princeton, NJ, 1944.

- [48] Pratt J.W.. Risk aversion in the small and in the large [J]. *Econometrica*, 1964, 32:122-136.
- [49] Rahwan T., Ramchurn S. D., Dang V. D., Jennings N. R.. Near-optimal anytime coalition structure generation[C]. In Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI'07) in Hyderabad, India. 2007, pp. 2365-2371.
- [50] Rahwan T., Ramchurn S. D., Giovannucci A., Dang V. D., Jennings N. R.. Anytime optimal coalition structure generation[C]. In Proceedings of the 22nd conference on artificial intelligence (AAAI'07) in Vancouver, Canada. 2007, pp. 1184-1190.
- [51] Rahwan T., Jennings N. R.. An algorithm for distributing coalitional value calculations among cooperating agents[J], *Artificial Intelligence*, 2007, 171: 535–567.
- [52] Rahwan T., Jennings N. R.. An improved dynamic programming algorithm for coalition structure generation[C]. In Proceedings of the 7th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'08), Portugal, 2008: 1417–1420,
- [53] Ramchurn S. D., Jennings N. R., C. Sierra, Persuasive negotiation for autonomous agents: A rhetorical approach[C], IJCAI'03, Acapulco, Mexico, 9-17. 2003.
- [54] Ramchurn S. D., Jennings N. R., Sierra C., Godo L., A Computational Trust Model for Multi-Agent Interactions based on Confidence and Reputation[C], AAMAS'04., New York, New York, USA, July, 2004.
- [55] Ramchurn S. D., Sierra C., Godo L., Jennings N. R., Negotiating using rewards[J], *Artificial Intelligence*, 2007, 171: 805–837.
- [56] Rao AS, Georgeff MP. Modeling rational agents within a BDI architecture[C]. In: Allen J, Fikes R, Sandewall E, eds. *Principles of Knowledge Representation and Reasoning: Proc. of the 2nd Int'l Conf. (KR-91)*. San Mateo: Morgan Kaufmann Publishers, 1991, pp. 473–484.
- [57] Rao AS, Georgeff MP. The semantics of intention maintenance for rational agents[C]. In: Mellish SC, ed. *Proc. of the 14th Int'l Joint Conf. on Artificial Intelligence*. San Mateo: Morgan Kaufmann Publishers, 1995, pp. 704–710.
- [58] Sabater, J., Sierra, C. Regret: A reputation model for gregarious societies[C]. In Proceedings of the 4th Workshop on Deception Fraud and Trust in Agent Societies, 2001, pp. 61–70.
- [59] Saha S., Sen S., An Efficient Protocol for Negotiation over Multiple Indivisible Resources[C], IJCAI'07, India, 2007, pp. 1494-1499.
- [60] Sandholm, T. and Lesser, V. Coalitions among Computationally Bounded Agent[J]. *Artificial Intelligence*, 94(1), 1997 Special issue on Economic Principles of MultiAgent Systems, pp. 99-137.
- [61] Sandholm, T., Larson, K., anderson, M., Shehory, O., and Tohme, F., coalition Structure Generation with Worst Case Guarantees[J], *Artificial Intelligence*, 1999, 111(1-2): 209-238.
- [62] Sandholm T., Algorithm for optimal winner determination in combinatorial auctions[J]. *Artificial Intelligence*, 2002, 135: 1-54.
- [63] Sandholm T., Suri S., BOB: Improved winner determination in combinatorial auctions and generalizations[J], *Artificial Intelligence*, 2003, 145: 33–58.
- [64] Schwartz D. G.. Agent-oriented epistemic reasoning: Subjective conditions of knowledge and belief[J]. *Artificial Intelligence*, 2003, 148: 177–195.
- [65] Sen S., Dutta P.. Searching for optimal coalition structures[C]. In Proceedings of the Fourth International Conference on MultiAgent Systems, 2000, pp. 286–292.
- [66] Sen S.. Believing others: Pros and cons[J], *Artificial Intelligence*, 2002, 142: 179-203.
- [67] Shehory O., Kraus S., Task allocation via coalition formation among autonomous Agents [C], in: Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI'95), 1995, pp. 655–661.
- [68] Shehory O., Kraus S.. Methods for task allocation via Agent coalition formation [J], *Artificial Intelligence*, 1998, 101(1–2): 165–200.
- [69] Teacy W. T. L., Patel J., Jennings N. R., Luck M.. Coping with Inaccurate Reputation Sources: Experimental Analysis of a Probabilistic Trust Model[C], AAMAS'05, July 25-29, 2005, Utrecht, Netherlands.

- [70] Teacy W. T. L., Patel J., Jennings N. R., Luck M.. TRAVOS: Trust and reputation in the context of inaccurate information sources[J], Journal of autonomous Agent multi-agent system, 2006, 12: 183-198.
- [71] Tong X., Zhang W., Agent multi-issue negotiation with cases[C]. The 5th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD'08), Jinan, 2008, pp.96-100.
- [72] Tong X., Huang H., Zhang W.. Agent long-term coalition credit[J]. Expert systems with applications, 2009, 36(5): 9457-9465.
- [73] Tong X., Huang H., Zhang W.. Group trust and group reputation[C]. ICNC'09, 2009.
- [74] Turing A. M. Computing machinery and intelligence[J]. In Computer and thought (ed. E. A. Feigenbaum). McGraw-Hill, 1963.
- [75] Vassileva J., Breban S., Horsch M. , Agent Reasoning Mechanism for Long-Term Coalitions Based on Decision Making and Trust[J], Computational Intelligence, 2002, 18(4): 583-595.
- [76] Wooldridge M., Jennings N.R. Intelligent agents: theory and practice[J]. The Knowledge Engineering Review, 1995, 10(2): 115-152.
- [77] Wooldridge M. An introduction to multi-agent systems[M]. 2002, John Wiley & Sons, Inc.
- [78] Wooldridge M., Dunne P. E., On the computational complexity of qualitative coalitional games[J], Artificial Intelligence, 2004,158: 27-73.
- [79] Wooldridge M., Dunne P. E.. On the Computational Complexity of Coalitional Resource Games[J]. Artificial Intelligence, 2006, 170(10): 835-871.
- [80] Wu X., Fan B., Zhang W., Study on Performance Improving for Resource Management of Computational Grid Based upon AT[C], Watam Press, 2006,5.
- [81] Yokoo M., Sakurai Y., Matsubara S., Robust combinatorial auction protocol against False-name Bids[C], In Proceedings of the National Conference on Artificial Intelligence. 2000, 110-115.
- [82] Yokoo M., Sakurai Y., Matsubara S., Bundle design in robust combinatorial auction protocol against False-name Bids[C], In Proceeding of the International Joint Conference on Artificial Intelligence, 2001: 1095~1101.
- [83] Yokoo M., Sakurai Y., Matsubara S., The effect of false-name bids in combinatorial auctions: new fraud in internet auctions[J], Games and Economic Behavior. 2004,46(1): 174-188.
- [84] Yu B., Singh P. Detecting deception in reputation management[C]. In Proceedings of the 2nd International Joint Conference on Autonomous Agents and Multi-Agent Systems, Melbourne, Australia, July 2003. ACM Press, pp.73-80.
- [85] Zhang D. M., Wong W. Y.. A Web-Based Negotiation Agent Using CBR[C]. PRICAI Workshops, Melbourne, Australia, 2000: 183-198.
- [86] Zhu F., Guan J., et al. An Automatic Negotiation Method Based on CBR and Agent Reasoning[C], Proceedings of CIT'05, Shanghai, China, 2005: 1019-1025.
- [87] 常志明, 毛新军, 齐治昌. 基于Agent的网构软件构件模型及其实现[J]. 软件学报, 2008,19 (05): 1113-1124.
- [88] 常志明, 毛新军, 王戟, 齐治昌. 多Agent系统中软构件的动态绑定机制及其操作语义[J]. 计算机研究与发展, 2007,44 (5): 806-814.
- [89] 董明楷, 张海俊, 史忠植. 基于动态描述逻辑的主体模型[J]. 计算机研究与发展, 2004, 41(5): 780-786
- [90] 窦文,王怀民,贾焰,邹鹏.构造基于推荐的 Peer-to-Peer 环境下的 Trust 模型[J]. 软件学报, 2004,15(4): 571-583.
- [91] 高坚, 张伟. 多Agent系统中双边多指标自动协商的ACEA算法[J]. 计算机研究与发展, 2006, 43(6): 1104-1108.
- [92] 郭庆, 陈纯. 基于整合效用的多议题协商优化[J]. 软件学报, 2004,15(5): 706-711.
- [93] 贺利坚, 张伟. 基于约束图分片求解 DCOP 的 Agent 组织结构[J]. 计算机研究与发展, 2007,44(3).
- [94] 贺利坚, 黄厚宽, 张伟. 多 Agent 系统中信任和信誉系统研究综述[J]. 计算机研究与发展, 2008,45(7).
- [95] 胡宁, 邹鹏, 朱培栋. 基于信誉机制的域间路由安全协同管理方法[J]. 软件学报. doi: 10.3724/SP.J.1001.2009.03479, <http://www.jos.org.cn/1000-9825/3479.htm>
- [96] 胡山立, 石纯一. Agent的意图模型[J]. 软件学报, 2000,11(7): 965~970.

- [97] 胡山立, 石纯一. Agent-BDI 逻辑[J]. 软件学报, 2000,11(10): 1353~1360.
- [98] 胡山立, 石纯一. 一种任一时间联盟结构生成算法[J]. 软件学报, 2001,12 (05): 729-734.
- [99] 胡山立, 石纯一. 给定限界要求的联盟结构生成[J]. 计算机学报, 2001,24 (11): 1185-1190.
- [100] 胡山立, 石纯一. Agent 逻辑和真假子集语义[J]. 软件学报, 2002,17(3):2112-2115.
- [101] 胡山立, 石纯一. Agent 意图的双子集语义改进模型[J]. 软件学报, 2006,17(3): 396-402.
- [102] 蒋建国, 夏娜, 齐美彬, 木春梅. 一种基于蚁群算法的多任务联盟串行生成算法[J]. 电子学报, 2005,(12): 2178-2182.
- [103] 金萍, 石纯一. 一种边际效用递减组合拍卖的胜者决定算法[J]. 计算机研究与发展, 2006,43(07): 1142-1148.
- [104] 金萍, 石纯一. 一种递增叫价的多属性拍卖方法[J]. 计算机研究与发展, 2006,43(7): 1135-1141.
- [105] 金萍, 石纯一. 一种暗标叫价的多属性拍卖方法[J]. 计算机学报, 2006,29 (1): 145-152.
- [106] 金萍, 石纯一. 多活性级递增叫价组合拍卖方法[J]. 清华大学学报, 2006,46(4).
- [107] 黎建兴, 毛新军, 束尧. 软件 Agent 的一种面向对象设计模型[J]. 软件学报, 2007,18(3): 582-591.
- [108] 李景涛, 荆一楠, 肖晓春, 王雪平, 张根度. 基于相似度加权推荐的 P2P 环境下的信任模型[J]. 软件学报, 2007,18(1): 157-167.
- [109] 李毅, 石纯一. 基于 BDI 的对手 Agent 模型[J]. 软件学报, 2002,13(4): 643-648.
- [110] 李毅, 石纯一. 对群体 Agent 的意图跟踪[J]. 软件学报, 2002,13(7): 1298-1302.
- [111] 刘惊雷, 童向荣, 张伟. 一种快速构建最优联盟结构的方法[J]. 计算机工程与应用, 2006, 2(4): 35-44.
- [112] 刘惊雷, 张伟, 范宝德, 郑小鹏. 角色分配格中的特异元[J]. 南京大学学报, 2008, 44(2).
- [113] 陆萍萍. 多 Agent 系统中信任管理研究[D], 硕士论文, 扬州大学, 2007 年.
- [114] 罗杰文, 史忠植, 王茂光等. 基于动态描述逻辑的多主体协作模型[J]. 计算机研究与发展, 2006,43(8): 1317-1322.
- [115] 马光伟, 徐晋晖, 石纯一. 社会 Agent 的 BDO 模型[J]. 计算机学报, 2001,24(5): 521~528.
- [116] 彭冬生, 林闯, 刘卫东. 一种直接评价节点诚信度的分布式信任机制[J]. 软件学报, 2008,19(4): 946-955.
- [117] 石纯一, 张伟 著. 基于 Agent 的计算[D]. 中国计算机学会学术著作丛书, 清华大学出版社, 2007 年 4 月.
- [118] 苏射雄, 胡山立, 林超峰, 郑盛福. 基于局部最优的联盟结构生成算法[J]. 计算机研究与发展, 2007,44(2): 277-281.
- [119] 唐文, 胡建斌, 陈钟. 基于模糊逻辑的主观信任管理模型研究[J]. 计算机研究与发展, 2005, 2(10): 1654-1659.
- [120] 童向荣, 张伟. 一种入侵检测系统的分布式多 Agent 通信机制[J]. 计算机工程与应用, 2005,12.
- [121] 童向荣, 张伟. 动态联盟收益值的再励学习[J]. 计算机工程与应用, 2006,6: 85-87.
- [122] 童向荣, 张伟. 基于模糊盟友关系的多主体系统长期联盟[J]. 计算机研究与发展, 2006, 43(8): 1445-1449.
- [123] 童向荣, 张伟. 基于信任和声誉的 Agent 组织信誉模型[J]. 计算机科学与探索, 2007,1(3):325-330.
- [124] 童向荣, 黄厚宽, 张伟. Agent 协商研究进展[J]. 计算机工程与应用, 2007, 32.
- [125] 童向荣, 黄厚宽, 张伟. Agent 动态交互信任预测与行为异常检测模型[J], 计算机研究与发展[J], 2009, 46(8): 1364-1370.
- [126] 童向荣, 黄厚宽, 张伟. 一种基于案例的 Agent 多议题协商模型[J]. 计算机研究与发展, 2009, 46(9).
- [127] 王立春, 陈世福. 多 Agent 多问题协商模型[J]. 软件学报, 2002, 13(8):1637~1643.
- [128] 王黎明, 黄厚宽. 一个基于多阶段的多 Agent 多问题协商框架[J]. 计算机研究与发展, 2005,42(11): 1849-1855.
- [129] 王黎明, 黄厚宽, 柴玉梅. 基于信任和 K 臂赌博机问题选择多问题协商对象[J]. 软件学报, 2006, 17(12): 2537-2546.
- [130] 王平. 多 Agent 系统中的信任模型研究[D], 硕士论文, 西南师范大学, 2005 年.

参考文献

- [131] 杨佩, 高阳, 陈兆乾. 一种劝说式多 Agent 多议题协商方法[J]. 计算机研究与发展, 2006,43(7): 1149-1154.
- [132] 张洪, 段海新, 刘武. RRM: 一种具有激励机制的信誉模型[J]. 中国科学 E 辑:信息科学, 2008,38(10): 1747-1759.
- [133] 张双民, 石纯一. 一种基于特征向量提取的 FMDP 模型求解方法[J]. 软件学报, 2005, 16(5).
- [134] 张双民, 石纯一. 基于群体 Agent 合作求解的测试床——MAS-Soccer[J]. 清华大学学报, 2005, 45(4).
- [135] 张伟, 石纯一. Agent 组织结构设计中的一种形式语义[J]. 软件学报, 2002,13 (03): 447-452.
- [136] 张伟, 石纯一. Agent 组织的一种递归模型[J]. 软件学报, 2002,13 (11): 2149-2154.
- [137] 张伟, 王一川, 石纯一. 一种基于资源约束的 Agent 组织规则生成机制[J]. 计算机研究与发展, 2002,39 (12): 1592-1597.
- [138] 张伟, 石纯一. Agent 的组织承诺和小组承诺[J]. 软件学报, 2003,14 (03): 473-478.
- [139] 张伟, 王一川, 石纯一. Agent 组织规则的再励学习[J]. 计算机研究与发展, 2003,40(3): 430-434.
- [140] 张伟. 面向 Agent 的软件工程研究进展[J]. 知识科学中的基本问题研究(刘大有主编), 中国计算机学会学术著作丛书, 清华大学出版社, 2006 年 10 月.
- [141] 张伟, 高坚, 贺利坚, 童向荣, 石纯一. Agent 组织研究进展[J], 计算机研究与发展, 2006 年 8 期(增刊): 6-11.
- [142] 张新良, 石纯一. M-POMOP 模型及其划分求解算法[J]. 清华大学学报: 自然科学版, 2005, 45(10).
- [143] 张新良, 石纯一. 基于描述逻辑的 Agent 组织[J]. 计算机研究与发展, 2005,42 (11): 1843-1848.
- [144] 张新良, 石纯一. 对称和非对称的启发式多 Agent 再励学习方法[J]. 清华大学学报, 2006, 46(4).
- [145] 张新良, 石纯一. 多 Agent 联盟结构动态生成算法[J]. 软件学报, 2007,18(3): 574-581.

攻读博士期间发表和已录用的学术论文

- [1] Tong Xiangrong, Huang Houkuan, Zhang Wei. Agent long-term coalition credit. *Expert Systems With Applications*, 2009, 36(5): 9457-9465.
- [2] 童向荣, 黄厚宽, 张伟. Agent 动态交互信任预测与行为异常检测模型. *计算机研究与发展*. 2009,46(8): 1364-1370.
- [3] 童向荣, 黄厚宽, 张伟. 一种基于案例的 Agent 多议题协商模型. *计算机研究与发展*, 2009,46(9): 1508-1514.
- [4] 童向荣, 黄厚宽, 张伟. 一种 Agent 组织信任计算模型. *北京交通大学学报*, 2009,33(5).
- [5] 童向荣, 黄厚宽, 张伟. Agent 协商研究进展. *计算机工程与应用*, 2007, 43(32): 23-28.
- [6] Tong Xiangrong, Huang Houkuan, Zhang Wei. Intangible payoff and credit of agent long-term coalition. *计算机科学(论文集)*, 2007.8.
- [7] 童向荣, 黄厚宽, 张伟. 一种非线性效用函数的 Agent 多资源配置协商模型, 已投软件学报.

独创性声明

本人声明所呈交的学位论文是本人在导师指导下进行的研究工作和取得的研究成果，除了文中特别加以标注和致谢之处外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得北京交通大学或其他教育机构的学位或证书而使用过的材料。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。

学位论文作者签名：章向东 签字日期：2009 年 12 月 2 日

学位论文数据集

关键词*	密级*	中图分类号*	UDC	论文资助
多 Agent 系统; 信任; 协商	公开	TP18		
学位授予单位名称*		学位授予单位代 码*	学位类别*	学位级别*
北京交通大学		10004	工学	博士
论文题名*		并列题名		论文语种*
多 Agent 合作求解中的信任与协商 研究				中文
作者姓名*	童向荣		学号*	06112066
培养单位名称*		培养单位代码*	培养单位地址	邮编
北京交通大学		10004	北京市海淀区西 直门外上园村 3 号	100044
学科专业*		研究方向*	学制*	学位授予年*
计算机应用技术		多 Agent 系统	3.5 年	2010 年
论文提交日期*	2009.9			
导师姓名*	黄厚宽		职称*	教授
评阅人	答辩委员会主席*		答辩委员会成员	
石纯一、史忠植、 田盛丰、于剑	石纯一		石纯一、史忠植、田盛丰、于剑、 王志海	
电子版论文提交格式 文本(√) 图像() 视频() 音频() 多媒体() 其他() 推荐格式: application/msword; application/pdf				
电子版论文出版(发布)者		电子版论文出版(发布)地		权限声明
论文总页数*	118			
共 33 项, 其中带*为必填数据, 为 22 项。				

