文章编号:1001 - 9081(2006)03 - 0663 - 03

一种具有自主学习能力的并发协商模型

张 谦,邱玉辉
(西南大学 计算机与信息科学学院,重庆 400715)
(qianzh@swnu.edu.cn)

摘 要:提出一种具有自主学习能力的并发协商模型,通过使用增强学习方法的 Q 学习算法生成协商提议,使用相似度方法评价提议,使得 Agent 能够在半竞争、信息不完全和不确定以及存在最大协商时间的情况下,更为有效地完成多议题多 Agent 并发协商。

关键词:并发协商;自动协商;增强学习;Q 学习;相似度方法
中图分类号: TP181 **文献标识码**:A

A concurrent negotiation model with automated learning capability

ZHANG Qian, QIU Yu-hui
(*College of Computer and Information Science, Southwest University, Chongqing* 400715, *China*)

Abstract: A concurrent negotiation model with automated learning capability was developed in trading environments. By using Q-learning algorithm to propose own proposal and similarity criteria to evaluate the opposing party's proposal, agents can participate in concurrent multi-issue negotiation in semi-competitive situations in which there exists information uncertainty and deadlines. This model enables the negotiation more effective.

Key words: concurrent negotiation; automated negotiation; reinforcement learning; Q-learning; similarity criteria

0 引言

在电子商务活动中为了达到更好的效果,获得更多的利益,买方希望与多个卖方或者卖方希望与多个买方进行并发协商(即同时进行协商),从中选择最佳的交易方案[1,2]。而理论和实验表明,在 Agent 决策过程中,如果能够考虑对手思考些什么并且在交互过程中学习对手的信念和偏好可以有效增加自我效用,如果所有的 Agent 都进行学习,系统的联合效用将接近最优[3~6]。因此本文提出了一个具有学习能力的并发协商模型,本模型使用增强学习方法的 Q 学习算法来生成提议,使用相似度方法来评价提议,使得参与协商的 Agent 具有自主学习能力,能够更为有效地完成协商。

自动协商是一组自治 Agent 为了某些与利益相关的议题相互进行通信已求达到一致的过程。它是 Multi-Agent 系统中一种重要的交互形式。协商的议题可以是价格、数量和质量等等。一个提议是不同议题值的组合,协商双方通过轮流提议,即修改提议中不满意的议题值从而达成一致,形成一个双方都认可的合同。并发协商是指一个 Agent 同时与多个对手进行协商,增加达成一致的机率,以求获得最多的效用,它属于一对多协商形式,也可以扩展为多对多协商形式。

1 并发协商模型

定义 1 将本并发协商模型定义为八元组 $\{A, I, X, T, Q, SIM, AT, ST\}$,其中,$A = \{c, b_1, b_2, \cdots, b_j, \cdots, b_n, s_1, s_2, \cdots, s_j, \cdots, s_n, 0 < j \leqslant n, n > 0\}$ 表示参与协商的 Agent 集合,c 表示协调 Agent,$b_1, b_2, \cdots, b_n$ 表示 n 个相对独立的子买方 Agent,$s_1, s_2, \cdots, s_n$ 表示 n 个相互独立的卖方 Agent;$I = \{i_1, i_2, \cdots, i_n, n > 0\}$ 表示协商议题集合;$X = \{x^i \mid x^i \in D, i \in I\}$ 表示协商议题集合 I 的所有可能取值集合,$D = [\min\{\min(x^i_{b_j}), \min(x^i_{s_j})\}, \max\{\max(x^i_{b_j}), \max(x^i_{s_j})\}]$,$[\min(x^i_{b_j}), \max(x^i_{b_j})]$ 和 $[\max(x^i_{s_j}), \max(x^i_{s_j})\}$ 分别表示子买方 b_j 和卖方 s_j 关于议题 i 的提议区间,$x_{b_j}(t) = \sum_{i \in I} w^i_b \cdot x^i_{b_j}(t)$ 和 $x_{s_j}(t) = \sum_{i \in I} w^i_{s_j} \cdot x^i_{s_j}(t)$ 分别表示 b_j 和 s_j 所做的第 t 轮提议,w^i_b 和 $w^i_{s_j}$ 分别是买方 b 和卖方 s_j 对于议题 i 所赋的权值,并且 $\sum_{i \in I} w^i_b = 1$, $\sum_{i \in I} w^i_{s_j} = 1$;$T = \{1, 2, \cdots, t, \cdots, \min(T_b, Ts_i)\}$ 表示协商轮数集合;Q 表示 Q 函数;SIM 表示相似度函数;$AT = \{a_t, a_h, a_c\}$ 表示协商类型集合,其中 a_t 表示非亲我类型,a_h 表示过渡类型,a_c 表示亲我类型;$ST = \{s_t, s_l, s_c\}$ 表示 Agent 采用的协商策略集合,其中 s_t 表示强硬策略,s_l 表示线性策略,s_c 表示让步策略,这些策略在本模型中对应选择相应的时间信念函数。

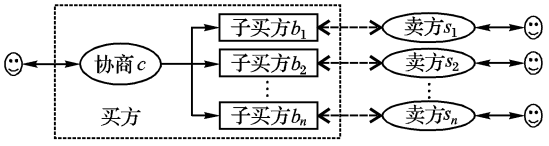

图 1 系统结构

本模型是在半竞争、信息不完全和不确定以及存在最大协商时间的动态协商环境中完成多议题多 Agent 并发协商。其中的 Agent 都是自利和理性的,每个 Agent 都有自己的信念和偏好,这些信息是私有的,对于其他 Agent 是保密的。Agent 的初始提议信息是从市场和历史经验中获得的。

如图 1 所示,本模型中需要购买服务的一方被称为买方,而出售服务的一方被称为卖方。本模型实现的是一对多形式的并发协商,即一个买方和多个卖方同时进行协商。买方由

收稿日期:2005 - 09 - 26 **修订日期**:2005 - 11 - 21

作者简介:张谦(1976 -),女,江苏靖江人,硕士研究生,主要研究方向:电子商务、机器学习; 邱玉辉(1938 -),男,重庆江津人,教授,博士生导师,主要研究方向:人工智能、机器学习.

一个协调 Agent 和多个子买方 Agent 组成。子买方 Agent 由协调 Agent 创建,负责与其中的某个卖方进行协商,将每一轮的协商结果传送给协调 Agent,再由协调 Agent 确定最终与哪个卖方进行交易。一个子买方与一个卖方进行一次协商的完整过程称为一个子协商。为便于讨论,将协商时间离散化,最大协商时间可以看作是允许协商的最大轮数。买方的最大协商轮数设为 T_b,子买方 Agent 和卖方进行协商以及协调 Agent 确定交易对象都必须在 T_b 内完成。每个卖方也有各自不同的最大协商轮数设为 T_{si}。买卖双方都有各自的协商策略,特别地,买方 Agent 在协商过程中可以对不同的卖方 Agent 采用不同的协商策略。本文只针对买方 Agent 进行建模,所建模型也容易扩展为针对卖方 Agent 的模型以及多对多的并发协商模型。

2 增强学习与生成提议

增强学习(reinforcement learning) 又称为强化学习,是一种无导师在线学习技术。增强学习强调在与环境的交互中学习,通过感知环境状态和从环境中获得不确定回报来学习动态系统的最优行为策略,它以获得极大化期望回报为学习目标,其中期望回报包括立即回报和延迟回报。对于现实世界中的某些优化决策和控制问题,增强学习的回报函数较易设计,因此成为求解复杂决策问题的一种可行手段和重要技术。增强学习在理论和算法方面已经取得大量研究成果,同时在机器人控制、优化调度等领域中也获得了若干成功的应用[3,5,7]。随着增强学习方法在 Multi-agent 系统中的应用,增强学习也被应用到 Agent 自动协商中。Q 学习算法是增强学习方法的一个重要算法。当存在延迟回报,并且 Agent 不具备关于立即回报和环境状态转移的完美知识时,Q 学习算法可以通过学习 Q 函数来很好地估计每个状态的延迟回报。本文正是利用 Q 学习算法来计算提议行为对应的 Q 值从而生成提议。

定义 2 时间信念是指 Agent 对协商对手接受其提议概率的认识。用 $p_{tbj\to sj}(t)$ 表示子买方 b_j 对卖方 s_j 接受其第 t 轮提议的概率。

定义 3 提议信念是指 Agent 对达成一致的提议在其议题值范围内的概率分布的认识。用 D_{tbj} 表示子买方 b_j 对达成一致的提议在$[\min(x^i_{b_j}), \max(x^i_{b_j})]$ 内的概率分布的认识。

本模型将协商双方交互提议的过程看作 Agent 在某一状态下选择某一提议行为来实现状态转移的过程。如果 Agent 在第 t 轮提议时的状态为 $s(t)$,提议为 $x(t)$,那么提议后,Agent 就进入后继状态 $s(t+1)$,如此进行下去直到协商结束。每个参与协商的 Agent 都有自己关于各个议题的提议区间,对手提议中的各个议题值必须在该区间内才是可能被接受的提议。用 $X_j(T)$ 表示子买方 b_j 与卖方 s_j 最终达成一致的提议,那么子买方 b_j 获得的回报为:

$$r_{bj} = \max(x_{b_j}) - X_j(T) \tag{1}$$

根据 Q 学习算法,子买方 b_j 的 Q 函数定义如下:

$$Q_{b_j}(s(t), x_{b_j}(t)) = r(s(t), x_{b_j}(t)) + \gamma \max_{x(t+1)} Q_{b_j}(\delta(s(t), x_{b_j}(t)), x_{b_j}(t+1)) \tag{2}$$

其中 γ 为时间贴现率,$\delta()$ 为状态转移函数。子买方 b_j 只有在最终协商达成一致后,才能获得相应的回报,因此在协商过程中,子买方 b_j 从一个状态转移到下一状态所得到的立即回报 $r_{b_j}(s(t), x_{b_j}(t)) = 0$。如果子买方 b_j 和卖方 s_j 在第 t 轮提议时达成一致,并且协调者 c 确定它和卖方 s_j 进行交易,此时子买方 b_j 的回报为 $Q^T_{b_j}$,那么子买方 b_j 在 t 轮之前各状态的 Q 值可由(1) 式计算得到:

$$Q_{b_j}(s(1), x_{b_j}(1)) = \gamma^{t-1} Q^T_{b_j} \tag{3}$$

$$Q_{b_j}(s(2), x_{b_j}(2)) = \gamma^{t-2} Q^T_{b_j} \tag{4}$$

……

$$Q_{b_j}(s(t), x_{b_j}(t)) = Q^T_{b_j} \tag{5}$$

由提议信念定义可知 $X_j(T)$ 按 D_{b_j} 分布,因此子买方 b_j 的回报 $r_{b_j} = \max(x_{b_j}) - X_j(T)$ 也按 D_{b_j} 分布,所以当买卖双方达成一致时,子买方 b_j 的回报应为期望值:

$$Q^T_{b_j} = \int_{\min(x_{b_j})}^{\max(x_{b_j})} (\max(x_{b_j} = X_j(T)) D_{b_j} \mathrm{d}X_j(T) \tag{6}$$

将(6)式代入(3) ~ (5)式可以计算出子买方 b_j 在各轮提议的 Q 值。

由时间信念定义可知,子买方 b_j 认为对于其第 1 轮至第 t 轮提议 $x_{b_j}(1), x_{b_j}(2), \cdots, x_{b_j}(t)$,卖方 s_j 都分别以概率 $p_{b_j\to s_j}(1), p_{b_j\to s_j}(2), \cdots, p_{b_j\to s_j}(t)$ 接受,于是子买方 b_j 从第 1 轮至第 t 轮提议的 Q 值序列也按时间概率分布。

子买方 b_j 在第 1 轮的 Q 值分别为:在第 1 轮达成一致时,$Q_{b_j}(s(1), x_{b_j}(1)) = Q^T_{b_j}$;在第 2 轮达成一致时,$Q_{b_j}(s(1), x_{b_j}(1)) = \gamma Q^T_{b_j}$;…;在第 t 轮达成一致时,$Q_{b_j}(s(1), x_{b_j}(1)) = \gamma^{t-1} Q^T_{b_j}$。

于是子买方 b_j 在第 1 轮 Q 值的平均期望为:

$$\overline{Q}_{b_j}(s(1), x_{b_j}(1)) = \left(\sum_{k=1}^{T_b} p_{b_j\to s_j}(k) \cdot \gamma^{k-1} \cdot Q^T_{b_j} \right) / T_b \tag{7}$$

以此类推,子买方 b_j 在第 t 轮 Q 值的平均期望为:

$$\overline{Q}_{b_j}(s(t), s_{b_j}(t)) = \left(\sum_{k=t}^{T_b} p_{b_j\to s_j}(k) \cdot \gamma^{k-1} \cdot Q^T_{b_j} \right) / (T_b - t + 1) \tag{8}$$

由此得到子买方 b_j 生成提议的方法:

$$x_{b_j}(t) = \max\left(x_{b_j} - \overline{Q}_{b_j}(s(t), x_{b_j}(t)) \right) \tag{9}$$

3 相似度方法与评估提议

定义 4 x 和 y 是给定议题集合 I 上的两个提议,那么 x 和 y 的相似度定义为 $Sim(x, y) = \sum_{i\in I} h^i \cdot Sim^i(x^i, y^i)$,其中,$h^i$ 是对协商对手关于议题 i 的权值的预测,并且 $\sum_{i\in I} h^i = 1$。

在本模型中,当卖方接受买方的提议时,规定卖方的提议等于买方的提议,此时两者提议的相似度等于 1;当卖方拒绝买方的提议时,规定卖方的提议值为 null,此时两者提议的相似度等于 0。根据卖方采取的策略,将卖方分为非亲我类型 a_t、过渡类型 a_h 和亲我类型 a_c,买方 Agent 根据卖方的类型采取以下三种策略:强硬策略 s_t、线性策略 s_l 和让步策略 s_c,这三种策略在模型中分别对应选择三种时间信念函数:时间信念增函数、时间信念常数函数和时间信念减函数。当选择时间信念增函数时表示买方 Agent 认为随着时间延长,协商达成一致的概率会越来越大;而选择时间信念减函数时,协商达成一致的概率就越来越小。采用相似度方法对卖方进行分类,也就是评估卖方提议,从而给出相应行动策略,具体是买方协调者 c 给定几个阈值 θ_1, θ_2,$0 < \theta_1 < \theta_2 < 1$。当 $0 \leqslant Sim(x_{b_j}, x_{s_j}) \leqslant \theta_1$ 时,判断卖方属于非亲我类型 a_t,买方采取强硬策略 s_t,这样就有可能使对方做出让步,使自己得到更多

的机会;当 $\theta_1 < Sim(x_{b_j}, x_{s_j}) \leqslant \theta_2$ 时,判断卖方属于过渡类型 a_h,买方采取线性策略 s_l;当 $\theta_2 < Sim(x_{b_j}, x_{s_j}) \leqslant 1$ 时,判断卖方属于亲我类型 a_c,买方采取让步策略 s_c,这样使双方达到一致的可能性最大。

4　算法实现

REPEAT
　FOR $j = 1$ to n
　{ IF $t < T_{sj}$ THEN
　　send($x_{b_j}(t)$) to c;
　　send($x_{s_j}(t)$) to c;
　　send(s_j . action) to c;
　　ELSE terminate (b_j, s_j);
　　END IF; }
　//c Decision:
　IF s_j . action = "accept " THEN $x_{b_j}(t) := x_{s_j}(t)$;
　IF s_j . action = " refuse " THEN $x_{s_j}(t) :=$ null;
　//评估提议
　$Sim(x_{b_j}(t), x_{s_j}(t)) = \sum_{i \in I} h^i \cdot Sim^i(x^i_{bj}(t), x^i_{sj}(t))$;
　IF $0 \leqslant Sim(x_{b_j}(t), x_{s_j}(t)) \leqslant \theta_1$ THEN
　　s_j . AT: = a_t;　　//非亲我类型
　　b_j . ST: = s_t;　　//强硬策略
　END IF;
　IF $\theta_1 < Sim(x_{b_j}(t), x_{s_j}(t)) \leqslant \theta_2$ THEN
　　s_j . AT: = a_h;　　//过渡类型
　　b_j . ST: = s_l;　　//线性策略
　END IF;
　IF $\theta_2 < Sim(x_{b_j}(t), x_{s_j}(t)) \leqslant 1$ THEN
　　s_j . AT: = a_c;　　//亲我类型
　　b_j . ST: = s_c;　　//让步策略
　END IF;
　send(b_j . ST) to b_j;
　//选择时间信念函数
　IF b_j . ST = s_t THEN $p_{b_j \to s_j}(t) = t/T_b$;
　IF b_j . ST = s_l THEN $p_{b_j \to s_j}(t) = 1/2$;
　IF b_j . ST = s_c THEN $p_{b_j \to s_j}(t) = 1 - t/T_b$;
　//子买方 b_j 生成提议
　$x_{b_j} = \max(x_{b_j}) - \left(\sum_{k=1}^{T_b} p_{b_j \to s_j}(k) \cdot \gamma^{k-1} \cdot Q^T_{bj} \right) /$
　$(T_b - t + 1)$;
　UNTIL $t \geqslant T_b$ OR c . action = "comfirm " END REPEAT;
　IF $t \geqslant T_b$ THEN terminate (b, s);
　IF c . action = "comfirm " THEN deal;

参考文献:

[1] NGUYEN TD, JENNINGS NR. A Heuristic Model for Concurrent Bi-lateral Negotiations in Incomplete Information Settings[A]. Proceedings of 18th International Joint Conference on AI[C]. Mexico, 2003.

[2] RAHWAN I, KOWALCZYK R, PHAM HH. Intelligent Agents for Automated One-to-Many E-Commerce Negotiation[A]. Twenty-Fifth Australian Computer Science Conference (ACSC2002) [C]. Australian, 2002.197 - 204.

[3] ARAI S, SYCARA K, PAYNE T. Experience-based Reinforcement Learning to Acquire Effective Behavior in a Multi-agent Domain[A]. Proceedings of the 6th Pacific Rim International Conference on Artificial Intelligence[C], 2000.

[4] ZENG D, SYCARA K. Bayesian Learning in Negotiation [A]. Working Notes for the AAAI Symposium on Adaptation, Co-evolution and Learning in Multiagent Systems[C]. Stanford University, CA, 1996.

[5] EXCELENTE-TOLEDO CB, JENNINGS NR. Using reinforcement learning to coordinate better[J]. Computational Intelligence, 2005, 21 (3): 217 - 245.

[6] OLIVER JR. A machine-learning approach to automated negotiation and prospects for electronic commerce[J]. Journal of Management Information Systems, 1997, 13(3): 83 - 112.

[7] TAN M. Multi-Agent Reinforcement Learning: Independent vs. Cooperative Agents[A]. Proceedings of the Tenth International Conference on Machine Learning[C]. 1993. 330 - 337.

[8] MITCHELL TM. Machine Learning[M]. Beijing: China Machine Press, 2003.

[9] NAGAYUKI Y, ISHII S, DOYA K. Multi - agent reinforcement learning: An approach based on the other agent's internal model [A]. Proceedings on the Fourth International Conference on Multi-Agent Systems (ICMAS-00)[C]. Boston, MA, 2000. 215 - 221.

[10] BUI H, KIERONSKA D, VENKATESH S. Learning other agents' preferences in multiagent negotiation[A]. Proceedings of the Thirteenth National Conference on Artificial Intelligence[C]. Menlo Park, CA, AAAI Press, 1996. 114 - 119.

(上接第 629 页)

端的序列信息,通过一系列的加热和冷却循环来扩增 DNA 片断[11];然后通过相应的序列读取装置读取序列;最后,转换序列为原始数据。

参考文献:

[1] HOCH JA, LOSICK R. Panspermia, spores and the bacillus subtilis genome[J]. Nature, 1997, 390: 237 - 238.

[2] CLELLAND CT, RISCA V, BANCROFT C. Hiding messages in DNA microdots[J]. Nature, 1999, 399: 533 - 534.

[3] WONG PC, WONG K-K, FOOTE H. Organic data memory using the DNA approach[J]. Communications of the ACM, 2003, 46 (1): 95 - 98.

[4] WASIEWICZ P, MALINOWSKI A, NOWAK R, *et al.* DNA computing: implementation of data flow logical operations[J]. Future Generation Computer Systems, 2001, 17(4): 361 - 378.

[5] JONATHAN P, COX L. Long-term data storage in DNA[J]. Trends in biotechnology, 2001, 19(7): 247 - 250.

[6] 楼士林, 杨盛昌, 龙敏南. 基因工程[M]. 北京: 科学出版社, 2002.

[7] MURRAY AW, SZOSTAK JW. Construction of ratification chromosome in yeast[J]. Nature, 1983, 305: 189.

[8] BURKE DT, CARLE GF, OLSON MV. Cloning of large segments of exogenous DNA into yeast by means of artificial chromosome vectors[J]. Science, 1987, 236: 806.

[9] 许进, 董亚非, 魏小朋. 粘贴 DNA 计算机模型(Ⅰ): 理论[J]. 科学通报, 2004, (3): 205 - 212.

[10] 许进, 李三平, 董亚非, 等. 粘贴 DNA 计算机模型(Ⅱ): 应用[J]. 科学通报, 2004, (4): 299 - 307.

[11] HARWOOD AJ. Basic DNA and RNA protocols[M]. 盛小禹, 等译. 北京: 科学出版社, 2002.