CP-nets 及其表达能力研究

刘惊雷 [1]

摘 要 偏好处理是人工智能中的一个重要研究内容, 它的 4 个研究热点是偏好的表示、提取、聚合和推理. 条件偏好网 (Conditional preference networks, CP-nets) 是一种简单直观的偏好表示的图形工具, 但很少有工作研究 CP-nets 的表达能力. 本文研究 CP-nets 的表达能力, 详细研究了 CP-nets 表达偏好的完备性, 其上构造的运算复杂度以及适用的场合. 首先给出了 CP-nets 模型上的几个运算, 利用改进的 Warshall 算法求出了二值网的强占优测试在最坏情况下的复杂度为 $O(4^n)$. 其次通过构造 CP-nets 导出图及其性质的研究, 得出 CP-nets 特别适合不完全信息下的多属性定性偏好决策. 当需要处理更完全信息时, 可借助于与 Agent 的交互来完成. 虽然我们给出了 CP-nets 的强占优测试的理论解, 但其理论上可解, 实际上不可解. 为了解决强占优测试的指数级复杂度问题, 本文最后给出了一种带有软约束的满足问题 (Soft constraint satisfaction problem, SCSP) 的求解方法. 它把 CP-nets 中的定性运算转为约束半环中的定量运算, 从而将指数级的复杂度转化为多项式的复杂度, 间接提高了部分 CP-nets 的表达能力. 本文所做的工作是对 Boutilier 和 Bistarelli 工作的改进和提高.

关键词 条件偏好网, 表达能力, 强占优测试, 偏好的完备性, 改进的 Warshall 算法, 不完全信息下的多属性定性偏好决策, 带有软约束的满足问题

DOI 10.3724/SP.J.1004.2011.00290

Research on CP-nets and Its Expressive Power

LIU Jing-Lei[1]

Abstract Preference handling is one of the important researching contents in artificial intelligence, whose four studying hotspots are preference representation, preference elicitation, preference aggregation and preference inference. The conditional preference networks (CP-nets) is a simple and intuitive graphical tool for representing conditional ceteris paribus preference statements over the values of a set of variables. But there are very few works that address the problem of expressive power of CP-nets. In this paper, the expressive power of CP-nets is studied, in particular, preference completeness, some operations complexity of CP-nets and situation to which the model is applicable are discussed detailedly. Firstly, some operations of CP-nets are discussed; by using the improved Warshall algorithm, we solve the problem of worst case complexity of strong dominance testing with respect to binary-valued CP-nets, and prove its complexity is $O(4^n)$. Secondly, by constructing induced graph of CP-nets and studying its properties, we draw a conclusion that CP-nets suit multiple attributes qualitative decision making under incomplete preference information situation especially. When handling complete preference information, it can be fulfilled by interactive communication with agent. Strong dominance testing is a tractable problem in theory, but it is a intractable problem in practice. In order to solve this exponential order complexity, at last, some reduction techniques from qualitative judging to quantitative judging with soft constraint satisfaction problem (SCSP) are given. Because qualitative judging on c-simiring is of linear complexity, it increases the expressive power of some CP-nets. All these can be seen as the improvement and refinement of Boutilier and Bistarelli′s related works.

Key words Conditional preference networks (CP-nets), expressive power, strong dominance testing, preferences completeness, improved Warshall algorithm, multiple attribute qualitative decision under incomplete information, soft constraint satisfaction problem (SCSP)

偏好决定选择, 因此在分布式人工智能中, 当多个 Agent 相遇并进行合作求解时, 理解 Agent 彼此的偏好并推理出彼此对各种事物的感兴趣程度, 进而采取有利于合作的方式来进行决策[1−2] 是非常重要的. Agent 的偏好决定了双方合作求解的方式及其速度, 因此在人工智能中研究 Agent 偏好很有意义.

偏好在日常生活中随处可见. 我们在购买汽车时, 许多人 (用 Agent a 表示) 的一个简单准则是便宜的比昂贵的好, 同样价钱下, 空间大的比空间小的好. 而有的人 (用 Agent b 表示) 的偏好则不同, 他可能首先认为安全的比不安全的好等. 因此汽车经销商需要根据客户的偏好描述推理出其对汽车的需求原则, 从而给出客户最需要的一些汽车配置, 以满足客户的需求.

本文研究偏好处理的基础 — 条件偏好网 (Con-

收稿日期 2010-05-24 录用日期 2010-10-13
Manuscript received May 24, 2010; accepted October 13, 2010
1. 烟台大学计算机学院 烟台 264005
1. School of Computer Science and Technology, Yantai University, Yantai 264005

ditional preference networks, CP-nets)[3] 的表达能力, 以期为参考文献中的相关研究提供数学基础. 第 1 节给出了偏好的研究内容和存在的问题; 第 2 节给出了基于 CP-nets 的偏好处理元模型的几个方面; 第 3 节详细讨论基于 CP-nets 的几种典型运算及其复杂度; 基于第 2 节和第 3 节的内容, 第 4 节给出了 CP-nets 的表达能力的描述. 为了解决 CP-nets 在强占优测试上的缺陷, 在第 5 节将其归约成受限半环 (C-semiring)[4−5] 上的解的优劣判断问题, 结合约束满足问题的求解方法[6] 来降低部分 CP-nets 的求解复杂度, 并给出相关工作的对比. 第 6 节给出了结论和未来工作.

本文的主要特色和贡献在于: 1) 基于 CP-nets 导出图上的跳变关系所对应的稀疏关系矩阵, 利用改进的 Warshall 算法求出了二值 CP-nets 上的强占优测试的时间复杂度为 $\mathrm{O}(4^n)$, 解决了文献 [3] 没有解决的任意结构 CP-nets 上的强占优测试算法及时间复杂度问题, 即阐述了强占优测试问题在理论上是可解的; 2) 首次从适用的范围、模型的获取、语言上所定义的运算及其复杂度等方面研究了 CP-nets 的表达能力. 指出 CP-nets 适合不完全信息下的定性偏好决策. 倘若要得到完全的偏好信息时, 可通过与 Agent 的交互计算来完成, 这间接提高了 CP-nets 的表达能力; 3) 鉴于 CP-nets 上的强占优测试的指数复杂度在实践上的不可行性, 本文将无环 CP-nets 上的强占优测试在多项式时间归约为软约束的满足问题来求解. 即本文不仅给出了文献 [3] 没有给出的强占优测试的理论解, 还给出了一个实际可行的 (复杂度低的) 部分解或近似解.

1 相关工作

1.1 Agent 偏好处理的研究内容

偏好处理是近几年人工智能的研究热点. 2004 年发表在世界顶级人工智能期刊 *Journal of Artificial Intelligence Research* 上的 Boutilier 等[3] 的论文在 2009 年被评为近 5 年最具影响力的一篇论文, 因为它开辟了一个新的研究方向. 2009 年世界顶级人工智能会议 IJCAI 2009 上的杰出论文是 Koriche 等[7] 的有关偏好的学习论文. 通过仔细阅读近几年人工智能会议及期刊的一些论文, 结合偏好的应用, 我们可把 Agent 的偏好研究归为如下 4 类.

1) 偏好的表示 (Preference representation): 研究如何将 Agent 的偏好用数学方式加以描述, 并重点探讨这种描述方式的简洁性, 易用性及其所用语言的表达能力; 2) 偏好的提取 (Preference elicitation): 研究如何通过与 Agent 交互来学习和推理 Agent 的偏好. 该方向是以机器学习的方式来获取 Agent 的偏好. 其中, Koriche 等[7] 提出了利用等价查询 (Equivalence query) 和成员查询 (Membership query) 来获取 Agent 的条件偏好. Lang 等[8] 提出了如何学习最简单的一种偏好—属性可分离的偏好 (属性之间没有依赖关系, 即 CP-nets 由一些孤立的点组成). Conitzer[9] 提出了利用比较查询的方法来提取 Agent 的单峰偏好, Chevaleyre 等[10] 给出了多属性域偏好学习的通用准则, 给出了哪些实例是可学习的, 哪些实例是不能学习的; 3) 偏好的聚合 (Preference aggregation): 研究如何将多个 Agent 偏好聚合成一个集体的偏好, 其本质是政治生活中的选举, 其核心是设计一种选举制度, 使得选举科学、民主、高效和公平. Tang 等[11] 通过程序验证了选举问题中的 “阿罗不可能定理” (Arrow impossible theorem), 即在具有 2 个投票者, 3 个被选举人的环境中, 任何投票规则都会出现独裁者. 但该文没有给出投票所应遵循的几个原则之间的相关性问题, 且所给的民主性质是否完备也不知道. Conitzer 等[12] 给出了在具有少量被选举人的情况下, 什么时候操纵选举是困难的问题, 其目的是避免选举中的舞弊行为. Zuckerman 等[13] 给出了 Agent 联盟在选举中的操纵问题; 4) 偏好的推理 (Preference inference): 研究如何从已有的偏好模型和语言中, 利用初始的偏好断言及 Agent 内嵌的推理规则, 推理出 Agent 关于某事物的断言是否一致, 是否还有哪些隐含的偏好等. 其中, Domshlak 等[14] 提出了带有硬约束和软约束的偏好的推理方法, 张志政等[15] 提出并构造了一个能够描述和推理多种类型偏好的逻辑系统 (Logic of many kinds of preference, MPL), 通过直接引入 4 个逻辑偏好算子构造了能够表示 4 类偏好的 MPL 语言 LMPL, 基于包含、交、补 3 个集合操作给出了偏好推理的规律.

在偏好处理的 4 个方面研究中, 偏好表示是最基础的工作. 目前偏好的应用大多都是基于 CP-nets 来加以描述[3, 7−8, 10, 14]. 其中最典型的是 Boutilier 的工作[3], 他对前人工作进行了总结, 详细描述了 CP-nets 的语法、语义及应用. 然而该文并没有探讨 CP-nets 的表达能力, 且其上的强占优测试 (Strong dominance testing) 在最坏情况下的算法及复杂度都没有给出. 近几年出现的一些新的偏好表示方式都是在 CP-nets 基础上产生. 例如 Brafman 等[16] 就提出了一种带有属性重要性的 TCP-net, 该模型在处理属性之间的偏好关系基础上, 增加了反映属性重要性的元素, 然而它的分析技术和 CP-nets 的一样. Rossi[17] 提出包含多个 Agent 偏好模型的 mCP-net, 它实质上是多 Agent 的偏好聚合模型,

因此这些模型都是特殊的一类 CP-nets.

1.2 偏好表示存在的问题

偏好表示是偏好处理的基础, 而偏好表示大多利用 CP-nets 来表示, 虽然对 CP-nets 的研究很多, 但是 CP-nets 的表达能力如何, 即它能表达什么样的偏好, 适合解决什么样的问题等, 其上运算的复杂度如何等, 其表达的偏好是否完备等却很少有人研究. 本文从偏好处理的元模型出发, 从模型的数学结构, 语言的表达能力, 运算的复杂度等方面研究 CP-nets 的表达能力, 给出它适合描述的问题及适合解决的问题. 另外还设计了一个可解决任意形状 CP-nets 的强占优测试算法, 给出了一种可得到部分解或近似解的一种归约技术, 这些都间接提高了部分 CP-nets 的表达能力.

2 CP-nets 的语言与语义

2.1 偏好处理的元模型

首先从 Brafman 等[18] 给出的图 1 所示的偏好处理的元模型出发, 逐步引入 CP-nets 及其表达能力的相关论述.

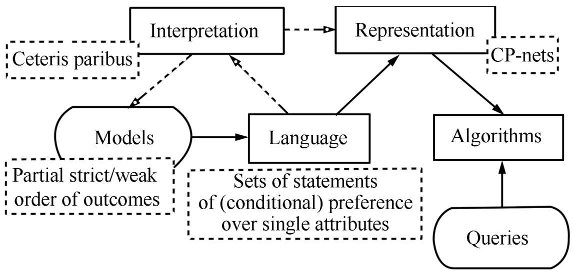


图 1 偏好处理的元模型

Fig. 1 The meta-model of preference handing

偏好处理的元模型主要包含如下元素: 偏好的模型 (Models), 偏好的语言 (Language), 用户对偏好的查询 (Query) 和实现用户对偏好查询的各种算法 (Algorithm). 下面围绕这些方面来描述基于 CP-nets 的偏好处理.

2.2 偏好的模型 — 严格偏序关系

模型就是一个数学结构, Agent 的偏好是用数学结构 — 严格偏好结构来描述.

定义 1. 设 $V=\{X_1,X_2,\cdots,X_n\}$ 是决策属性 (Attribute) 或变量的集合, $Dom(X_i)$ 代表属性 X_i 的有限定义域, 则决策空间 $\Omega=\times_{i=1}^{n}Dom(X_i)$ 表示所有属性的可能组合. o 是决策空间的一个配置 (Outcome), 代表决策空间的一种组合. 若两个配置 o 和 o' 仅有一个属性值不同, 而其他属性值都相同, 则称 o 和 o' 为可交换的配置 (Swap outcome).

现实中决策的两个属性之间可能具有依赖关系, 如果 Agent 对属性 X_i 的偏好取决于 X_j, 则称 X_j 是 X_i 的一个父亲, 用 $Para(X_i)$ 表示 X_i 的父亲, 一个属性可能有多个父亲.

定义 2. $>$ 是决策空间上的二元关系, 1) 若 $>$ 反对称, 即 $(\forall o,o'\in\Omega)$ $((o>o'\wedge o\neq o')\rightarrow o'\not> o)$; 2) 传递, 即 $(\forall o,o',o''\in\Omega)$ $((o>o'\wedge o'>o'')\rightarrow o>o'')$. 即 $>$ 是一个严格偏序关系时, 称 $>$ 为 Ω 上的偏好关系.

定义 3. 偏好处理的模型 (Preferences handling model, PHM) 是一个数学结构, 其形式化为 $\mathrm{PHM}=\langle\Omega,>\rangle$, 其中 Ω 是定义 1 中的决策空间, $>$ 是定义 2 的偏好关系.

2.3 偏好的语言及 CP-nets

偏好处理语言表达 Agent 对决策属性的偏好断言, 在 CP-nets 中, 通过决策属性的条件偏好表 $CPT(X_i)$ 来描述偏好语言.

定义 4. 设 $CPT(X_i)$ 为属性 X_i 的条件偏好表 (Condition preference table), 它表示属性 X_i 在其父属性 $Para(X_i)$ 的不同取值下, Agent 对 $Dom(X_i)$ 集合的一个偏好. 之所以称 $CPT(X_i)$ 为条件偏好表, 是因为 $Para(X_i)$ 在不同取值下, Agent 对属性 X_i 的几个取值的偏好排序也不同. 在 $Para(X_i)$ 的所有取值下, Agent 对属性 X_i 取值的偏好排序组成条件偏好表 $CPT(X_i)$.

定义 5. CP-nets 是一个有向图 $N=\langle V,CE\rangle$, 其中 V 是顶点集 (如定义 1 所示), CE 为有向边集, 代表所有属性之间的依赖关系. 即一条有向边起点的取值影响着终点取值的偏好 (边的源点取值确定了终点不同值的一种排序). 对于每一个顶点 X_i, 都有一个条件偏好表 $CPT(X_i)$ 与其关联.

定义 6. 偏好处理的语言 (Preferences handling language, PHL) 是 Agent 在每个决策变量 X_i 上的条件断言的并, 即 $\mathrm{PHL}=\bigcup_{X_i\in V}CPT(X_i)$.

为理解 CP-nets 的语法、语义及对其进行的各种操作, 下面给出一个典型的 "晚会穿衣" 实例[3]. 本文所有概念、性质与定理都可通过它来理解.

例 1. 某人参加晚会的穿衣主要考虑夹克 (Jacket), 裤子 (Pants) 和衬衫 (Shirt), 分别用 J, P 和 S 来代表. 对于夹克和裤子来说, 他无条件喜欢黑色, 而不是白色. 而穿什么颜色的衬衫取决于夹克和裤子颜色的组合. 当夹克和裤子是同一种颜色, 则他偏好红色而不是白色衬衫, 当夹克和裤子颜色不同时, 则偏好白色而不是红色衬衫. 该例子的 CP-nets 图 $N=\langle V,CE\rangle$ 如图 2 所示. 其中, $V=\{J,P,S\}$, $Dom(J)=\{J_b,J_w\}$, $Dom(P)=\{P_b,P_w\}$, $Dom(S)=\{S_r,S_w\}$; $CE=\{\langle J,S\rangle,\langle P,S\rangle\}$. 各顶点的偏好表在图 2 中.

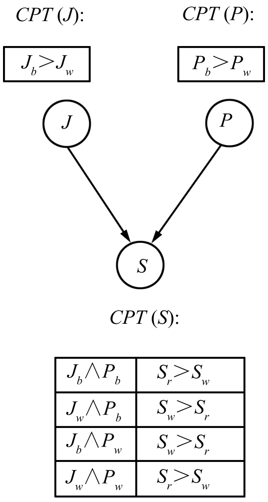


图 2 "晚会穿衣" 的 CP-nets

Fig. 2 CP-nets for "evening dress"

2.4 偏好语言的语义

Agent 给出了条件偏好语言 $CPT(X_i)$ 后, 如何将其映射或解释为偏好处理的模型呢? 为此引入 Ceteris paribus (all else being equal) 语义[3], 利用该语义, Agent 不用反省 (不用思考, 下意识地) 就可得到其含义. 该语义是指, 对于一个配置 o 来说, 主要关心除某个属性 X_i 的值不同, 而其他属性值都相同的话, Agent 对属性 X_i 不同取值的一种偏好断言. 利用该语义, 很容易对可交换的两个配置给出偏好断言. 因为可交换的配置的对比本质上是一个属性值的对比. 这正如对两台电脑的配置作对比时, 若两台电脑只有一项配置不同 (一个属性值不同), 而其他配置都一样的话 (其他属性值都一样), 任何人不用思考就可给出谁优谁劣的判断.

例 2. 考虑例 1 的实例, "$j \wedge p$: $s_1 > s_2$" 断言可用 Ceteris Paribus 语义解释为: 当其他属性值都一样的话, 若 $J = j$, $P = p$ 时, 则 Agent 对 $S = s_1$ 的偏好大于对 $S = s_2$ 的偏好. 显然从图 2 中 $CPT(S)$ 的第 1 行可知, j 为 J_b, p 为 P_b, s_1 为 S_r, s_2 为 S_w. 与 Ceteris paribus 语义有关的概念是如下的条件偏好无关.

定义 7. 设 X, Y 和 Z 是决策属性集 V 的一个划分[18] (即 $X \cap Y = \phi$, $X \cap Z = \phi$, $Y \cap Z = \phi$, $X \cup Y \cup Z = V$), 若对于 Z 的一个赋值 z 和任意的 x_1, $x_2 \in Dom(X)$, 任意的 y, $y' \in Dom(Y)$, $x_1yz > x_2yz$, 蕴含着 $x_1y'z > x_2y'z$, 则称属性集 X 与 Y 在条件 Z 的一个赋值 z 下偏好无关[3].

例 3. 假定一个 CP-nets 图 $N = \langle V, CE\rangle$, 其中, $V = \{A, B, C, D, E\}$, $CE = \{\langle A, C\rangle, \langle B, C\rangle, \langle C, D\rangle, \langle D, E\rangle\}$, $Dom(A) = \{a_1, a_0\}$, $Dom(B) = \{b_1, b_0\}$, $Dom(C) = \{c_1, c_0\}$, $Dom(D) = \{d_1, d_0\}$, $Dom(E) = \{e_1, e_0\}$, $CPT(A)$ = "$a_1 > a_0$", $CPT(B)$ = "$b_1 > b_0$", $CPT(C)$ = "$(a_1 \wedge b_1) \vee (a_0 \wedge b_0)$: $c_1 > c_0$, $(a_1 \wedge b_0) \vee (a_0 \wedge b_1)$: $c_0 > c_1$", $CPT(D)$ = "c_1: $d_1 > d_0$, c_0: $d_0 > d_1$", $CPT(E)$ = "d_1: $e_0 > e_1$, d_0: $e_1 > e_0$". 则属性集 $\{D\}$ 与属性集 $\{A, B, E\}$ 在 $\{C\}$ 的一个赋值 ($C = c_1$ 或 c_0) 下条件偏好无关; 属性集 $\{C\}$ 与属性集 $\{D, E\}$ 在 $\{A, B\}$ 的一个赋值 ($AB = a_1b_1$ 或 a_1b_0 或 a_0b_1 或 a_0b_0)) 下条件偏好无关.

定理 1. 设 $N = \langle V, CE\rangle$ 是 CP-nets, 对于可交换的配置 o_1 与 o_2, 则 Agent 对 o_1 与 o_2 的偏好比较可在 $\mathrm{O}(n)$ 内完成, 即在 $\mathrm{O}(n)$ 内确定出 $o_1 > o_2$, 或者 $o_2 > o_1$.

证明. 因为可交换的两个配置 o_1 与 o_2 只有一个属性值不一样, 首先可在线性时间内确定出哪个属性值不一样, 而其他属性值都一样 (比较 o_1 与 o_2 首个不相同的字符, 而后确定其对应的决策属性); 当确定出值不同的单属性变量 X_i 后, 求出其父亲 $Para(X_i)$, 此时 o_1 与 o_2 偏好比较对应 $CPT(X_i)$ 中的一条记录, 显然从 $CPT(X_i)$ 中可确定 Agent 对 o_1 与 o_2 的偏好强弱. □

例如例 1 中, 若 o_1 = "$J_bP_bS_w$", o_2 = "$J_bP_bS_r$", 则在 $\mathrm{O}(n)$ 内可确定 o_1 与 o_2 在属性 S 处的值不同; S 的父亲是 $\{J, P\}$, 因此 o_1 与 o_2 的偏好比较就是在 $CPT(S)$ 中寻找 "J_bP_b" 这一行的过程, 显然在第 1 行包含断言 "$J_b \wedge P_b$: $S_r > S_w$", 因此对 o_2 的偏好大于对 o_1 的偏好, 即 $o_2 > o_1$.

3 CP-nets 上的几种运算

利用 CP-nets 可对 Agent 的偏好进行描述, 在 CP-nets 上定义一些运算, 就成为实用的偏好描述语言, 且可对该模型所能表达的配置进行优劣比较. 其实图 1 给出 "Algorithm" 就是指该模型给用户提供了哪些算法, 以满足用户对模型查询的需要. 要想给出这些算法的实现及其复杂度, 首先引入 CP-nets 的导出图.

3.1 CP-nets 的导出图及强占优测试

3.1.1 CP-nets 的导出图及性质

根据定理 1, Agent 可快速在可交换的两个配置间作出偏好判断. 那么如何对任意两个配置作出偏好判断? 为此引入 CP-nets 的导出图.

定义 8. 设 $N = \langle V, CE\rangle$ 是一个 CP-nets, 则有向图 $N' = \langle \Omega, IE\rangle$ 是 N 的导出图, 其中 IE 是可交换的配置所构成的有向边集, 且对有向边终点的偏好大于对有向边起点的偏好.

定理 2. $N' = \langle \Omega, IE\rangle$ 是二值 CP-nets 图 N 的导出图, 即在 N 中 $|V| = n$, $|Dom(X_i)| = 2$, 则:

1) $|\Omega| = 2^n$, $|IE| = n \times 2^{n-1}$. 即 N' 中顶点个数是 2^n, 边的个数是 $n \times 2^{n-1}$.

2) $\forall o \in \Omega$, $Degree(o) = n$ (每个顶点的度都是 n).

证明. 1) 先证 $|\Omega| = 2^n$. 因为 $|Dom(X_i)| = 2$, 而 $\Omega = Dom(X_1) \times Dom(X_2) \times \cdots \times Dom(X_n)$, 因此 $|\Omega| = |Dom(X_1)| \times |Dom(X_2)| \times \cdots \times |Dom(X_n)| = 2^n$.

再证 $|IE| = n \times 2^{n-1}$. 对于任意一个配置 o 来说, 由于 o 中包含 n 个属性, 且每个属性有两种取值, 故与 o 可交换的配置有 n 个, 另外 Ω 中共有 2^n 个配置, 而每个配置有 n 个可交换配置, 因此总共可交换的配置是 $n \times 2^n$ 个. 但由于可交换的配置具有对称性, 而 N' 中的边为有向边, 两个可交换的配置在 N' 中仅有一条有向边相连, 故总共边数是 $n \times 2^n/2 = n \times 2^{n-1}$.

2) 因为 $|V| = n$, 即一个配置 o 含有 n 个属性, o 共有 n 个可交换的配置, 所以 o 的度为 n. □

例如图 3 就是图 2 的导出图, 其中顶点个数是 8, 边的个数是 12, 每个顶点的度都是 3, 满足定理 2.

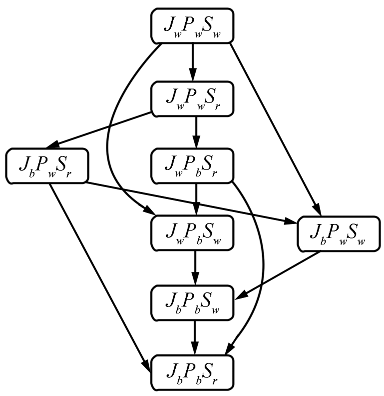


图 3 CP-nets 的导出图

Fig. 3 The induced graph of CP-nets

3.1.2 跳变关系及强占优测试

定义 9. 设 $N' = \langle \Omega, IE \rangle$ 是 CP-net 图 N 的导出图, 对于 $o, o' \in \Omega$, 若从顶点 o' 到顶点 o 可达, 即存在着一条路径连接顶点 o' 和 o, 则称 o 强占优 o', 记作 $N \models o > o'$. 判断 $N \models o > o'$ 是否成立的测试称作强占优测试.

定义 10. 若两个配置 o 和 o' 可交换, 且 $N \models o > o'$, 则称 o' 和 o 具有跳变关系 (Flip relation, FR), 即 $o'FRo$.

事实上, 跳变关系 FR 就是 N' 中的边集 IE, 即 $FR = IE$. 跳变关系给出了可交换的配置之间的强占优关系, 但不是可交换的配置也可能具有强占优关系. 由于强占优关系具有传递性, 因此对 FR 或 IE 关系求传递闭包[19], 即可求出所有的强占优关系.

3.1.3 CP-nets 的强占优测试算法

Warshall 算法是基于关系矩阵运算而得到关系传递闭包的一个算法[19], 当关系矩阵是 $m \times m$ 的矩阵时, 其时间复杂度为 $O(m^3)$. 由于 CP-nets 导出图 N' 的关系矩阵是稀疏矩阵, 我们可进一步将其复杂度降低为 $O(m^2)$. 改进的 Warshall 算法的完整描述如算法 1 所示, 可认为它是 CP-nets 的强占优测试算法.

算法 1. 求强占优测试的改进的 Warshall 算法

输入. 导出图 $N' = \langle \Omega, IE \rangle$ 中 IE 的关系矩阵 $A_{m \times m}$, 其中 $m = 2^n$.

输出. IE 关系的传递闭包的关系矩阵 $B_{m \times m}$.

步骤 1. //求矩阵 A 所有行中非零元素所在的列号于数组 C

For $i = 1$ to m Do
For $j = 1$ to m Do
If $(A[i, j] ==1)$ Then $C[i] = C[i] \cup \{j\}$
//将关系矩阵中 i 行为 1 的列号保存在数组 $C[i]$ 中;

步骤 2. //把 $C[j]$ 中出现的元素所在的集合都并到 $C[j]$ 中

For $i = 1$ to m Do
For $j = 1$ to m Do
If $(i \in C[j]$ and $(j \neq i)$)
Then $C[j] = C[j] \cup C[i]$
//把 $C[j]$ 中元素所在的集合并入 $C[j]$, 其本质是把第 i 行加到第 j 列为 1 的元素所在的那些行中;

步骤 3. // 根据数组 C 的值设置传递闭包矩阵 B

For $i = 1$ to m Do
For $j = 1$ to m Do
If $(j \in C[i]$ and $(B[i, j] == 0))$
Then $B[i, j] = 1$
//将顶点 i 和顶点 j 有强占优关系, 则置 B 矩阵的 i 行 j 列为 1.

例 4. 利用算法 1 求图 3 中 $N' = \langle \Omega, IE \rangle$ 的传递闭包. 其中 $\Omega = \{o_1, o_2, o_3, o_4, o_5, o_6, o_7, o_8\}$, $o_1 =$ "$J_wP_wS_w$", $o_2 =$ "$J_wP_wS_r$", $o_3 =$ "$J_wP_bS_r$", $o_4 =$ "$J_wP_bS_w$", $o_5 =$ "$J_bP_bS_w$", $o_6 =$ "$J_bP_bS_r$", $o_7 =$ "$J_bP_wS_r$", $o_8 =$ "$J_bP_wS_w$"; $IE = \{\langle o_1, o_2 \rangle, \langle o_2, o_3 \rangle, \langle o_3, o_4 \rangle, \langle o_4, o_5 \rangle, \langle o_5, o_6 \rangle, \langle o_1, o_4 \rangle, \langle o_1, o_8 \rangle, \langle o_2, o_7 \rangle, \langle o_3, o_6 \rangle, \langle o_7, o_6 \rangle, \langle o_7, o_8 \rangle, \langle o_8, o_5 \rangle\}$.

解. 矩阵
$$A = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

1) 执行完步骤 1 的双层循环后, 数组 C 的数值如下: $C[1]=\{2,4,8\}$, $C[2]=\{3,7\}$, $C[3]=\{4,6\}$, $C[4]=\{5\}$, $C[5]=\{6\}$, $C[6]=\Phi$, $C[7]=\{6,8\}$, $C[8]=\{5\}$.

2) 算法的步骤 2 的执行步骤较多, 执行完后数组 C 的数值为: $C[1]=\{2,3,4,5,6,7,8\}$, $C[2]=\{3,4,5,6,7,8\}$, $C[3]=\{4,5,6\}$, $C[4]=\{5,6\}$, $C[5]=\{6\}$, $C[6]=\Phi$, $C[7]=\{5,6,8\}$, $C[8]=\{5\}$.

3) 步骤 3 执行完后传递闭包矩阵为

$$B=\begin{bmatrix} 0&1&1&1&1&1&1&1\\ 0&0&1&1&1&1&1&1\\ 0&0&0&1&1&1&0&0\\ 0&0&0&0&1&1&0&0\\ 0&0&0&0&0&1&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&1&1&0&1\\ 0&0&0&0&1&0&0&0 \end{bmatrix}$$

此时传递闭包矩阵 B 中元素 1 的个数是 23, 代表 CP-nets 可表达的强占优个数为 23. 而图 3 中总共应该有 28 对可比较的配置, 但利用图 2 的 CP-nets 只表达了 23 对强占优关系, 其余 5 对没有作阐述, 即图 2 的 CP-nets 对偏好的表示不完备.

3.2 弱占优测试

从第 3.1 节可知, CP-nets 所能表达的强占优关系有限, 当 CP-nets 所表达的偏好不完备时, 如何增强 CP-nets 的表达能力呢? 为此引入弱占优测试.

定义 11. 设 $N'=\langle\Omega, IE\rangle$ 是无环 CP-nets 图 N 的导出图, 若从顶点 o 出发的所有路径都不能到达 o' ($o'\not\succ o$), 则称 o 弱占优 o', 记作 $N\nvDash o'>o$. 判断 $N\nvDash o'>o$ 是否成立的算法称作弱占优测试.

o 弱占优 o' 的含义是 o' 不比 o 强. 显然 o 强占优 o', 则 o' 不可能强占优 o. 但 o 弱占优 o', 则 o' 也可能弱占优 o. 弱占优关系的作用在于对配置进行拓扑排序. 当 $N\nvDash o'>o$ 时, 则对不同配置的产品排序时, o 可以排在 o' 的前面. 换句话说, 在关于 Agent 的偏好断言中, 没有足够的信息确定 o 和 o' 谁强谁弱, 但可确定谁不比谁强.

定理 3. 设 N 是无环 CP-nets, $\forall o, o'\in\Omega$, $N\models o>o'$ 是 $N\nvDash o'>o$ 的充分条件, 但不一定是必要条件.

证明. $N\models o>o'$ 说明在 N 的导出图 N' 中, 存在着一条从 o' 到 o 的路径, 由于 N 无环, 则不存在一条从 o 到 o' 的路径, $N\nvDash o'>o$; 但 $N\nvDash o'>o$ 也可能表示 o' 和 o 不可比较, 因此 $o>o'$ 也可能不成立, 即由 $N\nvDash o'>o$ 得不出 $N\models o>o'$. □

定理 4. N 是无环 CP-nets, o 和 o' 是 Ω 中两个不同的配置, 若在 N 中存在一个属性 X, 使得 o 和 o' 的属性 X 的所有祖先值[19] 相同, 即 $o[Ance(X)]=o'[Ance(X)]$, 但 $o[X]>o'[X]$, 则 $N\nvDash o'>o$. 其中 $Ance(X)$ 表示 X 的所有祖先.

证明. 通过构造法来证明. 由于 N 是无环图, 因此存在一个顶点 Y_1, 其入度为 0 (可看作根顶点). 然后由 $CPT(Y_1)$ 比较 Agent 对 $o[Y_1]$ 和 $o'[Y_1]$ 的偏好大小, 若 $o[Y_1]>o'[Y_1]$, 即 o 在 Y_1 属性上的值优于 o' 在 Y_1 属性上的值, 则 o' 不会比 o 强, 即 $N\nvDash o'>o$; 同理若 $o'[Y_1]>o[Y_1]$, 则 $N\nvDash o>o'$; 若 $o[Y_1]=o'[Y_1]$, 则在 N 中删除顶点 Y_1, 形成新的 CP-nets 图 N_1, 而后在 N_1 中寻找根顶点 Y_2, 再根据 $CPT(Y_2)$ 比较 Agent 对 $o[Y_2]$ 和 $o'[Y_2]$ 的偏好大小. 若 $o[Y_2]>o'[Y_2]$, 同理得到 $N\nvDash o'>o$; 若 $o'[Y_2]>o[Y_2]$, 则 $N\models o>o'$; 若 $o[Y_2]=o'[Y_2]$, 则在 N 中删除顶点 Y_2, 形成新的 CP-nets 图 $N_2,\cdots$, 依次类推. 由于 o 和 o' 是不同的配置, 因此必定能找到首个不相同的属性值, 并根据条件偏好表判断该属性值的优劣. □

例 5. 考虑例 3 中的 CP-nets, 假设配置 $o_1=$ “$a_1b_1c_1d_1e_1$”, $o_2=$ “$a_1b_1c_0d_1e_0$”, $o_3=$ “$a_1b_1c_0d_0e_1$”, 由于 $o_1[Ance(C)]=o_2[Ance(C)]=$ “a_1b_1”, 而从 $CPT(C)$ 可知, $o_1[C]>o_2[C]$. 因为 $o_1[C]=c_1$, $o_2[C]=c_0$, 在 C 的条件偏好表中有一行记录 “$a_1\wedge b_1$: $c_1>c_0$”. 根据定理 4 有 $N\nvDash o_2>o_1$. 同理有 $N\nvDash o_2>o_3$.

基于定理 4 可给出算法 2 所示的 CP-nets 的弱占优测试算法, 它是基于 CP-nets 图 N 的搜索算法.

算法 2. 弱占优测试算法

输入. $N=\langle V, CE\rangle$ 和配置 $o, o'\in\Omega$.
输出. $N\nvDash o'>o$ 或 $N\nvDash o>o'$.

```
U = V
While (U ! = NULL) Do
{
Y = GetRoot(U) //取得 U 的根顶点
If (o[Y] > o′[Y]) Then N ⊭ o′ > o    break
If (o′[Y] > o[Y]) Then N ⊭ o > o′    break
If (o[Y] == o′[Y]) U = U − Y
}
End While
```

定理 5.

1) 二值 CP-nets 图 N 的强占优测试 $N\models o>o'$ 算法的时间复杂度为 $O(4^n)$;

2) 二值 CP-nets 图 N 的弱占优测试 $N\nvDash o'>o$ 算法的时间复杂度为 $O(n)$.

证明. 1) 因为算法 1 可在 $O(m^2)$ 时间内求出

关系 IE 的传递闭包, 其中 m 为关系图中顶点的个数. 由于 N' 中的顶点个数是 2^n, 此时 $m=2^n$, 故算法 2 可在 $\mathrm{O}(4^n)$ 求出其传递闭包, 倘若配置 $\langle o,o'\rangle \in IE^*$, 则 o' 强占优 o. 2) 略. □

3.3 基于弱占优的拓扑排序

现实中的决策空间有指数级个配置, 希望 Agent 对任何两个配置作出强占优判断是不可能的, 因为 Agent 需要进行 $2^{n-1}\times(2^n-1)$ 次判断. 倘若 $n=10$, 则进行约 50 万次的偏好判断, 这显然不可能. 另外, 当两个配置之间有多个属性值不一样的话, Agent 会面临 "鱼和熊掌不可兼得的局面", 因此很难做出快速而又一致的判断. 但基于 CP-nets 的 Ceteris paribus 语义, Agent 可进行条件偏好无关的断言, 而后利用弱占优测试来对感兴趣的一些配置进行排序.

由于偏好是一种偏序, 因此对所有配置进行全序排列不可能, 为此我们可使用拓扑排序. 另外, 虽然决策空间很大, 但现实中往往只关心其一个子集 T. 例如, 捷达轿车在大众 4S 店仅有十种左右配置 (其他配置的不生产), 客户购买汽车时也只是在 T 子空间中进行决策, 为此可基于弱占优查询来实现 T 的拓扑排序. T 中配置进行拓扑排序后, 就可找出最优配置和最差配置了.

定理 6. 设 $T\subseteq\Omega$ 是 Agent 感兴趣的配置集, 则基于弱占优测试可对 T 进行拓扑排序, 且拓扑排序的复杂度为 $\mathrm{O}(n\times p\log(p))$. 其中 $p=|T|$ 为感兴趣的配置的个数.

证明. 显然 $N\nvDash o'>o$ 时, o 可排在 o' 前面. 利用快速排序 (Quick sort) 算法可在 $|T|\times\log(|T|)$ 时间内将 $|T|$ 个配置排好序, 而排序所进行的比较操作 (弱占优测试) 需进行 $|V|=n$ 次, 因此拓扑排序操作的次数是 $|V|\times|T|\times\log(|T|)=n\times p\log(p)$, 故拓扑排序的复杂度为 $\mathrm{O}(n\times p\log(p))$. □

定理 6 表明, 基于弱占优测试对感兴趣的配置进行排序为多项式时间复杂度, 因此其在实践中可行.

4 CP-nets 的表达能力

4.1 表达能力的衡量方法

基于模型产生的语言的表达能力如何, 直接决定了用户使用该语言所能解决的问题的能力和复杂度. 那么我们希望该语言具有什么样的特性, 即什么样的语言的表达能力较强? 为此我们给出判断偏好表达能力强弱的如下标准: 1) 模型是否简洁而又清楚地表达 Agent 的偏好 (即能否对偏好说明白); 2) 给出的偏好模型是否让别的 Agent 不用反省就能听明白 (即能否听明白 Agent 表达的偏好); 3) 语言中是否有足够的算法来实现用户对偏好的各种查询 (即提供偏好处理的算法是否充分); 4) 实现偏好测试的算法的可行性如何 (即时间复杂度是否太大); 5) 模型可表达的断言和实际中的断言数量相差多少 (即表达的偏好是否完备); 6) 偏好语言的适用范围是否宽泛 (即适合解决一个还是一类偏好的处理问题).

4.2 CP-nets 表达能力的结论

基于表达能力的 6 个衡量方法, 下面依次给出 CP-nets 的表达能力的对应描述及相应性质与定理. 1) 对于外行用户来说, 利用 CP-nets 很容易表达它对事物的偏好断言, 即直观性很强. 因为 CP-nets 中的 $CPT(X_i)$ 是 Agent 的偏好表示语言, 其对偏好的断言基于单属性 X_i 值不同而其他属性值都相同的配置的偏好比较. 所以利用 CP-nets, Agent 能正确而直观地表达自己的偏好, 即基于 CP-nets, Agent 能将其欲表达的偏好说明白. 2) 对于可交换的两个配置来说, Agent 可在线性时间内得出其强占优关系, 因此别的 Agent 能听明白两个可交换配置的强占优关系, 对于任意两个配置来说, 也可在线性时间内得出其弱占优关系. 倘若要对任意两个配置确定其强占优关系, 则基于 CP-nets 的偏好表示却需指数级的时间才可得到. 即使强占优测试的复杂度很高, 但还有些配置之间的强占优关系本身就没有表达出来 (不可比的配置, 从例 3 可知). 因此基于 CP-nets 的偏好表示, 听者能部分听清楚、听明白 (判断可交换的配置间的强占优关系); 还有大部分能听明白 (导出图的传递闭包包含了大多数的强占优关系), 但听起来费劲 (求传递闭包需指数时间复杂度), 还有部分偏好听不明白 (部分强占优关系 CP-nets 没有表达), 因为 CP-nets 的模型本身就没有表达. 3) 基于 CP-nets 的偏好表示可实现的算法有强占优测试, 弱占优测试, 拓扑排序, 求最优配置, 最差配置 (拓扑排序后的第 1 个和最后 1 个元素). 这对于不完全信息下的偏好处理是充分的, 即基于这些算法, Agent 可查询得到配置间的任何比较信息. 4) CP-nets 上的算法除强占优测试的复杂度很高外, 其他算法都是多项式时间复杂度. 因此基于 CP-nets 的偏好表示中的算法在理论上是可行的 (都能求出时间复杂度), 在实践中大部分是可行的 (时间复杂度是多项式级). 5) CP-nets 能表达的偏好关系就是 N 的导出图 N' 所能表达的信息. 从例 3 可看出 CP-nets 所表达的偏好信息不完备. 但是 CP-nets 理论上所能表达的偏好信息和实际应该表达的偏好信息相差多少, CP-nets 图 N 的结构与其所表达的偏好关系是什么, 这还是一个悬而未决的问题, 也是后续工作继续探求其表达能力所努力的

方向. 6) CP-nets 的适用范围比较宽泛, 尤其适合于定性条件下的偏好处理. 由于现实中的许多偏好难以定量表达, 这为利用 CP-nets 处理定性偏好提供了舞台. 将上述 6 条表达能力的直观描述加以总结, 可到达如下关于 CP-nets 表达能力的两个性质和四个定理.

性质 1 (偏好表达的直观性). 利用 CP-nets 图来表达 Agent 对配置的偏好是直观的 (Intuitive).

证明. 根据定义 5 及图 2 给出的一个实例可知, CP-nets 是一种图形化的偏好表示工具, 其直观性体现在如下 3 点: 1) 顶点集 V 是决策的变量集, 因此从 V 可知配置包含哪些决策变量; 2) 边集确定了决策变量之间的依赖关系, 一个顶点的父亲集合决定了该顶点的偏好, 例如图 2 中 J 和 P 的不同取值, 确定了对 S 的偏好; 3) 利用条件偏好表容易表达可交换配置之间的强占优关系. 对于例 1 来说, 顶点 J 的条件偏好表中有一个条目为 "$J_b > J_w$", 因此根据 Ceteris Paribus 语义, 可产生如下 4 对可交换配置之间的强占优关系 $J_bP_bS_r > J_wP_bS_r$, $J_bP_bS_w > J_wP_bS_w$,$J_bP_wS_r > J_wP_wS_r$, $J_bP_wS_w > J_wP_wSw$. 同样很容易根据其他顶点的条件偏好表生成可交换配置的强占优关系. □

性质 2 (偏好理解的部分简易性). 利用 CP-nets 图 N 来表达的 Agent 的偏好, 对于具有可交换关系的两个配置的理解很简单, 其可在 $\mathrm{O}(n)$ 时间内求出其强占优关系.

定理 7 (弱占优运算的充分性). CP-nets 图 N 上的弱占优查询可表达决策空间中的所有弱占优关系.

证明. Agent 可完成偏好的比较, 求拓扑排序, 最优配置, 求最差配置等其他运算, 且这些运算表达了决策空间中的所有弱占优关系, 因此弱占优运算是充分性. □

定理 8 (弱占优运算的有效性). 在不完全信息下, 基于弱占优查询而实现的偏好比较, 拓扑排序, 求最优配置, 求最差配置的运算能在多项式时间完成, 因此算法是有效的.

证明. 由第 3.2 节和第 3.3 节的定理 5 和定理 6 很容易得出结论. □

定理 9 (强占优关系的不完备性). 利用 CP-nets 来表达 Agent 对配置间的强占优关系一般情况下是不完备的, 即存在某个 CP-nets, 其所表达的强占优关系是不完备的.

证明. 通过构造反例来证明. 对于n个顶点的二值 CP-nets 图, 其决策空间有 2^n 个配置, 它可以表达的强占优关系图应该是 2^n 个配置组成的单向完全图, 即图中顶点数目是 $2^n \times (2^n - 1)/2$. 但从例 3 可知 $n = 3$, 此时该 CP-nets 导出图求了传递闭包后元组为 1 的数目为 23, 小于 3 个顶点应该表达的偏好个数 28, 即该 CP-nets 图可表达的强占优关系的数目比其应该表达的数目要少, 因此存在某些配置, 其强占优关系在该 CP-nets 上是无法表达的 (例如配置 $J_bP_wS_r$ 和 $J_wP_bS_r$ 是不可比的). 因此存在某个 CP-nets, 其所表达的偏好不完备. □

定理 10 (适用范围的宽泛性). 在不完全信息下, 利用 CP-nets 图 N 可实现多属性的定性偏好处理.

证明. CP-nets 图 N 中 $CPT(X_i)$ 中每一个记录是对 X_i 不同取值的一个排序, 即它表达定性关系, $CPT(X_i)$ 中的 "$x_1yz > x_2yz$" 表达了对 x_1yz 的偏好大于对 x_2yz 的偏好, 但并没有表达具体的偏好值是多少, 也没有表达 x_1yz 的偏好比 x_2yz 的偏好大多少. 结合定理 9 所阐述的某些 CP-nets 表达的偏好不完备, 则可得到 CP-nets 适合不完全信息下的多属性定性偏好处理, 且它对定性偏好的表示是宽泛的. □

事实上, 偏好包含两类, 一类是定量偏好 (Quantitative preference), 一类是定性偏好 (Qualitative preference)[3]. 其中定量偏好应用在古典的决策论和决策分析中, 且采用的概念是效用函数 (Utility function). 效用函数的本质就是货币化, 每个产品就有了一个价格, 根据价格大小就可确定对不同产品的偏好. 但是现实中还有大量的定性偏好, 人们去寻求它的效用函数时很困难或根本不可能时 (例如人们对同一牌子汽车的不同配置的偏好是不一样的, 但又很难说出两种不同配置的效用函数是多少), 我们可借助于定性偏好的表示方法来研究, 而 CP-nets 恰好就是一种定性偏好的表示语言, 因此利用 CP-nets 可实现多属性的定性偏好处理.

从定理 9 可知, 某些 CP-nets 可表达的配置间的强占优关系不完备. 而定理 10 阐述了 CP-nets 可对不完全信息下的多属性下的定性偏好进行处理. 虽然在决策空间中存在一些强占优关系在 CP-nets 中没有表达, 但借助于第 3.2 节的弱占优查询和第 4.3 节的与 Agent 之间的交互, 可间接处理 CP-nets 没有表达的一些偏好关系. 所以第 4.3 节描述的是如何提高 CP-nets 处理能力的一个方法.

4.3　与 Agent 交互实现更强的处理能力

基于 CP-nets 的偏好处理是经典的基于字符的计算, 近几年还出现了一些新型的计算—网格计算和云计算等. 与经典计算相比, 所有新型计算的本质特点是交互, 交互一方面是求解问题复杂的源泉, 另一方面又是处理复杂问题的有效手段. 计算和交互互为表里: 计算是交互的外部表现, 交互是计算的内部实现[20]. 计算和交互的统一理论是计算机科学的

基础, 以支撑计算机科学的整套理论体系.

虽然 CP-nets 可表达的强占优关系不完备, 且强占优测试是指数级复杂度, 但幸运的是 CP-nets 上的弱占优测试是充分的, 且该运算是多项式时间复杂度, 因此基于弱占优测试可实现决策空间的拓扑排序, 求极大 (小) 元[19] 等. 但最大 (小) 元却不一定能求出. 而根据定理 9 阐述的强占优关系若不完备的话, 则这些极大 (小) 元的强占优关系的判断无法从 CP-nets 中获得, 为此产生和 Agent 的交互, 将多个配置孰优孰劣的判断交由 Agent 自己决策. 此时偏好断言只是个体的事情, 与 Agent 系统无关. 下面给出提高 CP-nets 表达能力的算法 3.

算法 3. 借助交互计算实现完备的强占优判断算法

输入. CP-nets 图 $N = \langle V, CE\rangle$.

输出. N 中极大元的强占优序列.

步骤 1. 基于弱占优测试算法求出 CP-nets 图 N 的极大元集 $N_{\max} = \{o_1, o_2, \cdots, o_m\}$, 其中 m 是极大元的个数 (即 $|N_{\max}| = m$).

步骤 2. 基于强占优测试算法 Improved-Warshall 判断 $N_{\max}$ 中是否存在最大元和最小元:

1) 如果存在最大元 $o_{\max}$, 则将 $o_{\max}$ 排在最前面;

2) 如果存在最小元 $o_{\min}$, 则将 $o_{\min}$ 排在最后面.

步骤 3. 将除了最大元和最小元的极大元集 $N_{\max} - \{o_{\max}, o_{\min}\}$返回给提供 CP-nets 图 N 的 Agent 进行交互计算:

1) 对 $N_{\max} - \{o_{\max}, o_{\min}\}$ 中的配置进行拓扑排序, 得到序列 $P_{N'}$;

2) 将最大元 $o_{\max}$ 排在 $P_{N'}$ 的最前面, 最小元 $o_{\min}$ 排在 $P_{N'}$ 的最后面. 得到交互计算后的拓扑排序序列 $\langle o_{\max}, P_{N'}, o_{\min}\rangle$.

步骤 4. Agent 根据最终得到的强占优序列 $\langle o_{\max}, P_{N'}, o_{\min}\rangle$ 进行决策.

定理 11 (弱占优和交互计算结合的完备性). 基于弱占优运算和 Agent 的交互计算的算法 3, 对于极大元的强占优关系的求取是完备的.

证明. 显然基于弱占优运算可在多项式时间得到所有的极大元集合, 倘若决策空间中存在最大元 $o_{\max}$ 和最小元 $o_{\min}$, 其必定可通过强占优算法得到. 对于 $N_{\max} - \{o_{\max}, o_{\min}\}$ 中的元素来说, 他们之间的强占优关系在 CP-nets 图中没有表达, 而通过与 Agent 交互, 可以得到 $N_{\max} - \{o_{\max}, o_{\min}\}$ 配置间的强占优关系. □

5 利用 SCSP 增强部分 CP-nets 的表达能力

从第 3 节和第 4 节的描述可知, 基于 CP-nets 的偏好描述有如下缺陷: 1) 并不一定能将所有的偏好都表达出来; 2) 强占优测试的复杂度大. 其中, 缺陷 1) 可通过第 4.3 节与 Agent 交互来解决, 缺陷 2) 可通过第 5.3 节采用的归约技术来得到部分解决. 因此第 4.3 节和第 5.3 节都是间接增强 CP-nets 表达能力的一些方法.

5.1 SCSP 介绍

CP-nets 与近几年的研究热点 — 带有软约束满足问题 SCSP 具有很强的关联性, 下面给出相关的概念.

定义 12. 受限半环 (C-semiring)[4−5] $S = \langle A, +, \times, 0, 1\rangle$ 是一个包含两个二元运算 "+" 和 "×" 的代数结构, 其中

1) $\langle A, +, 0\rangle$ 是一个满足交换律和幂等律的独异点, 且 0 是 "+" 的幺元, 1 是 "+" 运算的零元.

2) $\langle A, \times, 1\rangle$ 是一个满足交换律的独异点, 1 是 "×" 运算的幺元, 0 是 "×" 运算的零元, 并且 "×" 运算对 "+" 运算满足分配律.

定义 13.

1) SCSP 是一个四元组的约束满足问题 $P = \langle C, S, V, D\rangle$, 其中 S 是定义 12 所示的受限半环, V 是决策变量的集合, D 是 V 的定义域, 即 $D = Dom(V)$. 其中 C 是所有形如 $\langle Con, Util\rangle$ 的约束集合, 即一个约束是二元组 $\langle Con, Util\rangle$, $Con \subseteq V$ 是受限的决策变量集, $Util$ 是约束 Con 的效用函数 (Utility function), 即 $Util$: $D^{|Con|} \to A$. $Util$ 代表约束 Con 所产生的效用大小, 它将一个约束的定义域 Con 的一个值映射为半环数据域 A 中的一个元素.

2) 给定一个软约束满足问题 $P = \langle C, S, V, D\rangle$, P 的一个解 Sol 是对 V 的赋值, 解的效用值是 Sol 在所有约束上的效用值的乘积, 即 $Util(Sol) = \times_{c\in C} Util(Sol \downarrow Conc)$, 其中, $Sol \downarrow Conc$ 代表解 Sol 在约束定义域 $Conc$ 上的投影[4−5], × 是半环 S 中的 × 运算.

例 6. 几个常见的软约束满足问题如下[5]:

1) 经典的约束满足问题 (Constraint satisfaction problem) 是基于半环 $SCSP = \langle\{\text{false}, \text{true}\}, \vee, \wedge, \text{false}, \text{true}\rangle$ 来求解.

2) 模糊约束满足问题 (Fuzzy constraint satisfaction problem) 是基于半环 $SFCSP = \langle [0, 1], \max, \min, 0, 1\rangle$ 来求解.

3) 带权的约束满足问题 (Weight constraint satisfaction problem) 是基于半环 $SWCSP = \langle Z^+, \max, +, 0, +\infty\rangle$ 来求解.

例 7. 给定模糊约束满足问题 $P = \langle C, S, V, D\rangle$, 其中, $S = \langle [0, 1], \max, \min, 0, 1\rangle$, $V = \{X, Y, Z\}$, $Dom(X) = Dom(Y) = Dom(Z) = \{a, b\}$, 约束集 $C = \{CXY, CYZ\}$, 其中, $CXY = \{\langle aa, 0.4\rangle$,

$\langle ab, 0.1\rangle, \langle ba, 0.3\rangle, \langle bb, 0.5\rangle\}$, $CYZ = \{\langle aa, 0.4\rangle, \langle ab, 0.3\rangle, \langle ba, 0.1\rangle, \langle bb, 0.5\rangle\}$.

解. $XYZ = bbb$ (即 $X = b$, $Y = b$, $Z = b$) 是最优解, 因此有 $bbb > abb$.

5.2 Soft Constraint 与 CP-nets 的关系

CP-nets 是一种定性偏好断言, 它的一个缺点是强占优测试为指数级复杂度. 根据 Bistarelli 等[5]的论述知, SCSP 的优点是可在多项式时间完成解的优劣判断. 可见 CP-nets 的强占优测试和 SCSP 上的解的优劣判断具有互补性: CP-nets 上的强占优测试的缺点恰好是 SCSP 上解的优劣判断的优点. 倘若能将部分 CP-nets 上的强占优测试问题在多项式时间归约为 SCSP 上的解的优劣判断问题, 则可解决部分 CP-nets 上强占优测试的指数级复杂度问题.

5.3 从 CP-nets 向 Soft Constraint 的归约

归约是一种重要的计算机技术, 它将一个不熟悉的, 难以求解的新问题, 转化为一个熟悉的, 容易求解的另一个领域的问题. 从 CP-nets 向 SCSP 归约就是在保留 CP-nets 核心特性的基础上, 将 CP-nets 上的强占优测试转化为 SCSP 上效用值的比较. 受数据库中关系分解理论的启发[21] (关系分解要保持原有的函数依赖和实现无损连接), 我们在转化过程中要遵循如下原则:

1) Ceteris Paribus 语义保持: 即在原 CP-nets 图中条件偏好表 $CPT(X_i)$ 所定义的条件偏好无关在 SCSP 中的语义要保持;

2) 强占优保持: 即在原 CP-nets 图中具有强占优关系的配置在 SCSP 中仍然具有解的优劣关系.

5.3.1 求取 SCSP 中的约束

为了将 CP-nets 下的强占优测试转变为 SCSP 中解的优劣判断问题, 首先要从 CP-nets 图 $N = \langle V, CE\rangle$ 中构造一个带有软约束的满足问题 $P = \langle C, S, V, D\rangle$. 由于 P 中的 V 就是 N 中的 V, D 是 V 的定义域, S 是一个受限半环, 本文取带权约束半环 $S = SWCSP = \langle Z^+, \max, +, 0, +\infty\rangle$, 因此转换的核心是构造 P 中的约束集 C.

由于一个约束 $c = \langle Con, Util\rangle$ 包含两部分 Con 和 $Util$ (一个约束本质上是一个函数), 其中, Con 是约束 c 的定义域, $Util$ 是约束 c 的值域, 该值域是一个从 $D^{|Con|}$ 到环中 A 上的函数. 因此构造一个约束的核心是先构造每个约束的变量集 $Con \subseteq V$, 然后设置 Con 的效用值 $Util(Con)$, 下面给出由 CP-nets 构造 SCSP 中所有约束 C 的规则.

规则 1. CP-nets 图 $N = \langle V, CE\rangle$, 若 $X_i \in V$, 则从属性 X_i 可构造一个约束 Con_{X_i}, 称 Con_{X_i} 为属性 X_i 所诱导的约束. 约束 Con_{Xi} 的定义域为 $Para(X_i) \cup \{X_i\}$, V 中所有属性所诱导的约束的定义域为 $\{Para(X_i) \cup \{X_i\} | X_i \in V\}$.

从规则 1 知, N 中每个属性和其父亲构成一个约束的定义域. 约束的定义域很直观, 因为 Agent 对属性 X_i 的各个取值的偏好取决于其父亲 $Para(X_i)$ 的不同取值, 因此 $Para(X_i) \cup \{X_i\}$ 可作为一个约束的定义域.

对于图 2 的 CP-nets 来说, N 中包含三个决策变量 J, P, S. 由于 J 和 P 没有父亲, 其对应的约束的定义域为 $\{J\}$ 和 $\{P\}$, 而 S 的父亲是 J 和 P, 其对应的约束的定义域为 $\{J, P, S\}$, $J_b > J_w$ 代表对黑颜色夹克的偏好大于对白颜色的夹克的偏好, 可理解成效用函数值与定义域 $\{J\}$ 有关; $J_bP_bS_r > J_bP_bS_w$ 代表对黑夹克、黑裤子配红衬衫的偏好大于对黑夹克、黑裤子配白衬衫的偏好, 我们可理解成偏好值与约束的定义域 $\{J, P, S\}$ 有关.

规则 2. 考虑无环 m 值 CP-nets 图 $N = \langle V, CE\rangle$, 对 N 中所有顶点进行拓扑排序后保存在单链表 L, 则:

1) $Weig(Con_{Xi}) = m^{Index(Xi)}$ 为 X_i 所诱导的约束 Con_{Xi} 的权值, 其中 $Index(X_i)$ 为属性 X_i 在 L 中的位置 (拓扑序列最后一个变量的位置为 0, 倒数第二个变量的位置为 1, $\cdots$, 即类似于十进制数 341, 1 的$Index$值为 0; 4 的 $Index$ 值为 1; 3 的 $Index$ 值为 2 (从右往左数第 0 位, 第 1 位, 第 2 位, $\cdots$).

2) 若 $CPT(X_i)$ 中一个偏好断言为 "$v_1 \wedge v_2 \wedge \cdots \wedge v_k : x_{i1} > x_{i2} > x_{i3} > \cdots > x_{im}$", 其中 "$v_1v_2\cdots v_k$" 代表 $\langle v_1, v_2, \cdots, v_k\rangle$ 是 $Para(X_i)$ 的一个取值 (X_i 有 k 个父亲), "$x_{i1} > x_{i2} > x_{i3} > \cdots > x_{im}$" 代表对 x_{i1} 的偏好最大, $\cdots$, 对 x_{im} 的偏好最小. 则约束 Con_{Xi} 的序列值 (用 $Sequ$ 来表示) 为 $Sequ(\langle v_1, v_2, \cdots, v_k, x_{i1}\rangle) = m - 1$, $Sequ(\langle v_1, v_2, \cdots, v_k, x_{i2}\rangle) = m - 2$, $\cdots$, $Sequ(\langle v_1, v_2, \cdots, v_k, x_{im}\rangle) = 0$. 可见约束 Con_{Xi} 的序列值 $Sequ$ 最大为 $m - 1$ (代表对 $Dom(X_i)$ 中的一个最大偏好的取值), $\cdots$, 最小为 0 (代表对最小偏好的取值).

3) 若 c 是约束 Con_{Xi} 定义域内的一个取值, 即 $c \in Dom(Con_{Xi})$, 且约束 Con_{Xi} 的权值为 $Weig(Con_{Xi})$, 序列值为 $Sequ(c)$, 则约束 Con_{Xi} 的效用函数值 $Util(Con_{Xi}) = \{Weig(Con_{Xi}) \times Sequ(c) | c \in Dom(Con_{Xi})\}$.

5.3.2 在 SCSP 中进行解的优劣判断

将 CP-nets 归约为 SCSP 后, 我们可将 CP-nets 上的强占优测试转化为 SCSP 上的解的优劣判

断问题. 两个 SCSP 中解的优劣判断类似于两个二进制数的大小比较.

例如要想比较 11010011 与 10011011 的大小, 首先将其转化为十进制 $11010011 = 1\times2^7+1\times2^6+0\times2^5+1\times2^4+0\times2^3+0\times2^2+1\times2^1+1\times2^0 = 211$, 而 $10011011 = 1\times2^7+0\times2^6+0\times2^5+1\times2^4+1\times2^3+0\times2^2+1\times2^1+1\times2^0 = 155$. 因为 $211 > 155$, 所以 $11010011 > 10011011$.

同样, 要想在 SCSP 中判断解 sol_1 与 sol_2 的优劣, 需要判断 $sol_1 > sol_2$ 是否成立.

规则 3. $P = \langle C, S, V, D\rangle$ 是一个基于半环的约束满足问题.

1) 当半环 $S = SWCSP = \langle Z^+, \max, +, 0, +\infty\rangle$, 若对于两个不同的解 sol_1, $sol_2 \in Z^+$, 有 $\max(sol_1, sol_2) = sol_1$, 称解 sol_1 优于解 sol_2, 即 $sol_1 > sol_2$ ("$>$" 为 max 运算诱导的偏序关系).

2) 当 $P = \langle C, S, V, D\rangle$ 是通过第 5.3.1 节的归约技术而得到的软约束满足问题, 即 $S = SWCSP = \langle Z^+, Max, +, 0, +\infty\rangle$, C、V 和 D 是通过规则 1 和规则 2 而得到的集合, 则对于 CP-nets 中的两个不同的配置 o_1 和 o_2 来说, 若 $\max(Util(o_1), Util(o_2)) = Util(o_1)$, 称配置 o_1 优于 o_2. 其中, $Util(o) = Util(o\downarrow Con_{X1}) + Util(o\downarrow Con_{X2}) + \cdots + Util(o\downarrow Con_{Xn})$. 可形象地把一个配置的效用值理解为配置中所有属性诱导的约束效用值之和 ($SWCSP$ 中的 "$+$" 运算).

例 8. CP-nets 图 N 如图 2 所示, 配置 $o_2 =$ "$J_wP_wS_r$", $o_5 =$ "$J_bP_bS_w$", $N \models o_5 > o_2$ 成立吗?

解. 图 2 的 N 的一个拓扑排序是 "J, P, S", 由规则 2 知 $Weig(Con_J) = 2^2$, $Weig(Con_P) = 2^1$, $Weig(Con_S) = 2^0$, 序列值为 $Sequ(o_5\downarrow Con_J) = 1$, $Sequ(o_5\downarrow Con_P) = 1$, $Sequ(o_5\downarrow Con_S) = 0$, 而 $Sequ(o_2\downarrow Con_J) = 0$, $Sequ(o_2\downarrow Con_P) = 0$, $Sequ(o_2\downarrow Con_S) = 1$, 故 $Util(o_5) = Util(o_5\downarrow Con_J) + Util(o_5\downarrow Con_P) + Util(o_5\downarrow Con_S) = Util(J_w) + Util(P_w) + Util(J_wP_wS_r) = 2^2\times1 + 2^1\times1 + 2^0\times0 = 6$, 而 $Util(o_2) = Util(o_2\downarrow Con_J) + Util(o_2\downarrow Con_P) + Util(o_2\downarrow Con_S) = Util(J_b) + Util(P_b) + Util(J_bP_bS_w) = 2^2\times0 + 2^1\times0 + 2^0\times1 = 1$, 由于 $Max(Util(o_5), Util(o_2)) = Max(6, 1) = 6 = Util(o_5)$, 故 $N \models o_5 > o_2$.

5.3.3 归约的特性

定义 14. 通过规则 1 和规则 2 所实现的 CP-nets 向 SCSP 的转化:

1) 若 N 中的条件偏好表中有一个条件无关条目 $yx_1z > yx_2z$, 有对应的 SCSP 中解 yx_1z 优于解 yx_2z, 则称归约具有条件无关语义保持;

2) 若 N 中配置 o_1 强占优于 o_2, 有对应 SCSP 中的解 o_1 优于 o_2, 则称归约具有强占优保持.

定理 12. 通过规则 1 和规则 2 所实现的归约满足条件无关语义保持和强占优关系保持.

证明. 1) 先证条件无关语义保持, 即证 $yx_1z > yx_2z$ 蕴含着 $Util(yx_1z) > Util(yx_2z)$. 对于顶点 X 的条件偏好表 $CPT(X)$ 来说, $yx_1z > yx_2z$ 是该表的一个条目, 其中 $y \in Dom(Para(X))$, $z \in Dom(V - \{X\} - Y)$, 即 z 是 $V - \{X\} - Y$ 中的任意一个取值. 根据规则 2 的 1), 配置 yx_1z 和 yx_2z 的权值 $Weig$ 一样, 即 $Weig(yx_1z) = Weig(yx_2z)$. 根据转换规则 2 的 2), yx_1z 的序列值大于 yx_2z 的序列值, 即 $Sequ(yx_1z) > Sequ(yx_2z)$, 因此 $Util(yx_1z) > Util(yx_2z)$.

2) 再证强占优关系保持, 即证 $o_1 > o_2$ 蕴含着 $Util(o_1) > Util(o_2)$. 设 o_1 与 o_2 第一个不同的属性为 X, 在拓扑排序得到的序列 L 中, 排在 X 之前的属性集是 V_1, 则 o_1 与 o_2 在 V_1 上的值都相同, o_1 与 o_2 在 V_1 所诱导的约束上的效用值也一样, 因此只需证明 o_1 在 $V - V_1$ 所诱导的所有约束的效用值之和大于 o_2 在 $V - V_1$ 上所诱导的所有约束的效用值之和. 由于 $o_1 > o_2$, 则 o_1 在 X 所诱导的约束上的序列值大于 o_2 在 X 所诱导的约束上的序列值, 而 o_1 与 o_2 在属性 X 所诱导的约束上的权值 $Weig$ 一样, 根据规则 2 的 2), 有 o_1 在 X 及其祖先上所诱导的约束上的效用值之和大于 o_2 在 X 及其祖先所诱导的约束上的效用值之和. 根据 $Weig$ 值的计算规则, 即使 o_2 在 $V - (V_1 \cup \{X\})$ 所诱导的约束的效用值之和大于 o_1 在 $V - (V_1 \cup \{X\})$ 所诱导的约束的效用值之和, o_2 在 $V - (V_1 \cup \{X\})$ 上所诱导的约束的效用值之和也小于 o_1 在 X 上所诱导的约束的权值 $Weig$, 因此 $Util(o_1) > Util(o_2)$. 这正如二进制 1000 与 0111 的大小比较一样, 前者高位的权值为 $1\times2^3 = 8$, 后者后三位的效用值之和为 7, 小于前者最高位的权值 8. □

定理 13. 基于无环 CP-nets 向 SCSP 归约后的强占优测试是多项式时间复杂度.

证明. 在带权约束半环中进行解的优劣判断可在多项式时间内完成, 而归约步骤中的拓扑排序也可在多项式时间完成, 加之 CP-nets 中顶点的父亲个数有限, 即求 N 中每个变量的约束也可在多项式时间内完成. 这样在 CP-nets 中进行强占优测试的时间就等于转化时间加上在 SCSP 中进行解的优劣判断时间. 而这两部分的时间都是多项式时间, 因此定理成立. □

5.4 相关工作的对比

CP-nets 已经成为一种重要的偏好表示工具, 为了理解本文的特色和贡献, 表 1 给出了本文与有关

表 1　CP-nets 的相关工作对比
Table 1　Related work comparison for CP-nets

相关论文	主要工作	有 CP-nets 表达能力分析?	有偏好完备性的探讨?	强占优全部解决? 方法复杂?	强占优实际解决?
Boutilier's[3]	论述 CP-nets 语法, 语义和应用	没有	没有	少部分解决 复杂	没有
Goldsmith's[22]	求 CP-nets 强占优测试的复杂度	没有	没有	全部解决 极其复杂	没有
Gelain's works[4]	SCSP 在缺失偏好下的处理	给出 SCSP 部分求解能力	没有	没有	没有
Bistarelli's[5]	基于半环的约束满足问题求解	没有	没有	没有	没有
本文工作	论述 CP-nets 的性质, 算法及表达能力	较全面	有	全部解决　简单	部分解决

论文工作的对比.

从表 1 可知, 本文的主要优点在于比较全面地论述了 CP-nets 的表达能力, 第一次用初等方法彻底解决了 CP-nets 的强占优测试问题.

6　结论和未来工作

本文研究了 Agent 的偏好表示工具 — CP-nets 及其表达能力, 通过对 CP-nets 导出图性质的研究, 指出二值 CP-nets 上的 $n \times 2^{n-1}$ 对可交换配置上的强占优测试可在线性时间内完成, 但对于其他配置进行强占优测试时, 其复杂度则为 $\mathrm{O}(4^n)$, 第一次以初等方法给出 CP-nets 的强占优测试的算法及其复杂度. 更为重要的是, 本文较为全面地论述了 CP-nets 的表达能力, 指出 CP-nets 适合不完全信息下的定性偏好推理.

进一步的研究方向为: 1) 研究 CP-nets 图的其他数学性质, 如可满足性、一致性等, 特别是深入研究 CP-nets 的完备性, 重点研究 CP-nets 结构与其表达的强占优关系的联系, 以进一步明确 CP-nets 表达能力严重依赖于 N 的图结构; 2) 基于 CP-nets 的表达能力, 设计 CP-nets 求解器, 从客户的偏好断言中求出其表达的含义, 从而投其所好, 为设计个性化的产品推荐系统原型奠定基础; 3) 结合文献 [23] 提出的偏好表示方法, 研究如何进一步增强 CP-nets 的表达能力, 并将 CP-nets 的表达能力与 Game Theory 的表达能力进行对比.

References

1 Wang Hong-Wei, Qi Chao, Wei Yong-Chang, Li Bin, Zhu Song. Review on data-based decision making methodologies. *Acta Automatica Sinica*, 2009, **35**(6): 820−833
(王红卫, 祁超, 魏永长, 李彬, 朱松. 基于数据的决策方法综述. 自动化学报, 2009, **35**(6): 820−833)

2 Liu Kai-Di, Pang Yan-Jun, Li Wen-Guo. Membership transforming algorithm in multi-index decision and its application. *Acta Automatica Sinica*, 2010, **35**(3): 315−319
(刘开第, 庞彦军, 栗文国. 多指标决策中隶属度转换算法及其应用. 自动化学报, 2010, **35**(3): 315−319)

3 Boutilier C, Brafman R I, Domshlak C, Hoos H H, Poole D. CP-nets: a tool for representing and reasoning with conditional ceteris paribus preference statements. *Journal of Artificial Intelligence Research*, 2004, **21**(1): 135−191

4 Gelain M, Pini M S, Rossi F, Venable K B, Walsh T. Elicitation strategies for soft constraint problems with missing preferences: properties, algorithms and experimental studies. *Artificial Intelligence*, 2010, **174**(3−4): 270−294

5 Bistarelli S, Montanari U, Rossi F. Semiring-based constraint satisfaction and optimization. *Journal of the ACM*, 1997, **44**(2): 201−231

6 Liu Shi-Xin, Guo Zhe, Tang Jia-Fu. Constraint propagation for cumulative scheduling problems with precedences. *Acta Automatica Sinica*, 2010, **36**(4): 603−609
(刘士新, 郭哲, 唐加福. 具有优先关系的累积调度问题的约束传播算法. 自动化学报, 2010, **36**(4): 603−609)

7 Koriche F, Zanuttini B. Learning conditional preference networks with queries. In: Proceedings of the 21st International Joint Conference on Artificial Intelligence. San Francisco, USA: Morgan Kaufmann Publishers, 2009. 1930−1935

8 Lang J, Mengin J. The complexity of learning separable ceteris paribus preferences. In: Proceedings of the 21st International Joint Conference on Artificial Intelligence. Francisco, USA: Morgan Kaufmann Publishers, 2009. 848−853

9 Conitzer V. Eliciting single-peaked preferences using comparison queries. *Journal of Artificial Intelligence Research*, 2009, **35**: 161−191

10 Chevaleyre Y, Koriche F, Lang J, Mengin J, Zanuttini B. *Learning Ordinal Preferences on Multiattribute Domains: the Case of CP-nets*. Berlin: Springer-Verlag, 2011. 273−296

11 Tang P Z, Lin F Z. Computer-aided proofs of arrow's and other impossibility theorems. *Artificial Intelligence*, 2010, **173**(11): 1041−1053

12 Conitzer V, Sandholm T, Lang J. When are elections with few candidates hard to manipulate? *Journal of the ACM*, 2007, **54**(3): 1−33

13 Zuckerman M, Procaccia A D, Rosenschein J S. Algorithms for the coalitional manipulation problem. *Artificial Intelligence*, 2009, **173**(2): 392−412

14 Domshlak C, Prestwich S, Rossi F, Venable K B, Walsh T. Hard and soft constraints for reasoning about qualitative conditional preferences. *Journal of Heuristics*, 2006, **12**(4−5): 263−285

15 Zhang Zhi-Zheng, Xing Han-Cheng, Wang Zhen-Zhen, Ni Qing-Jian. A preference logic based on various kinds of preferences. *Journal of Software*, 2007, **18**(11): 2728−2739
(张志政, 邢汉承, 王蓁蓁, 倪庆剑. 一种基于多类型偏好的偏好逻辑. 软件学报, 2007, **18**(11): 2728−2739)

16 Brafman R, Domshlak C, Shimony S E. On graphical modeling of preference and importance. *Journal of Artificial Intelligence Research*, 2006, **25**(1): 389−424

17 Rossi F, Venable K B, Walsh T. MCP nets: representing and reasoning with preferences of multiple agents. In: Proceedings of the 19th National Conference on Artificial Intelligence. San Jose, USA: AAAI Press, 2004. 729−734

18 Brafman R, Domshlak C. Preference handling — an introductory tutorial. *AI Magazine*, 2009, **30**(1): 58−86

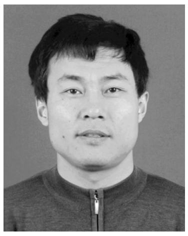
19 Rosen K H [Author], Yuan Chong-Yi, Qu Wan-Ling, Wang Han-Pin, Liu Tian [Translator]. *Discrete Mathematics and Its Applications*. Beijing: China Machine Press, 2002. 290−299
(Rosen K H [著], 袁崇义, 屈婉玲, 王捍贫, 刘田 [译]. 离散数学及其应用. 北京: 机械工业出版社, 2002. 290−299)

20 Fu Yu-Xi. Foundations, structure and problem of computer science. *Communication of CCF*, 2010, **6**(3): 44−46
(傅育熙. 计算机科学的基础、结构和问题. 中国计算机学会通讯, 2010, **6**(3): 44−46)

21 Sa Shi-Xuan, Wang Shan. *Introduction to Database Systems (Third Edition)*. Beijing: Higher Education Press, 2005. 393−396
(萨师煊, 王珊. 数据库系统概论 (第三版). 北京: 高等教育出版社, 2005. 393−396)

22 Goldsmith J, Lang J, Truszczynski M, Wilson N. The computational complexity of dominance and consistency in CP-nets. *Journal of Artificial Intelligence Research*, 2008, **33**(1): 403−432

23 Wilson N. Extending CP-nets with stronger conditional preference statements. In: Proceedings of the 19th National Conference on Artificial Intelligence. San Jose, USA: AAAI Press, 2004. 735−741



刘惊雷 烟台大学计算机学院副教授. 主要研究方向为程序理论和计算方法. E-mail: jinglei_liu@sina.com
(**LIU Jing-Lei** Associate professor at Yantai University. His research interest covers program theory and computational method.)