# Multi-agent coalition in network public opinion monitoring based on cloud cultural algorithm

Sainan Liu[1, a]

[1]College of digital media and artistic design, Hangzhou Dianzi University, Hangzhou, 310018, China

[a]Liusn@hdu.edu.cn

**Keywords:** multi-agent system; coalition; cloud culture algorithm

**Abstract.** In the new media environment, the marketing strategy can be made better by constructing a cloud platform of network public opinion monitoring that has huge amounts of technologies. It would be cost a lot of time and money to search agent one by one for achieving the best coalition's behalf in such a platform. In order to solve the above problem, the concept of "ability group" was proposed in this paper. The agents that have the similar ability can be grouped together. The agents in the same "ability group" can cooperate with each other. The complex task can be completed by selecting agents from these "ability group" to form agent coalition. A coalition algorithm based on cloud culture algorithm was proposed. The rules of encoding, selection, crossover and mutation were discussed in detail. The simulation result shows that the approach provides a good way for studying the fusion mechanism of network public opinion monitoring technologies.

## Introduction

In this paper, we proposed the concept of "ability group" that is composed of agents that have the similar ability. One or several agents from one "ability group" cooperate with each other to complete the same kind task. The inert agent is deleted from "ability group" and new agent is added in "ability group". In this way, it's better for improving the performance of MAS and saving calculation time. We propose an agent "ability group" coalition calculation based on culture algorithm and cloud model, including the description of calculation and the coding rules. Finally, the simulation result shows that it's effective and can provide a method for studying fusion mechanism in network public opinion monitoring.

## Literature Review

The existing papers that study multi-agent coalition are mainly classified into two main streams. One is to maximize the benefit of agent coalition, coalition is arranged among the all of the agents before task can be completed. Cheng Bailiang, et al [1] study the trust coalition based on self-organization evolution. In order to get stable coalition, the free competition and trust evaluation is used for distributing income among coalition. Xiaohan Yu, et al[2] discussed the multi-agent coalitional decision making problems with two camps in detail in order to select the best coalitional strategy for the concerned camp to maximize the damage of the other camp. They devised two integer programming models to select the best coalitional strategy. Dayong Ye, et al [3] proposed a self-adaptation-based dynamic coalition formation mechanism. The agents can adjust their degrees of involvement in multiple coalitions by exploiting a negotiation protocol. The proposed mechanism is evaluated through a comparison with other three coalition formation mechanisms. Experimental results demonstrated that it can improve the coalition benefit and save calculation time.

The other is to study the algorithm to improve the solution quality and speed up the calculation. Xubo, et al [4] proposed a multi-objective quantum evolutionary algorithm to solve multi-agent coalition formation problem. The coding mapping, the combination of resources and task allocation are combined to a process, which reduced the complexity of the problem. Su Shexiong, et al [5] proposed OCS algorithm based on local optimum. Based on local optimum, the graph of agent coalition structure can be shirt cut and the graph of agent coalition structure is pruned by using the upper bound of coalition structures referred to the partition, which decreases the searching space.

Jiangan Yang, et al [6] proposed a GA-based algorithm for coalition structure formation which aims at achieving goals of high performance, scalability, and fast convergence rate simultaneously. The algorithms improved the efficiency and shorten the running time greatly.

## Problem description and analysis

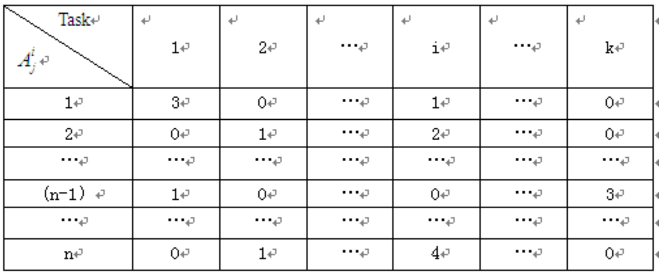
The agents are classified into different "ability group" according their task processing ability. $GA=\{ga_1,ga_2,\cdots,ga_i,\cdots ga_n\}$, $ga_i=\{A_1^i,A_2^i,\cdots,A_m^i\}$, $m\in N$ denotes that each "ability group" is composed of number $m$ agent. The number of agent $A_m^i$ in each $ga_i$ is different, $m$ is a natural number. The ability vector of each agent is denoted by $B_j^i=\langle b_{j1}^i,b_{j2}^i,\cdots,b_{jr}^i\rangle,b_{jk}^i>0(1\le i\le n,1\le j\le m,1\le k\le r)$ which is used to describe the agent's ability of processing one task. Task is denoted by $T=\{t_1,t_2,\cdots,t_l,\cdots t_k\}$, each subtask $t_l$ has a certain ability requirement which is denoted by $R_l=\langle r_1^l,r_2^l,\cdots,r_r^l\rangle$. For one task $T$, the agent coalition C is formed by selecting agents from different "ability group". We suppose that one coalition C only can complete one task T and one agent only can join in one agent coalition C. the ability vector of one coalition C is denoted by $B_c=\langle b_1^c,b_2^c,\cdots,b_r^c\rangle$. The total ability of all the agents in the coalition C is denoted by $B_c$. The requirement of coalition C completing subtask $t_l$ must be

$$\forall 1\le i\le r,r_i^l\le b_r^c \qquad (1)$$

The objective of our problem is to maximize the coalition benefit after completing the task, that is:

$$\max mize(\sum_{l=1}^{k}(V(C_{t_l})=P(C_{t_l})-\alpha F(B_c)-\beta L(C_{t_l})-\gamma C(C_{t_l}))) \quad (2)$$

$V(\mathrm{C}_{t_l})$ denotes the benefit of the coalition $C_{t_l}$. $V(\mathrm{C}_{t_l})$ is positive when the coalition $C_{t_l}$ can complete the subtask $t_l$. If not, $V(\mathrm{C}_{t_l})$ is zero. $P(C_{t_l})$ is the reward of coalition $C_{t_l}$ for completing subtask $t_l$. $F(B_c)$ is the ability cost of coalition $C_{t_l}$ to complete the subtask $t_l$. $L(C_{t_l})=B_c-R_l$ is the ability allowance that is the difference between the ability of the agents and the required ability of the task after coalition $C_{t_l}$ completing the subtask $t_l$. If the coalition which has stronger ability is used for solving simple task, it will lose its chance to complete more complex task. Then the benefit of the whole system will be reduced. $C(C_{t_l})=pC_n^2$ is the communication cost between different agents in one coalition. The distance of agents is not considered in this paper. The communication price is supposed to be $p$. The communication cost is higher when the coalition has more agents. $C_n^2$ is a combined probability which denotes the total number of communication between agents in the coalition. $\alpha,\beta,\gamma\in[0,1]$ and $\alpha+\beta+\gamma=1$.



| Task<br>$A_j^i$ | 1 | 2 | … | i | … | k |
|---|---|---|---|---|---|---|
| 1 | 3 | 0 | … | 1 | … | 0 |
| 2 | 0 | 1 | … | 2 | … | 0 |
| … | … | … | … | … | … | … |
| (n-1) | 1 | 0 | … | 0 | … | 3 |
| … | … | … | … | … | … | … |
| n | 0 | 1 | … | 4 | … | 0 |

Fig. 1 The encoding method of two-dimensional binary chromosome

## Optimization

### Coding Rules.

An encoding method of two-dimensional binary chromosome is proposed based on "ability group", as shown in figure 1. In our encoding method, the agents in one "ability group" are denoted by 1,2,3,... according their ability size from large to small. For example, 2{1,2,3,4,5} denoted $A_1^2, A_2^2, A_3^2, A_4^2, A_5^2$.

### Acceptance Function.

The acceptance function is defined in formula (4). $T$ is the maximum generation. $t$ is the current generation. $C_1$ and $C_2$ are the control coefficient. $p$ is the population. $k$ is a decimal number and $k \in [0,1]$.

$$accept() = [p \cdot k \cdot e^{-\frac{(t-E_x)^2)}{2E_n'^2}}] \qquad (4)$$

Subject to:

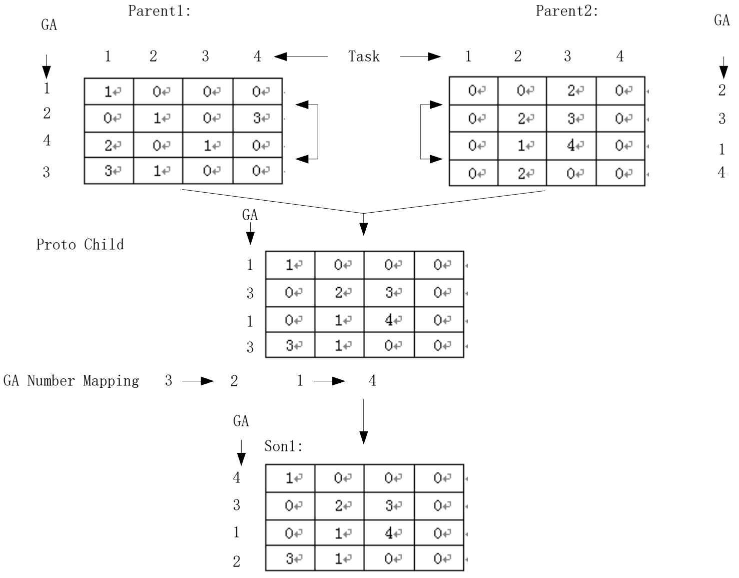
$$E_x = T/2 \ , \ E_n = (t - T/2)/C_1 \ , \ H_e = (E_n)/C_2 \ , \ E_n' = Norm(E_n, H_e) \qquad (5)$$

### Selection

The affinity function is $f(X) = 1/J(X)$. $J(X)$ is the objective function. The individual is selected by roulette wheel method.

### Crossover.

To generate more new individuals and make the solution more feasible, a crossover operation based on "ability group" is proposed, as shown in figure 2.



Parent1:

| GA | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| 1 | 1 | 0 | 0 | 0 |
| 2 | 0 | 1 | 0 | 3 |
| 4 | 2 | 0 | 1 | 0 |
| 3 | 3 | 1 | 0 | 0 |

Parent2:

| 1 | 2 | 3 | 4 | GA |
|---|---|---|---|---|
| 0 | 0 | 2 | 0 | 2 |
| 0 | 2 | 3 | 0 | 3 |
| 0 | 1 | 4 | 0 | 1 |
| 0 | 2 | 0 | 0 | 4 |

Proto Child

| GA | | | | |
|---|---|---|---|---|
| 1 | 1 | 0 | 0 | 0 |
| 3 | 0 | 2 | 3 | 0 |
| 1 | 0 | 1 | 4 | 0 |
| 3 | 3 | 1 | 0 | 0 |

Son1:

| GA | | | | |
|---|---|---|---|---|
| 4 | 1 | 0 | 0 | 0 |
| 3 | 0 | 2 | 3 | 0 |
| 1 | 0 | 1 | 4 | 0 |
| 2 | 3 | 1 | 0 | 0 |

Fig. 2 Crossover operation

### Mutation

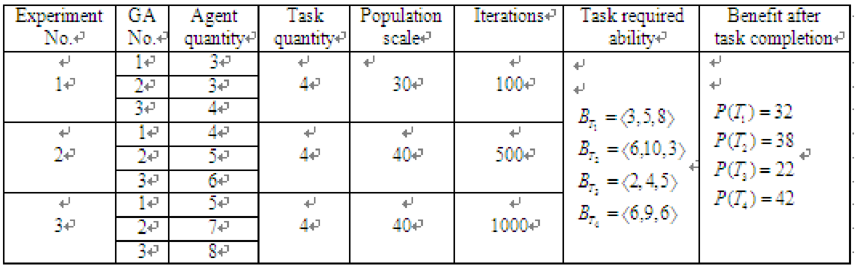
The mutation is to select randomly two rows of the chromosome and exchange their location.

## Simulation and analysis

In order to verify our algorithm, we design three experiments. Supposed that crossover probability is $P_c = 0.6$, mutation probability is $P_m = 0.01$. The important parameter of each experiment is shown in Table 1. The result of these experiments is shown in Table 2.

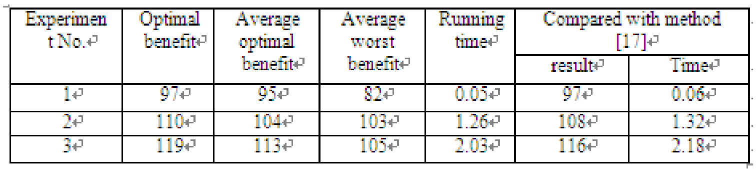
The calculation result of these experiments is shown in Table 2. The result shows that our proposed method can get the optimal solution more quickly. From the experiment, we can see that the optimal value is bigger when the number of agents in the "ability group" is more enough to select. Compared with method[7], the solution is more optimal in our proposed method in this paper. And the acceptance function is more objective and scientific than the method [7].

Table 1 Experiment Parameter Design



| Experiment No. | GA No. | Agent quantity | Task quantity | Population scale | Iterations | Task required ability | Benefit after task completion |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 3 | 4 | 30 | 100 | $B_{T_1}=\langle 3,5,8\rangle$<br>$B_{T_2}=\langle 6,10,3\rangle$<br>$B_{T_3}=\langle 2,4,5\rangle$<br>$B_{T_4}=\langle 6,9,6\rangle$ | $P(T_1)=32$<br>$P(T_2)=38$<br>$P(T_3)=22$<br>$P(T_4)=42$ |
|  | 2 | 3 |  |  |  |  |  |
|  | 3 | 4 |  |  |  |  |  |
| 2 | 1 | 4 | 4 | 40 | 500 |  |  |
|  | 2 | 5 |  |  |  |  |  |
|  | 3 | 6 |  |  |  |  |  |
| 3 | 1 | 5 | 4 | 40 | 1000 |  |  |
|  | 2 | 7 |  |  |  |  |  |
|  | 3 | 8 |  |  |  |  |  |

Table 2. Calculation result



| Experiment No. | Optimal benefit | Average optimal benefit | Average worst benefit | Running time | Compared with method [17] |  |
|---|---|---|---|---|---|---|
|  |  |  |  |  | result | Time |
| 1 | 97 | 95 | 82 | 0.05 | 97 | 0.06 |
| 2 | 110 | 104 | 103 | 1.26 | 108 | 1.32 |
| 3 | 119 | 113 | 105 | 2.03 | 116 | 2.18 |

## Conclusions

In this paper, we propose the concept of "ability group". An agent coalition based on "ability group" is proposed and an algorithm based on culture algorithm and cloud model is designed to solve the problem. The operations of this algorithm, including coding, crossover and mutation are discussed in detail. Compared with method [7], the convergence effect and calculation efficiency are better. In future research, how to deal with stochastic task would be considered in agent coalition problem.

## Acknowledgements

This work was supported by Zhejiang Province Postdoctoral Foundation of China under grant BSH1402043 and Zhejiang Province Natural Science Foundation of China under grant LQ15G010007.

## References

[1] Cheng Bailiang, Zeng Guosun, Jie Anquan. Journal of Computer Research and Development, 2010, 47(8): 1382-1391.(In Chinese)

[2] Xiaohan Yu, Zeshui Xu. Knowledge-Based Systems, 2012, 35: 271-278.

[3] Dayong ye, Minjie Zhang, Danny Sutanto. IEEE Trans on Parallel and Distributed Systems, 2013, 24(5): 1042-1051.

[4] Xu bo, Peng Zhiping, Yu Jianping, et al. Systems Engineering-Theory & Practice, 2012, 32(10): 2253-2261.(In Chinese)

[5] Su Shexiong, Hu Shanli, Lin Chaofeng, et al. Journal of Computer Research and Development, 2007, 44(2): 277-281.(In Chinese)

[6] Jingan Yang, Zhenghu Luo. Applied Soft Computing, 2007, (7): 561-568.

[7] Liu Sainan, Chen Mingliang. Control and Decision, 2014, 29(9): 1724-1728.(In Chinese)