

基于文化算法的多 agent 联盟在网络舆情监测中的应用

柳赛男^{1,2}, 陈明亮¹

(1. 浙江大学 管理学院, 杭州 310058; 2. 杭州电子科技大学 数字媒体与艺术设计学院, 杭州 310018)

摘要: 对网络舆情监测监控云平台上集聚的海量技术 agent, 如果只是逐一 agent 搜寻使联盟的利益最大化, 将耗费大量的时间和费用. 为解决该问题, 提出了基于“能力群”的动态联盟机制, 通过动态增加或删除“能力群”中的 agent 来增加联盟完成复杂任务的能力; 并提出一种基于文化算法的“能力群”agent 联盟算法, 详细讨论了算法的流程、编码、选择、交叉和变异操作的规则, 以及基于“能力群”特征的交叉算法. 仿真结果表明, 所提出的算法是可行和有效的.

关键词: 网络舆情; 多 agent 系统; 联盟; 文化算法; 编码规则

中图分类号: TP18

文献标志码: A

Application of multi-agent coalition in network public opinion monitoring based on cultural algorithm

LIU Sai-nan^{1,2}, CHEN Ming-liang¹

(1. College of Management, Zhejiang University, Hangzhou 310058, China; 2. College of Digital Media and Artistic Design, Hangzhou Dianzi University, Hangzhou 310018, China. Correspondent: CHEN Ming-liang, E-mail: Chenml@zju.edu.cn)

Abstract: It would waste a lot of time and money to search agent one by one for achieving the best coalition's behalf in a cloud platform of network public opinion monitoring. In order to solve the above problem, a dynamic coalition mechanism based on the concept of “ability group” is proposed. The proposed mechanism can add in new agent or delete old agent from “ability group” according to the performance of agent. In this way, more complex tasks can be completed by selecting agents from these “ability group” to form agent coalition. A coalition algorithm based on the culture algorithm is proposed. The calculation process and the rules of encoding, selection, crossover and mutation are discussed in detail. The simulation result shows that the proposed approach is feasible and effective.

Key words: network public opinion; multi-agent system; coalition; culture algorithm; encoding rules

0 引言

现阶段, 很多新技术都应用到网络舆情监测^[1]中, 但是在某种特定的网络舆情监测系统中所使用的技术方法都是有限的、固定的, 没有普适性和可扩展性, 同时构建某种特定的舆情监测系统需花费大量的人力和物力. multi-agent 技术作为一种分布式计算方式, 能够通过构建服务云平台^[2], 将各种新技术和方法融合在一起, 在最短的时间内对海量的信息进行过滤, 得到客户群体最关注的舆情, 为企业发现并引导舆情提供先机.

针对网络舆情监测中任务的动态性、不确定性和实时性的特点, 通过 multi-agent system (MAS) 的柔

性结构, 与 agents 之间可以组成联盟, 使得 agent 内部行为和 MAS 结构能够根据环境的动态变化迅速进行自适应调整, 更好地完成监测监控任务. 从 1993 年提出“联盟”概念以来, 联盟的形成和联盟内部利益的分配便成为 MAS 研究的一个重要方面, 并取得了一定的成果. Shehory 等^[3]提出了一种简单有效的算法来求解自治 agent 之间的任务分配问题, agent 联盟可以提高完成任务的效率. 蒋建国等^[4]提出了一种串行多任务联盟形成中的 agent 行为策略, 使用 Q-学习来求解单个 agent 的最优行为策略. 曾广周等^[5]给出了基于合同网协议和 agent 能力自信度的任务分配算法. Yang 等^[6]基于遗传算法求解 agent 联盟, 在可接受的

收稿日期: 2013-06-04; 修回日期: 2013-11-13.

基金项目: 国家自然科学基金项目(70971116); 浙江省自然科学基金项目(Y6090332).

作者简介: 柳赛男(1977—), 女, 讲师, 博士后, 从事电子商务、现代物流管理等研究; 陈明亮(1963—), 男, 教授, 博士生导师, 从事客户关系管理、政府和企业信息化规划等研究.

时间内求解的质量有所提高. An等^[7]研究了基于过渡性信任的多agent系统联盟算法, 能够为特定的目标寻找最优的agent联盟. Ye等^[8]提出了一种基于自适应的动态联盟机制, 通过谈判协议agent能够动态调整在多种联盟中的参与度, 并与其他3种agent联盟机制进行了比较, 实验结果表明此机制可以提高联盟的利益、节省计算时间.

在参考以上研究^[3-8]的基础上, 本文设计了一种基于文化算法的“能力群”agent联盟算法, 详细讨论了算法的流程和编码规则, 提出了基于“能力群”特征的交叉算法. 最后, 通过对比实验验证了算法的有效性, 为海量网络舆情监测监控技术的融合机制研究提供了一种实践途径.

1 问题描述与分析

针对网络舆情监测监控云平台上集聚的海量技术agent, 如果只是逐一agent搜寻使联盟的利益最大化, 则将耗费大量的时间. 本文提出基于“能力群”的自适应动态联盟机制, 即具有解决同类性质任务能力的agent组成一个“能力群”, “能力群”通过群内的一个或多个agent协作来完成此类性质的任务, 通过在不同的“能力群”中挑选最优的一个或几个agent组成联盟, 从而进行协作来解决复杂任务, 如图1所示. 在一段时间后, 根据“能力群”中agent执行任务的表现, 删除“能力”群中活跃度不高的agent, 加入新的agent, 从而提高MAS解决问题的能力 and 节省计算时间.

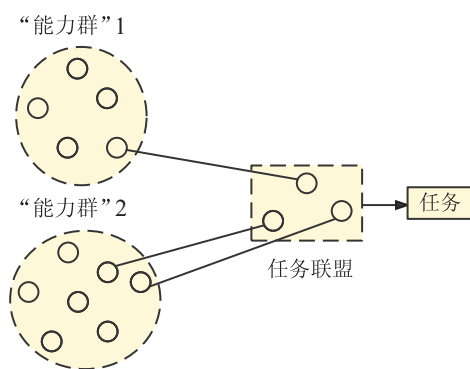


图1 “能力群”的概念

将MAS中的agents按照处理相同性质任务类别分为 n 个“能力群”, 表示为 $GA = \{ga_1, ga_2, \dots, ga_i, \dots, ga_n\}$, 其中 $ga_i = \{A_1^i, A_2^i, \dots, A_m^i\}$, $m \in N$, 即每一个“能力群”由 m 个agent组成, 每个 ga_i 的agent的个数不一定相同, m 为自然数. 任意 A_j^i 都有一个能力向量, 表示为 $B_j^i = \langle b_{j1}^i, b_{j2}^i, \dots, b_{jr}^i \rangle$, $b_{jk}^i > 0$, $1 \leq i \leq n$, $1 \leq j \leq m$, $1 \leq k \leq r$, 用于定量描述 A_j^i 执行某种特定任务的能力大小. 任务 $T = \{t_1, t_2, \dots, t_l, \dots, t_k\}$, 每个任务 t_l 具有一定的能力需求 $R_l = \langle r_l^1, r_l^2, \dots, r_l^r \rangle$. 从MAS的“能力群”中选择一组agents结

成联盟 C 来完成某个特定任务, 同时假设每一个联盟 C 只能完成一个任务, 每个agent只能参加一个联盟 C , 此联盟 C 的能力向量为 $B_c = (b_1^c, b_2^c, \dots, b_r^c)$, B_c 是联盟 C 中所有agent能力向量的总和, 即

$$B_c = \sum_{A_j^i \in C} B_j^i. \quad (1)$$

联盟 C 能够完成任务 t_l 需要满足的必要条件是: 存在

$$1 \leq i \leq r, r_i^l \leq b_r^c; \quad (2)$$

同时, 定义目标函数完成所有任务时系统的总收益是最大的, 即

$$\max \left(\sum_{l=1}^k V(C_{t_l}) \right) = P(C_{t_l}) - \alpha F(B_c) - \beta L(C_{t_l}) - \gamma C(C_{t_l}). \quad (3)$$

其中: $P(C_{t_l})$ 为完成任务 t_l 时联盟 C_{t_l} 所获得的收益, $F(B_c)$ 为联盟 C_{t_l} 完成任务 t_l 所付出的能力成本, $L(C_{t_l}) = B_c - R_l$ 是当联盟 C_{t_l} 能够完成任务 t_l 时, 所付出的能力与任务要求的能力之间的差值, 为能力余量, 表明当为完成某任务使用能力较强的联盟时, 会丧失此联盟解决更复杂任务的能力, 从而影响所有任务完成时系统的总收益, $C(C_{t_l}) = pC_n^2$ 为组成联盟的各agent之间的通信费用. 不考虑agent之间距离的影响, 假设各agent之间的通信单价为 p , 当联盟中的agent越多时, 通信费用越高, C_n^2 为组合概率, 是当联盟中有 n 个agent时产生的通讯关系总数. 当联盟 C_{t_l} 能够完成任务 t_l 时, $V(C_{t_l})$ 为正值; 当不能完成时, $V(C_{t_l})$ 为0. $\alpha, \beta, \gamma \in [0, 1]$, 且 $\alpha + \beta + \gamma = 1$.

2 算法设计

针对特定任务的基于“能力群”agent动态联盟算法的设计步骤如下:

- 1) 首先按照每个“能力群”解决某种性质问题的能力, 计算群中每个agent的能力向量 b_{jk}^i , 并按照从大到小的顺序排列;
- 2) 按照任务的优先等级对任务进行排列后, 优先级高的任务排在前面;
- 3) 按照执行任务所要达到的各向向量由大到小的顺序, 从不同的“能力群”选择相应的agent组成任务联盟;
- 4) 判断任务联盟的各向能力向量是否满足任务各向能力的要求, 如果满足, 则计算解决下一个任务的agent联盟, 如果不满足, 则转到3);
- 5) 计算所有任务完成后, 判断系统的收益是否最大, 如果是最大, 则转到9), 同时, 将各参与任务联盟的agent的活跃度增加1次, 如果不是, 则转到6);
- 6) 计算各“能力群”中各agent的活跃度, 即在一段时间内, 累计参加任务联盟的次数;

7) 判断各 agent 是否满足继续留在“能力群”中的条件, 如果满足, 则将此 agent 继续留在“能力群”中进行服务, 接着执行 3), 如果不满足, 则执行 8);

8) 删除此 agent, 同时可添加其他待加入的 agent 到“能力群”中, 接着执行 3);

9) 计算结束.

当面对复杂任务时, 根据需要完成的任务的优先级顺序, 计算各任务所需要达到的各向向量的大小, 并按照从大到小的顺序进行排列. 此时, 从对应“能力群”中选择对应能力向量最大的 agent 来满足任务需要向量的要求. 这样, 使搜索 agent 形成联盟完成任务的时间大大缩短, 不再需要一一搜索, 而是搜索对应任务“能力群”中最大能够完成任务的能力向量即可. 采用这种方式能使搜索问题的复杂度大大降低, 更加灵活有效, 减少了联盟生成的搜索时间和计算量, 可实现性较好.

3 基于文化算法的优化方法

作为一种新型的进化算法, 文化算法能够精确地反映进化过程, 并提高计算效率^[9-10]. 文化算法一般由种群空间和信仰空间两部分组成. 文化算法的结构如图 2 所示, 种群空间是基于传统种群的进化, 信仰空间是基于信念文化的进化, 是知识经验的形成、储存和传播, 两者相对独立但又相互联系. 本文提出了一种基于文化算法的遗传算法来解决 agent 联盟的求解问题.

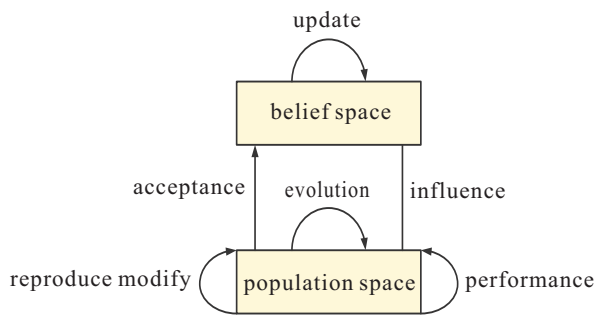


图 2 文化算法框架

3.1 优化算法设计

3.1.1 编码规则

对于每个“能力群”中 agent, 都根据处理同类性质任务的能力大小按照从大到小的顺序进行排列, 即 A_j^i 表示来自第 i 个“能力群”排在第 j 位的 agent, 采用的编码方式如表 1 所示. 将任务按照优先等级的顺序进行排列, 假设共有 k 个任务, 排列后表示为 $1, 2, \dots, i, \dots, k$. 根据前面的假设, 一个联盟只能完成一个任务, 每个 agent 只能参加一个联盟. 本文提出一种二维编码方法, 如表 2 所示.

表 1 agent 编码方法

染色体	1	2	1	2	3	2	3	1	...
A_j^i	A_1^1	A_1^2	A_2^1	A_2^2	A_3^1	A_3^2	A_3^3	A_3^1	...

表 2 染色体的编码方法

A_j^i	任务					
	1	2	...	i	...	k
1	0	0	...	1	...	0
...
2	0	1	...	0	...	0
...
$(n-1)$	1	0	...	0	...	0
...
n	0	1	...	0	...	0
...

3.1.2 接受函数

在进化的早期阶段, 信仰空间为了使个体优化, 防止早熟, 不接受太多的种群个体. 在进化的晚期阶段, 信仰空间为了避免相似信息的冗余, 也不接受太多的种群个体. 在这两个阶段的中间, 群体有价值的信息不断增加, 信仰空间可以接受更多的种群个体. 本文定义接受函数

$$Anum = -(\text{Currentgeneration} - A)^2 + B. \quad (4)$$

其中 A, B 为大小合适的常数, 例如, 当进化代数 300, 种群规模为 100 时, A 可以为 150, B 可以为 60.

3.1.3 影响函数

信仰空间通过如下影响函数来影响种群:

$$\begin{cases} \text{individual from belief space, rand} < C; \\ \text{original individual, otherwise.} \end{cases} \quad (5)$$

其中: C 为一个大小合适的常数, rand 为服从均匀分布的随机函数. 当 rand 函数产生的随机数小于 C 时, 由信仰空间中的一个个体替代种群空间中的个体, 反之, 则不替代.

3.1.4 信仰空间更新

信仰空间根据下式进行更新:

$$i^{t+1} = \begin{cases} x_{\text{best}}, f(x_{\text{best}}) < f(i^t), \\ i^t, \text{ otherwise.} \end{cases} \quad (6)$$

在更新代数 t 时, 如果接受函数中某个个体的适应度函数值小于信仰空间中某个个体的适应度函数值, 则对信仰空间进行更新, 替代信仰空间中的个体, 从而形成更新代数 $t + 1$.

3.1.5 选择算子

首先确定适应度函数即目标函数

$$f = \max \left(\sum_{l=1}^k V(C_{tl}) \right),$$

计算 $p = f_i / \sum f_i$, 然后使用轮盘赌法对种群内的染色体进行个体的选择.

3.1.6 交叉算子

为了产生更多的新个体, 增加空间的搜索能力, 同时保证交叉后的解集可行, 本文采用基于可变“能力群”特征的交叉操作, 即保留父代个体某“能力群”中组成联盟完成任务的agent的位置和个数, 改变来自其他“能力群”中组成联盟的agent. 用 $\{1, 2, \dots, m\}$ 对“能力群” ga_i 进行编号, 来自不同“能力群”的agent的编码方式如表1所示. 在交叉操作中, 随机选择一个“能力群”编号 $i, i = 1, 2, \dots, m$, 交换两个父代染色体中“能力群”中agent参与组成联盟的位置和个数而形成两个子代, 其交换机制如图3所示.

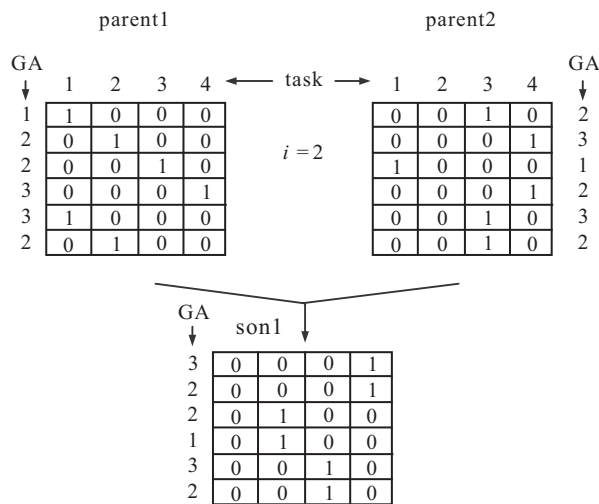


图3 交叉操作

3.1.7 变异算子

在变异操作中, 任意交换染色体中两个GA群的位置即可完成, 如图4所示.

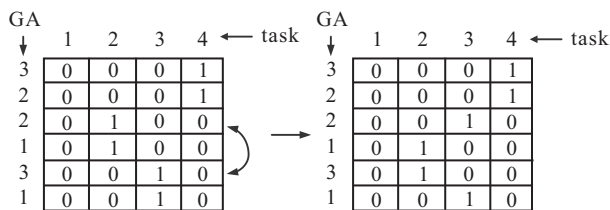


图4 变异操作

3.2 优化算法描述

本文所提出的基于文化算法的遗传算法的计算步骤如下:

1) 初始化种群空间, 种群空间的大小为 N_{is} , 每个种群有 N_i 个染色体. 每个染色体由前述的编码方式进行编码; 然后初始化信仰空间, 在最初阶段, 信仰空间的大小为 $k = N_{is}$, 随着信仰空间的更新, k 由 k_L 变到 k_U . k_L 为下限, k_U 为上限.

2) 计算每个染色体的适应度值, 然后判断是否从种群空间中接受个体到信仰空间中. 一旦执行接受函数, 信仰空间中表现最差的个体将被种群空间中表现最好的个体所代替. 如果不执行接受函数, 则需要判断是否满足选择、交叉和变异操作的条件.

3) 如果满足选择、交叉和变异操作的条件, 则执行选择、交叉和变异操作; 同时, 判断是否执行影响函数. 如果执行影响函数, 则种群空间中表现最差的个体将被信仰空间中表现最好的个体所代替; 如果不执行影响函数, 则转到5). 如果不满足选择、交叉和变异操作的条件, 也转到5).

4) 计算每个染色体的适应度值.

5) 判断是否满足终止条件, 如果满足, 则进化结束; 如果不满足, 则回到2).

4 仿真实验

为了验证本文算法的有效性和正确性, 本文设计3组实验, 假定交叉概率 $P_c = 0.6$, 变异概率 $P_m = 0.01$, 每组实验的重要参数见表3, 其他参数由于篇幅的原因, 在此省略. 实验结果如表4所示.

表3 实验参数设计

实验编号	GA 编号	agent 数量	任务数	种群规模	迭代次数	任务能力	完成任务收益
1	1	3	4	20	100	$B_{T1} = \langle 3, 5, 8 \rangle$	$P(T_1) = 32$
	2	3					
	3	4					
2	1	4	4	30	500	$B_{T2} = \langle 6, 10, 3 \rangle$	$P(T_2) = 38$
	2	5					
	3	6					
3	1	5	4	30	1000	$B_{T3} = \langle 2, 4, 5 \rangle$	$P(T_3) = 22$
	2	7					
	3	8					

表4 计算结果

实验编号	最佳收益值	平均最佳收益值	平均最差收益值	运行时间	文献[6]的结果	文献[6]的时间
1	97	95	80	0.06	95	0.06
2	108	103	99	1.32	103	1.57
3	116	112	103	2.18	111	2.32

表4的计算结果表明本文给出的计算方法可以很快地收敛到最优解, 可见“能力群”中可供选择的agent的数量越多, 得到的最优值越大. 与文献[6]中的算法相比, 本文方法的解更优、收敛速度更快. 同时与文献[6]中染色体的操作相比, 本文方法操作上更方便, 而且能提高解的质量.

5 结论

本文提出基于“能力群”的自适应动态联盟机制, 并设计了一种基于文化算法的“能力群”agent联盟算法, 详细讨论了算法的流程、编码、选择、交叉和变异操作的规则, 提出了基于“能力群”特征的交叉算法.

仿真结果表明,该算法是可行的,与文献[6]中的方法相比,收敛效果更好,计算效率较高,获得的结果较为理想,可以为agent联盟最大化系统总收益的计算提供参考.对此研究还可以更深入,如考虑随机任务情况下的agent动态联盟系统收益问题,MAS体系结构随环境动态变化时联盟如何进行快速调整的问题等.

参考文献(References)

- [1] 朱恒民,苏新宁,张相斌.互联网舆情演化的动态网络模型研究[J].情报理论与实践,2010,33(10):75-78.
(Zhu H M, Su X N, Zhang X B. Research on dynamic network model of online public opinion evolution[J]. Information Studies: Theory & Application, 2010, 33(10): 75-78.)
- [2] Erdil D C. Autonomic cloud resource sharing for intercloud federations[J]. Future Generation Computer Systems, 2012, 29(7): 1-9.
- [3] Shehory O, Kraus S. Task allocation via coalition formation among autonomous agents[C]. Proc of IJCAI-95. Los Angeles: Morgan Kaufmann Publishers, 1995: 655-661.
- [4] 蒋建国,苏兆品,张国富,等.多任务联盟形成中的Agent行为策略研究[J].控制理论与应用,2008,25(5):853-856.
(Jiang J G, Su Z P, Zhang G F, et al. Agent-behavior strategy in serial multi-task coalition formation[J]. Control Theory & Applications, 2008, 25(5): 853-856.)
- [5] 曾广周,杨公平,王晓林.基于Agent能力自信度的任务分配问题研究[J].计算机学报,2007,30(11):1922-1929.
(Zeng G Z, Yang G P, Wang X L. Study of task allocation problem based on agent ability confidence[J]. Chinese J of Computers, 2007, 30(11): 1922-1929.)
- [6] Yang J G, Luo Z G. Coalition formation mechanism in multi-agent systems based on genetic algorithms[J]. Applied Soft Computing, 2007, 7(2): 561-568.
- [7] Bo An, Zhiqi Shen, Chunyan Miao, et al. Algorithms for transitive dependence-based coalition formation[J]. IEEE Trans on Industrial Informatics, 2007, 3(3): 234-245.
- [8] Dayong Ye, Minjie Zhang, Danny Sutanto. Self-adaptation-based dynamic coalition formation in a distributed agent network: A mechanism and a brief survey[J]. IEEE Trans on Parallel and Distributed Systems, 2013, 24(5): 1042-1051.
- [9] Amira Haikal, Mostafa El-Hosseni. Modified cultural-based genetic algorithm for optimization[J]. J of Ain Shams Engineering, 2011, 2(3/4): 173-182.
- [10] 刘纯青,杨莘元.文化算法及其应用研究[D].哈尔滨:哈尔滨工程大学信息与通信工程学院,2007.
(Liu C Q, Yang X Y. Research on cultural algorithms and their applications[D]. Harbin: College of Information and Communication Engineering, Harbin Engineering University, 2007.)

(责任编辑:孙艺红)