

信任和效用关系约束的联盟结构生成

童向荣* 任子仪

(烟台大学计算机与控制工程学院 烟台 264005)

摘要: 联盟结构生成是分布式人工智能的重要研究内容, 一般仅依据智能体效用生成任意数量的联盟, 这导致最优联盟结构生成的计算复杂度NP难。实际上, 信任是合作的基础, 信任关系对最终效用有直接的影响, 应该综合考虑信任和效用关系。针对以上问题, 该文扩展效用约束为信任和效用约束, 用信任和效用二元组表示, 以此作为联盟结构生成的依据。借鉴图割的s-t-cut算法, 研究了基于信任和效用关系的联盟结构生成, 在保证智能体个体理性和联盟稳定(无块)的前提下, 使用信任和效用关系对网络进行切割, 从而形成联盟。由此, 该文提出了两种多项式时间的精确算法: 信任关系约束下的MT-s-t-cut算法和信任效用关系约束下的MTU-s-t-cut算法, 这两种算法均能够在多项式时间内得到最优联盟结构。仿真实验验证了信任关系影响所形成的联盟结构, 社会整体效用随智能体数量的增加而增加, 并且算法的运行时间远小于动态规划法(DP)和ODP-IP算法。

关键词: 联盟结构生成; 信任效用关系; 信任传递; 图割

中图分类号: TP18

文献标识码: A

文章编号: 1009-5896(2021)07-2055-08

DOI: 10.11999/JEIT200509

Coalition Structure Generation Constrained by Trust and Utility Relationship

TONG Xiangrong REN Ziyi

(School of Computer and Control Engineering, Yantai University, Yantai 264005, China)

Abstract: The coalition structure generation is an important domain of distributed artificial intelligence. Most coalition formation models are only based on the utility, and any number of coalitions are permitted, which makes it be NP complexity difficult to generate the optimal coalition structure. Actually, Trust is the base of cooperation and has direct effect on the final utility. So, not only utility but also trust relationship should be seriously considered. To this end, the utility constraint is extended to trust and utility constraint, a two-tuples is used to represent utility and trust, which is the base of coalition structure generation. Inspired by the classic s-t-cut algorithm for graph cut, coalition structure generation constrained by trust and utility relationship is investigated. Assuming that individual rationality of agents and the stability of coalition (there is no block) is satisfied, the network is cut by the relationship of utility and trust to formation coalitions. The proposed algorithms of coalition structure generation named MT-s-t-cut and MTU-s-t-cut (Trust s-t-cut) can output the optimal coalition structure in polynomial time. The results of simulated experiments show that the social utility increases with the number of agents, and the running time of the algorithms are far less than that of Dynamic Programming (DP) and Optimal Dynamic Programming and Integer Partition (ODP-IP) algorithms.

Key words: Coalition structure generation; Trust utility relationship; Trust transitivity; Graph cut

收稿日期: 2020-06-23; 改回日期: 2020-12-26; 网络出版: 2021-02-06

*通信作者: 童向荣 xr_tong@163.com

基金项目: 国家自然科学基金(62072392, 61972360), 山东省重大科技创新工程项目(2019JZZY020131)

Foundation Items: The National Natural Science Foundation of China (62072392,61972360), Shandong Province Major Science and Technology Innovation Project (2019JZZY020131)

1 引言

5G网络切片、导频分配和云计算等资源分配问题是研究的热点, 但是该问题通常是NP难的, 近年来, 联盟和可信联盟逐渐被应用到资源分配领域, 智慧等人^[1]用联盟分裂形式的思想, 即随机划分的方法提出了一种联合用户分组和联盟博弈的动态导频分配方案, 用户被分成若干个互不相交的子

联盟。联盟形成(Coalition Formation, CF)是多智能体系统(Multi-智能体 System, MAS)的重要研究内容之一,通过对智能体集合进行划分得到联盟结构(Coalition Structure, CS),其中包含若干不相交的集合,即联盟(Coalition, C)。CF的过程通常以最大化社会福利和个体效用为目标^[2],称为联盟结构生成(Coalition Structure Generation, CSG)。合作博弈论是CF研究的基础,例如特征函数博弈^[3]、不可转移效用的博弈^[4]和享乐博弈^[5]。联盟博弈论一般假设特征函数具有超加性,即生成的主联盟是最优的。核^[6]、Shapley值^[7]为联盟博弈提供了划分收益的解决方案。最近,研究人员考虑特征函数不是超加性的情况,有的模型使用对称可加性可分离特征博弈^[8]。

CSG存在着诸多约束^[9],如通常情况下信任是合作的基础,信任是一方对另一方实现承诺的主观评估。信任程度越高越容易形成长期稳定的合作关系,而信任的传递性可以促进不熟悉的智能体之间的合作关系。信任关系对最终效用有直接的影响,因而,将信任信息融入到效用中是合理的。王海艳等人^[10]提出了基于可信联盟的服务推荐方法,将信任关系融入相似度计算;Sless等人^[11]将社交关系引入联盟形成,负值社交关系表示合作很难成功。童向荣等人^[12]和Wang等人^[13]对信任传递的特性进行了研究。Mao等人^[14]提出信任传递取最小信任值或信任值相乘的方式,信任聚合取最大信任值或对多条路径信任值进行加权平均。

近年来,研究人员在图上研究博弈,假设有向带权图的边表示智能体之间的关系。边收缩方法可以枚举智能体集的所有可行分区,Karger^[15]应用该方法解决Min-Cut问题,但未应用在联盟形成中。图割将图划分为互不相交的区域,在同一区域内的特征具有较高的相似性,而不同的区域内的特征则具有明显的差异性,其原理正是一种划分。受图割s-t-cut算法的启发,研究了基于信任和效用关系约束的CSG,在保证智能体理性和联盟稳定(无块)的情况下,使用信任和效用关系对网络进行切割,从而形成联盟。由此,提出了两种多项式时间的精确算法:信任约束下CSG和信任效用关系约束下CSG,均能够求解设定情况下的最优CS。仿真实验结果验证了所提方法的有效性。

本文主要贡献:(1)将CF的效用关系扩展为信任和效用关系,即不仅关注效用约束,还关注信任约束,并用信任和效用二元组表示,以此作为CSG的依据。(2)在保证智能体个体理性和联盟稳定(不存在块)的前提下,用分割信任网络的方法,生成

有 k 个联盟的稳定CS,提出了两种多项式时间的精确算法。

2 基本定义

传统的联盟形成只基于效用关系,没有考虑社交关系对效用的影响,近年来,学者们注意到社交关系对合作成功有必然的影响,因此,信任关系应该与效用关系一起考虑,这能提高联盟形成的效率和速度。

定义1 信任和效用关系图:智能体之间的约束关系用非对称有向加权图来表示。令 $G = \langle A, E, \rho, \omega \rangle$ 是一个有向加权图,其中, $A = \{a_1, a_2, \dots, a_n\}$ 表示有限、非空的智能体集合, $n = |A|$ 表示集合 A 中的智能体数量。 E 表示集合 A 中智能体之间的关系集,边 $\langle a_i, a_j \rangle \in E$ 有权重 ρ 和 ω ,分别表示 a_i 对 a_j 的信任关系和效用关系, $\rho_{i,j} \in [0, 1]$ 表示 a_i 对于 a_j 的信任值,当 $i = j$ 时, $\rho_{i,i} = 0$, $\omega_{i,j}$ 表示 a_i 可从 a_j 得到的效用值。如果 $\langle a_i, a_j \rangle \notin E$,则 a_i, a_j 之间的信任和效用可以通过第3部分的信任传递计算得知。

定义2 信任和效用函数: $f_{\rho, \omega}(a_i, a_j)$ 表示 a_i 和 a_j 之间信任和效用函数,能够表征智能体之间合作的可能性,该值越大,表示合作的可能性越高,反之,则表示合作的可能性越低。特别地,当 $\omega = 0$ 时,表示CSG只考虑信任关系;当 $\rho = 0$ 时,表示CSG只考虑效用关系。

令 $\mu(a_i, C)$ 为 $a_i \in C$ 从联盟 C 中得到的效用,即 $\mu(a_i, C) = \sum_{a_j \in C, j \neq i} \omega(a_i, a_j)$,简记为 μ_i 。则联盟效用可表示为 $V(C) = \sum_{a_i \in C} \mu(a_i, C)$ 。集合 A 的社会福利为 $SW(A), SW(A) = \sum_{C \in \mathcal{A}} V(C)$ 。图1给出了不同联盟结构的效用示例。

假设需要完成任务数为 $k, 0 < k \leq n$ 。 $S_k(A)$ 表示将集合 A 划分为 k 个互不相交的非空子集, $S_k(A)$ 的每种可能情况称之为联盟结构,记为CS, CS'...

例1:智能体集合 $A = \{a_1, a_2, a_3, a_4\}$,当 $k=3$ 时, $S_3(A)$ 共有6种情况: $\{\{a_1\}, \{a_2\}, \{a_3, a_4\}\}; \{\{a_1, a_2\}, \{a_3\}, \{a_4\}\}; \{\{a_1\}, \{a_3\}, \{a_2, a_4\}\}; \{\{a_2\}, \{a_3\}, \{a_1, a_4\}\}; \{\{a_1\}, \{a_4\}, \{a_2, a_3\}\}; \{\{a_2\}, \{a_4\}, \{a_1, a_3\}\}$ 。若 $CS = \{\{a_1\}, \{a_2\}, \{a_3, a_4\}\}$,则 $C_1 \rightarrow \{a_1\}, C_2 \rightarrow \{a_2\}, C_3 \rightarrow \{a_3, a_4\}$ 。

如果CS的智能体可以通过组建一个新联盟,在不降低新联盟内其它智能体收益的前提下,达到提高自身收益的目的,那么这个新联盟将破坏原有的CS,该CS是不稳定的。而这个新联盟是有更大信任和效用值的联盟,称为块。

定义3 k-联盟块:令 $CS = \{C_1, C_2, \dots, C_k\}$ 是一

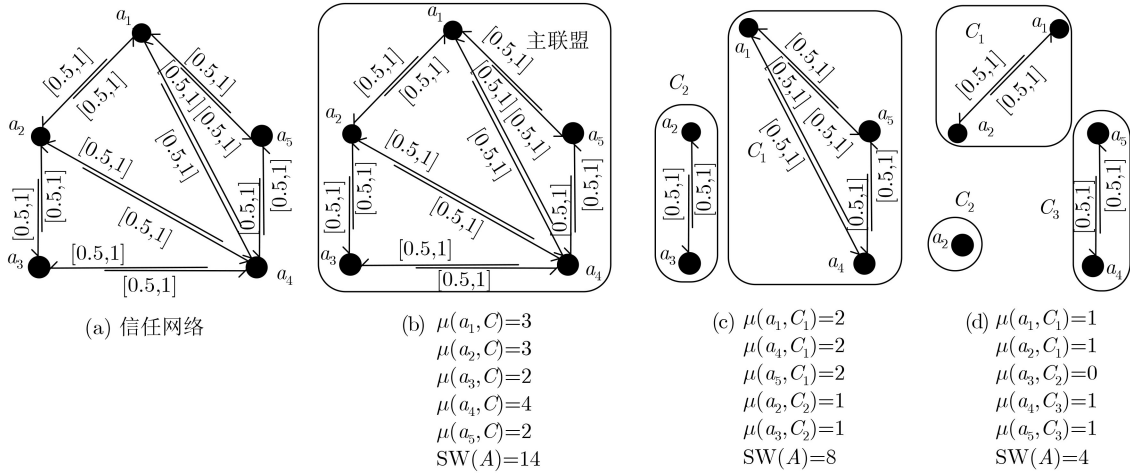


图1 不同联盟结构的效用

个联盟结构。如果在CS之外可能存在一个联盟 $B \subseteq A, B \notin CS$ (该联盟 B 内的智能体与CS中的智能体是重复的, 但联盟是不同的), 对于 $\forall a_i, a_j \in B$ 满足 $\sum f_{\rho,w}^B(a_i, a_j) > \sum f_{\rho,w}^{CS}(a_i, a_j)$, $\exists C_m \in CS$, 同时 $C_m \subseteq B$, 那么称联盟 B 为 k -联盟块, 其中 $f_{\rho,w}^B(a_i, a_j)$ 表示 a_i, a_j 在块 B 中的信任和效用值, $f_{\rho,w}^{CS}(a_i, a_j)$ 表示 a_i, a_j 在CS中的信任和效用值。

例2: 在图2中, 已知联盟结构 $CS = \{\{a_1, a_2, a_3\}, \{a_4\}, \{a_5, a_6\}\}$, 块 $B = \{a_4, a_5\}$, 存在联盟 $C_m = \{a_4\}$, 使得 $C_m \in CS$ 和 $C_m \subseteq B$ 。根据定义3, 联盟 B 是一个3-联盟块。

根据定义3易知, 如果有一个联盟是块, 那么CS中的智能体倾向于离开原联盟而形成新的联盟, 这说明块破坏了联盟结构的稳定性, 因此在满

足超加性的前提下, 没有可能的块就成为了联盟稳定的条件。若联盟的数量固定为常数 k , 则不允许出现块, 要求智能体集合恰好生成 k 个联盟, 当 $k=2$ 时, 分别记为联盟 s 和 t , s 和 t 组成新的联盟结构。

联盟结构 $CS = \{C_1, C_2, \dots, C_k\}$ 中的每个智能体所得至少与其单独所得效用是一样的, 即 $\forall a \in C_i, i \in (1, k), \mu(a, C_i) \geq \mu(a, \{a\})$, 那么CS满足个体理性。

定义4 k -联盟核: 需要同时满足以下3个条件:

- (1) 联盟结构: $CS \in S_k(A)$;
- (2) 个体理性: $\forall C \in CS, \forall a \in C, \mu(a, C) \geq \mu(a, \{a\})$;
- (3) 没有可能的块: 稳定性。

图3给出了2-联盟核的示例。在图3(b)中, $\{C_1, C_2\}$ 是2-联盟核的成员。易知, a_2 和 a_3 如果倾向于 C_2 , 其信任效用不变, 所以他们不会离开; 因为此时 a_1 的效用最大, a_1 倾向于现状; 同样道理, 在图3(c)中, $\{C_3, C_4\}$ 是2-联盟核的成员。图3中, 这两个CS的社会福利都是40, 这也是最大可能社会福利。

3 信任传递

信任网络中, 智能体之间能否形成联盟与其信

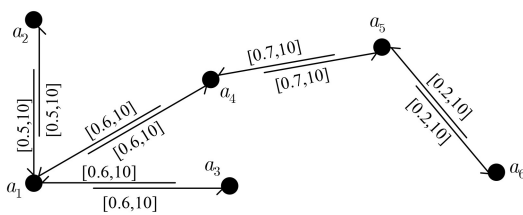


图2 智能体信任网络

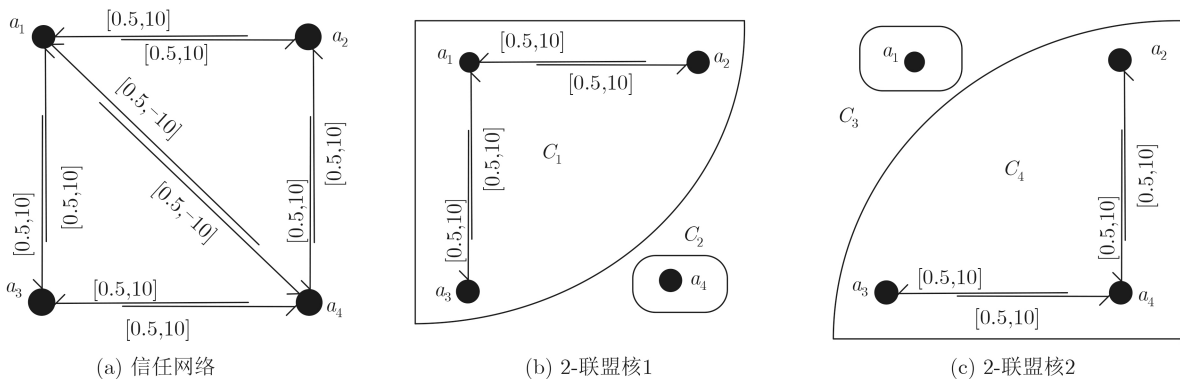


图3 2-联盟核成员

任值直接相关,信任值越大,概率越高,通过归一化处理后, a_i 与 a_j 在 t 时刻能够形成联盟的条件概率记为 $p_{ij}(t)$ 。

$$p_{ij}(t) = p\{N_{t+1} = a_j | N_t = a_i\} = \frac{\rho_{i,j}}{\sum_{u=1}^n \rho_{i,u}}, p_{ij} \geq 0, a_i, a_j \in A \quad (1)$$

例3: 图4中, 根据式(1), 得 $p_{12} = \frac{0.7}{0.7 + 0.8 + 0.6} = \frac{1}{3} = 0.333$,

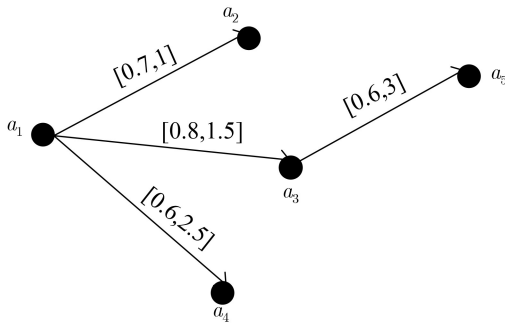


图4 信任网络

若 a_i 与 a_j 直接相连, $P_{ij}(t)$ 被称为直接信任生成概率, 即 $P_{ij}^{(1)} = \rho_{ij} \cdot P_{ij}^{(1)} = \rho_{ij}$;

若 a_i 与 a_j 不存在直接相连, a_i 经过 h 步可与 a_j 进行联系, h 步信任生成概率记为 $P_{ij}^{(h)}(t) \cdot P_{ij}^{(h)}(t)$ 。

$$P_{ij}^{(h)}(t) = p\{N_{t+h} = a_j | N_t = a_i\} = \sum_{l \in A} p_{il}^{(r)}(t) p_{lj}^{(h-r)}(t+r), a_i, a_j \in A, k \in \mathbb{N}, t \geq 0, 0 < r < h \quad (2)$$

可以证明 h 步信任生成概率, 证明: 根据边缘分布

$$\begin{aligned} P_{ij}^{(h)}(t) &= p\{N_{t+h} = a_j | N_t = a_i\} \\ &= \frac{p(N_{t+h} = j, N_t = i)}{p(N_t = i)} \\ &= \frac{\sum_{l \in A} p(N_{t+r} = l, N_{t+h} = j, N_t = i)}{p(N_t = i)} \\ &= \sum_{l \in A} \frac{p(N_{t+r} = l, N_{t+h} = j, N_t = i)}{p(N_{t+r} = l, N_t = i)} \cdot \frac{p(N_{t+r} = l, N_t = i)}{p(N_{t+r} = l, N_t = i)} \\ &= \sum_{l \in A} p(N_{t+r} = l | N_t = i) \cdot p(N_{t+h} = j | N_{t+r} = l, N_t = i) \\ &= \sum_{l \in A} p_{il}^{(r)}(t) p_{lj}^{(h-r)}(t+r) \end{aligned}$$

$P_{ij}^{(h)}(t)$ 不仅与 a_i, a_j 有关, 还与时刻 t 和步数 h 有关。若其不依赖于时刻 t , 仅与 a_i, a_j 和步数 h 有关, 简记为 $P_{ij}^{(h)}$ 。

此时,

$$P_{ij} = \max\{P_{ij}^{(h)}, h = 2, 3, 4, 5\} \quad (3)$$

h 步信任生成的效用值

$$\Omega_{ij} = \max\left\{\sum_{l \in A} (\omega_{il} p_{il})^{(r)} (\omega_{lj} p_{lj})^{(h-r)}, h = 2, 3, 4, 5\right\} \quad (4)$$

易知, 信任传递的时间复杂度是 $O(hn^2)$ 。证毕

例4: 在图4中, 求解 P_{15} 和 Ω_{15} , 根据式(3)和式(4), 得 $P_{15} = \frac{8}{21} = 0.38, \Omega_{15} = 2.16$ 。

4 MT-s-t-cut和MTU-s-t-cut算法

定义5 k-切割: $Cut_2(C_1, C_2)$ 表示将信任网络 G 根据信任值切割成互不相交的两部分, 每部分组成一个联盟, 称为2-切割。 $Cut_k(C_1, C_2, \dots, C_k)$ 表示将信任网络 G 切割成互不相交的 k 个部分, 组成 k 个联盟, 称为 k -切割。

4.1 MT-s-t-cut算法

仅考虑智能体的信任关系对CSG的影响。给定信任网络 $G = \langle A, E, \omega, \rho \rangle$, 集合 A 的基为 $|A| = n, P_{ij}, \Omega_{ij}, f_{\rho, \omega}(a_i, a_j)$ 分别为 a_i 和 a_j 之间的信任生成概率、效用和信任效用关系, $cf(p)$ 为寻找路径的残余容量(路径 p 中最小的切割对象, 此处为信任值), $V(C)$ 为联盟 C 的效用。根据信任关系和效用关系计算 P_{ij} 和 Ω_{ij} , 根据 P_{ij} 进行图切割将得到社会福利最大化的联盟结构和社会福利输出。表1为算法1的伪代码如下:

算法1的主要过程描述: 第(1)步, 算法初始化, 根据式(1)一式(4)对输入的信任网络进行计算, 得到信任生成概率 P_{ij} 与智能体之间的效用值 Ω_{ij} ; 第(2)步, 根据 P_{ij} 寻找社会福利最大化的联盟, 并输出该联盟及其效用。

计算复杂度分析: 第(1)步计算传递概率和效用, 其时间复杂度为 $O(n^2)$ 。第(2)步, 进行图割生成CS, 可以通过广度优先搜索路径, 每个路径都可以在 $O(n)$ 时间内找到。为保证找到的最终结果的切割量为最小值, 需要对智能体集遍历, 所以时间复杂度为 $O(n^3)$ 。综上, 算法1的多项式时间计算复杂度为 $O(n^2 + n^3) = O(n^3)$ 。

4.2 MTU-s-t-cut算法

除了考虑信任对于社会福利的影响, 还应该综合考虑信任和效用关系。给定信任网络 $G = \langle A, E, \omega, \rho \rangle$, $cf(p)$ 为寻找路径的残余容量(路径 p 中最

表1 算法1: MT-s-t-cut算法

输入: $G = \langle A, E, \omega, \rho \rangle$
 输出: C_1, C_2

```

// (1) 计算传递概率和效用
For all Agent  $a_i, a_j \in A$ 
     $P_{ij}, \Omega_{ij}$ 
Endfor
// (2) 进行图割生成CS
For all Agent  $a_s, a_t \in A$ 
    Do
        while exist a path p from  $a_s$  to  $a_t$  in the residual
        network Gf
            Do
                 $cf(p) \leftarrow \min \{cf(a_i, a_j) \mid (a_i, a_j) \text{ in } p\}$ 
                For each edge  $\langle a_i, a_j \rangle$  in p
                     $P_{ij} \leftarrow P_{ij} - cf(p)$ 
                     $P_{ji} \leftarrow P_{ji} + cf(p)$ 
                Endfor
            EndDo
            If  $(V(C_1) + V(C_2)) < (V(C) + V(C'))$  then
                 $C_1 \leftarrow C, C_2 \leftarrow C'$ 
            Endif
        EndDo
    Endfor
  
```

小的切割对象, 此处为信任效用关系)。根据 G 的信任关系和效用关系计算 P_{ij} 和 Ω_{ij} 。然后计算智能体之间的信任和效用关系 $f_{\rho, \omega}(a_i, a_j)$ 。循环智能体点, 根据 $f_{\rho, \omega}(a_i, a_j)$ 的值进行图割, 并输出最优社会福利的CS及其最优社会福利值。具体步骤见表2的算法2。

算法2的详细描述: 第(1)步, 算法初始化。根据式(1)–式(4)计算得到联盟形成的生成概率 P_{ij} 与智能体之间的效用值 Ω_{ij} ; 第(2)步, 将第(1)步计算结果应用于信任效用关系函数的计算, 综合考虑信任和效用关系在联盟形成过程中的影响; 第(3)步, 根据信任效用关系将智能体集划分为互不相交的两部分, 即两个联盟, 并输出最优社会福利的CS及其最优社会福利。

计算复杂度分析: 第(1)步, 计算传递概率和效用, 其时间复杂度为 $O(n^2)$ 。第(2)步, 计算信任效用关系, 其时间复杂度为 $O(n^2)$ 。第(3)步, 进行图割生成联盟, 其时间复杂度为 $O(n^3)$ 。综上, 算法2的多项式时间计算复杂度为 $O(n^2 + n^2 + n^3) = O(n^3)$ 。

Brânzei等人^[16]研究了所有联盟结构的社会福

表2 算法2: MTU-s-t-cut算法

输入: $G = \langle A, E, \omega, \rho \rangle$
 输出: C_1, C_2

```

// (1) 计算传递概率和效用
For all Agent  $a_s, a_t \in A$ 
     $P_{ij}, \Omega_{ij}$ 
Endfor
// (2) 计算信任效用关系
For each edge  $\langle a_i, a_j \rangle \notin E$ 
     $f_{\rho, \omega}(a_i, a_j)$ 
Endfor
// (3) 进行图割生成联盟
For all Agent  $a_s, a_t \in A$ 
    Do
        while exist a path p from  $a_s$  to  $a_t$  in the residual
        network Gf
            Do
                 $cf(p) \leftarrow \min \{cf(a_i, a_j) \mid (a_i, a_j) \text{ in } p\}$ 
                For each edge  $\langle a_i, a_j \rangle$  in p
                     $f_{\rho, \omega}(a_i, a_j) \leftarrow f_{\rho, \omega}(a_i, a_j) - cf(p)$ 
                     $f_{\rho, \omega}(a_j, a_i) \leftarrow f_{\rho, \omega}(a_j, a_i) + cf(p)$ 
                Endfor
            EndDo
            If  $SW(CS) < SW(CS')$  then
                 $CS \rightarrow CS'$ 
            Endif
        EndDo
    Endfor
  
```

利最大化问题, 通过最小化 k -切割规模实现。因为图中所有边的总和恒定, 通过最小化联盟之外边的总和, 可得联盟内边的权重的总和最大化, 即社会福利最大化。因此, 最小 k -切割算法可以找到社会福利最大化的联盟结构。Sless等人^[11]对其进行了改进, 只考虑联盟数量为 k 的联盟结构划分情况, 提出了一个复杂度为 $O(n^{k^2})$ 的算法, 但当 $k=2$ 时, 复杂度为 $O(n^4)$, 大于本文提出的两种算法。

5 实验

仿真实验验证了智能体数量、算法运行时间与最终得到的社会福利之间的关系, 并与几种典型算法进行了对比, 验证了所提算法的效率。

5.1 实验环境与数据

计算机的系统环境是Windows7, 64位操作系统, 8 GB内存, 3.2 GHz主频, i5-6500英特尔处理器。软件设计采用Java程序语言, Eclipse运行环境。实验数据规模即为智能体数量, 随机生成从

1个到25个智能体，实验数据满足超加性，智能体数量最大设置为25个，25个智能体的联盟结构生成，如果使用精确算法，DP和ODP-IP需要的时间是 3^{25} ，是指数级的，本文提出的算法的复杂度是 25^3 ，是多项式级的。

实验所用方法为求解s-t-cut的经典算法最大流FF(Ford-Fulkerson)算法，图的最小cut问题可以转换为最大流问题，即最小割问题和最大流问题是等价的。

5.2 对比算法

本文所提算法与以下几种算法进行对比。

(1) 动态规划(Dynamic Programming, DP): Yeh^[17]提出使用DP方法用于解决完整的集合划分问题，进而求解CSG问题，通过DP所求得的联盟结构为最优联盟结构。

(2) ODP-IP算法: Michalak等人^[18]在2015年结合任意时间算法和DP方法开发的一种称为ODP-IP的算法。

(3) MT-s-t-cut算法: 本文提出的第1个算法。

(4) MTU-s-t-cut算法: 本文提出的第2个算法。

(5) CSG-UCT算法: Wu等人^[19]在2020年基于蒙特卡罗树的搜索方法。

5.3 评价指标

将效用和时间作为评估CSG的指标。一方面研究信任对联盟效用的影响，另一方面研究智能体的基|A|对社会福利和算法运行时间的影响。

5.4 参数设置与实验结果

$0 < \rho < 0.5$ 为低信任关系， $0.5 \leq \rho < 1$ 为高信任关系，而高低信任为同时存在高、低信任关系。

5.4.1 |A|对社会福利的影响

如图5所示，横坐标表示智能体数量，纵坐标表示生成的联盟结构社会福利。

通常情况下，社会福利随着智能体数量增加而增加。分析图5易知，随着|A|的数量增多，MT-s-t-cut算法、MTU-s-t-cut算法和DP算法的社会福利均增加，并且增幅基本一致。这说明在CSG过程

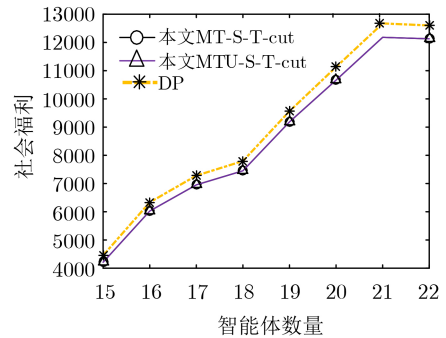


图5 |A|对社会福利的影响

中，智能体具有个体理性，即符合超加性原则。

5.4.2 |A|对运行时间的影响

通过信任网络，分别对不同数量的智能体集进行CSG，记录CSG所需的时间。图6表示两种算法的智能体数量与运行时间的关系，图7表示MT-s-t-cut算法与MTU-s-t-cut算法的运行时间对比，3个图的横坐标表示CSG的智能体数量，纵坐标为算法运行时间，单位为ms。

分析图6的易知，随着|A|的增大，两种算法的运行时间均相应增加，但不受信任关系的影响。观察图7，易知MTU-s-t-cut的算法运行时间一直比MT-s-t-cut算法的运行时间短。原因在于，使用信任效用函数后，智能体间的非对称信任效用关系被转换为对称关系，节省了最后一步生成联盟的时间。

5.4.3 运行时间对比

由图8可知，随着智能体数量的增加，DP算法和ODP-IP算法的工作量增加量要远大于MT-s-t-cut算法和MTU-s-t-cut算法。

运行时间如图9所示，横坐标表示智能体数量，纵坐标表示运行时间，单位是ms。如图9所示，所提两种算法的运行时间远小于DP算法与ODP-IP算法，并且随智能体数量的增加效果越明显。

分析原因：DP算法和ODP-IP算法的时间复杂度都为 $O(3^n)$ 。DP算法的基本思想是将待求解的问题分解成若干个子问题，先求解子问题，并保存已求解的子问题的结果，需要时直接查找。虽然可以

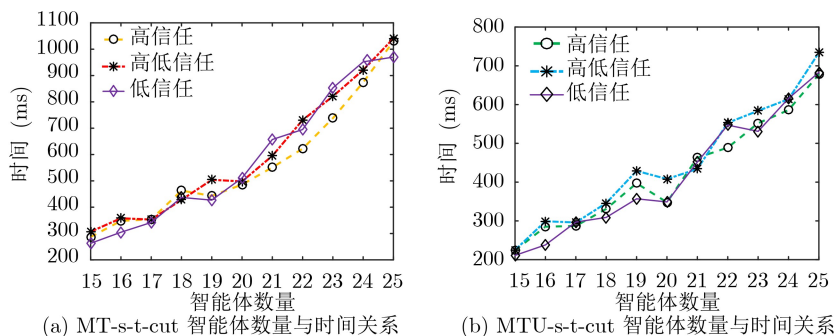


图6 两种算法的智能体数量与运行时间的关系

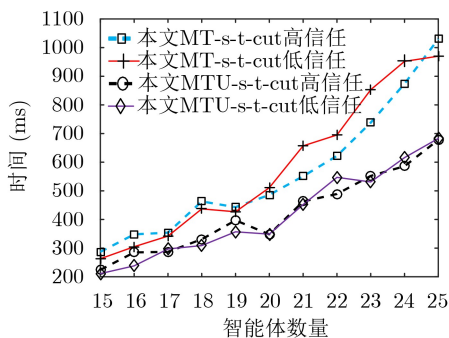


图7 算法运行时间对比

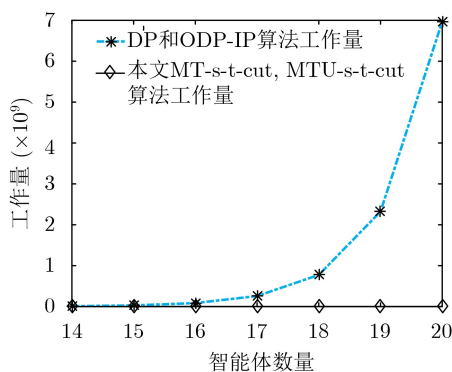


图8 工作量对比

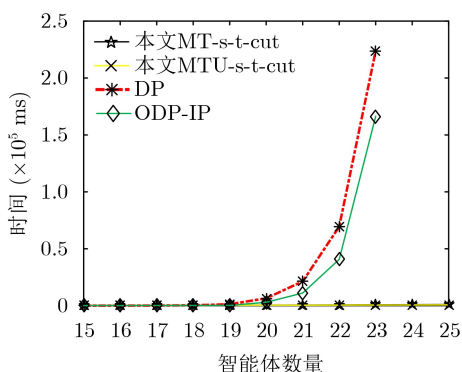


图9 前4种算法时间对比

避免大量的重复计算，但已求解的子问题无法确定在后续操作中能否被用到，造成空间的浪费。ODP-IP算法是一种混合算法，开发了一种新的搜索空间表示形式，避免了DP算法的冗余操作和其组成部分的局限性，同时保留并显著提高了每个部分的优势。因而ODP-IP算法的运行时间要小于DP算法的运行时间。MT-s-t-cut算法和MTU-s-t-cut算法的时间复杂度为 $O(n^3)$ 。基本思想是将智能体使用s-t-cut进行图分割形成联盟。易知 $O(n^3) \leq O(3^n)$ ，当且仅当 $n=3$ 时，才取等号。

若智能体集合中增加1个智能体，则DP算法和ODP-IP算法需要增加的子问题数量为指数级的，执行算法所需要的计算工作量由 3^n 变为 3^{n+1} ，工作

量增加 2×3^n 。对于MT-s-t-cut算法和MTU-s-t-cut算法，新增加的智能体与原智能体存在信任效用关系，经过信任传递，形成一个新的信任网络 G' 。 G' 与原信任网络相比，增加了 $2n$ 条边，这是因为原集合的智能体经过信任传递与新增智能体产生了信任效用关系，执行算法所需要的计算工作量由 n^3 变为 $(n+1)^3$ ，工作量增加 $3n^2 + 3n + 1$ 。当 $n=2$ 时， $2 \times 3^n < 3n^2 + 3n + 1$ ，而当 $n \geq 2$ 时， $2 \times 3^n > 3n^2 + 3n + 1$ ，且随着 n 的增大，差距越来越大。

Wu等人^[19]在2020年基于蒙特卡罗树的搜索方法对联盟结构图进行采样迭代展开搜索树，提出一种可扩展的Anytime联盟形成方法。CSG-UCT算法找到最优解的时间取决于实验的迭代次数，每次迭代都需要进行一次蒙特卡罗树搜索过程，在寻找最优解的过程中需要进行大量的迭代。当智能体数量越多时，需要的时间越多。CSG-UCT算法与MTU-s-t-cut算法时间对比如图10。随着智能体数量的增多，MTU-s-t-cut算法运行时间远少于CSG-UCT算法，智能体数量越多，运行效果越明显。

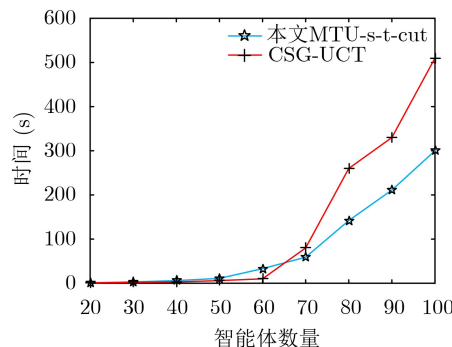


图10 与CSG-UCT算法对比

6 结束语

本文研究了基于信任和效用关系的CSG问题，给出了两个多项式时间算法计算最优联盟结构。根据信任的传递性进行信任和效用的传递；然后对传递后的信任网络进行图割，得到最优社会福利的联盟结构。最后通过仿真实验，验证了信任关系能够影响CSG的过程，并对所得联盟结构的社会福利造成影响。随智能体数量的增加，算法的运行时间相应增加，但远小于DP算法和ODP-IP算法的运行时间。联盟结构中联盟核问题及其稳定性进行分析会是未来的研究重点。

参考文献

[1] 智慧, 王飞跃, 黄子菊. 大规模MIMO系统中联合用户分组和联盟博弈的动态导频分配方案[J]. 电子与信息学报, 2020, 42(7): 1686-1693. doi: 10.11999/5EIT190445.

- ZHI Hui, WANG Feiyue, and HUANG Ziju. Dynamic pilot allocation scheme for joint user grouping and alliance game in massive MIMO systems[J]. *Journal of Electronics & Information Technology*, 2020, 42(7): 1686–1693. doi: [10.11999/5EIT190445](https://doi.org/10.11999/5EIT190445).
- [2] CHANGDER N, AKNINE S, and DUTTA A. An imperfect algorithm for coalition structure generation[C]. The Thirty-Third AAAI Conference on Artificial Intelligence, Hawaii, USA, 2019: 9923–9924. doi: [10.1609/aaai.v33i01.33019923](https://doi.org/10.1609/aaai.v33i01.33019923).
- [3] UEDA S, IWASAKI A, CONITZER V, et al. Coalition structure generation in cooperative games with compact representations[J]. *Autonomous Agents and Multi-Agent Systems*, 2018, 32(4): 503–533. doi: [10.1007/s10458-018-9386-z](https://doi.org/10.1007/s10458-018-9386-z).
- [4] GALLARDO J M, JIMÉNEZ N, and JIMÉNEZ-LOSADA A. Nontransferable utility games with fuzzy coalition restrictions[J]. *Fuzzy Sets and Systems*, 2018, 349: 42–52. doi: [10.1016/j.fss.2017.08.004](https://doi.org/10.1016/j.fss.2017.08.004).
- [5] KARAKAYA M and KLAUS B. Hedonic coalition formation games with variable populations: Core characterizations and (im)possibilities[J]. *International Journal of Game Theory*, 2017, 46(2): 435–455. doi: [10.1007/s00182-016-0533-y](https://doi.org/10.1007/s00182-016-0533-y).
- [6] İNAL H. The existence of a unique core partition in coalition formation games[J]. *Games and Economic Behavior*, 2019, 114: 215–231. doi: [10.1016/j.geb.2019.01.009](https://doi.org/10.1016/j.geb.2019.01.009).
- [7] DENG Yuan, SHEN Weiran, and TANG Pingzhong. Coalitional permutation manipulations in the gale-Shapley algorithm[C]. The 17th International Conference on Autonomous Agents and Multiagent Systems, Stockholm, Sweden, 2018: 928–936.
- [8] LIN Dianchao and JABARI S E. Transferable utility games based intersection control for connected vehicles[C]. 2019 IEEE Intelligent Transportation Systems Conference (ITSC), Auckland, New Zealand, 2019: 3496–3501.
- [9] RAHWAN T, MICHALAK T P, WOOLDRIDGE M, et al. Coalition structure generation: A survey[J]. *Artificial Intelligence*, 2015, 229: 139–174. doi: [10.1016/j.artint.2015.08.004](https://doi.org/10.1016/j.artint.2015.08.004).
- [10] 王海艳, 杨文彬, 王随昌, 等. 基于可信联盟的服务推荐方法[J]. 计算机学报, 2014, 37(2): 301–311. doi: [10.3724/SP.J.1016.2014.00301](https://doi.org/10.3724/SP.J.1016.2014.00301).
- WANG Haiyan, YANG Wenbin, WANG Suichang, et al. A service recommendation method based on trustworthy community[J]. *Chinese Journal of Computers*, 2014, 37(2): 301–311. doi: [10.3724/SP.J.1016.2014.00301](https://doi.org/10.3724/SP.J.1016.2014.00301).
- [11] SLESS L, HAZON N, KRAUS S, et al. Forming k coalitions and facilitating relationships in social networks[J]. *Artificial Intelligence*, 2018, 259: 217–245. doi: [10.1016/j.artint.2018.03.004](https://doi.org/10.1016/j.artint.2018.03.004).
- [12] 童向荣, 张伟, 龙宇. Agent主观信任的传递性[J]. 软件学报, 2012, 23(11): 2862–2870. doi: [10.3724/SP.J.1001.2012.04303](https://doi.org/10.3724/SP.J.1001.2012.04303).
- TONG Xiangrong, ZHANG Wei, and LONG Yu. Transitivity of Agent subjective trust[J]. *Journal of Software*, 2012, 23(11): 2862–2870. doi: [10.3724/SP.J.1001.2012.04303](https://doi.org/10.3724/SP.J.1001.2012.04303).
- [13] WANG Xiaofeng, SU Jinshu, WANG Baosheng, et al. Trust description and propagation system: Semantics and axiomatization[J]. *Knowledge-Based Systems*, 2015, 90: 81–91. doi: [10.1016/j.knsys.2015.09.030](https://doi.org/10.1016/j.knsys.2015.09.030).
- [14] MAO Chengying, XU Changfu, and HE Qiang. A cost-effective algorithm for inferring the trust between two individuals in social networks[J]. *Knowledge-Based Systems*, 2019, 164: 122–138. doi: [10.1016/j.knsys.2018.10.027](https://doi.org/10.1016/j.knsys.2018.10.027).
- [15] KARGER D R. Global min-cuts in RNC , and other ramifications of a simple min-cut algorithm[C]. The 4th Annual ACM-SIAM Symposium on Discrete Algorithms, Austin, USA, 1993: 21–30. doi: [10.1145/313559.313605](https://doi.org/10.1145/313559.313605).
- [16] BRÂNZEI S and LARSON K. Coalitional affinity games[C]. The 8th International Conference on Autonomous Agents and Multiagent Systems, Budapest, Hungary, 2009: 1319–1320.
- [17] YEH D Y. A dynamic programming approach to the complete set partitioning problem[J]. *Bit Numerical Mathematics*, 1986, 26(4): 467–474. doi: [10.1007/BF01935053](https://doi.org/10.1007/BF01935053).
- [18] MICHALAK T, RAHWAN T, ELKIND E, et al. A hybrid exact algorithm for complete set partitioning[J]. *Artificial Intelligence*, 2016, 230: 14–50. doi: [10.1016/j.artint.2015.09.006](https://doi.org/10.1016/j.artint.2015.09.006).
- [19] WU Feng and RAMCHURN S D. Monte-Carlo tree search for scalable coalition formation[C]. The 29th International Joint Conference on Artificial Intelligence, Yokohama, Japan, 2020: 407–413.
- 童向荣: 男, 1975年生, 博士, 教授, 主要研究方向为多Agent系统、分布式人工智能、数据挖掘。
- 任子仪: 女, 1994年生, 硕士生, 研究方向为联盟结构生成和数据挖掘。

责任编辑: 马秀强