Improved Multi-objective Evolutionary Algorithm for Multi-agent Coalition Formation

Bo Xu
Department of Computer Science and Technology of Guangdong University of Petrochemical Technology, Guangdong Province, P.R.China
Email: xubo807127940@163.com

Ruizhe Zhang
Computer Science and Technic Academy Department, Pingdingshan University, Henan Province, P.R.China
Email: pdszrz@163.com

JianPing Yu
College of Mathematics and Computer Science, Key Laboratory of High Performance Computing and Stochastic Information Processing,Hunan Normal University,Hunan Province, P.R.China

***Abstract*—one of the key problems in multi-agent coalition formation is to optimally assign and schedule resources. An improved multi-objective evolutionary Algorithm (IMOEA) is proposed to solve this problem. Compared with several well-known algorithms such as NSGA, MOEA, experimental results show the algorithm is very suitable for coalition formation problem.**

***Index Terms*—Multi-agent; coalition formation; multi-objective optimization; evolutionary algorithm**

I. Introduction

With the demand of production practice and development of agent technology, the needs of people no longer limited to a single agent. The researchers have been interested in multi-agent system which composed by multiple agents. Multi-agent system has become a hot topic in artificial intelligence, because the multi-agent system has many advantages that can not be found in single agent. The current important trend of agent technology is highly intelligent and group.

On the one hand, the researchers design and develop a more sophisticated independent intelligent agent, which is a more complex individual that full of information perception, target recognition, decision analysis, optimization and search integrated; on the other hand, with the idea of swarm intelligence in recent years, people use a number of relatively simple agent composition of groups to implement complex tasks that some single or small number of agent can not complete [1].

How to generate optimal agent coalition is a key issue of multi-agent system. Researchers generally believe that the problem was first introduced around 1993, and then this problem was followed by a variety of important conferences, academic journals and research reports, coalition problems became an important research direction. Sandholm[2] proved that finding coalition structure was a NP complete problem. DeVany [3] found that with the number of agent increasing, in order to quickly find the optimal coalition structure was very difficult. He thought it should look for sub-optimal instead of optimal solution, and used information theory to measure the complexity of coalition structure. But he did not provide this algorithm can be used to find the sub-optimal solution. Jingan Yang [4] used genetic algorithm for agent coalition, Na Xia[5] utilized ant colony optimization algorithm for this problem, Guo-fu Zhang[6] introduced the particle swarm optimization algorithm for solving complex coalition problems, etc. Many algorithms were proposed to improve the quality of solutions, but they were for single task of a single objective optimization [7]. In real life, the system may have multiple tasks at any time; such problem was called multi-task multi-coalition [8]. Chao-feng Lin [9] used ant colony optimization algorithm in multi-task environment, Zhao-pin Su [10] used the immune algorithm for multi-task coalition parallel study.

Until now, many algorithms and methods were proposed to coalition problems, But the convergence of the existence of the above algorithm was slow, while global optimization was not strong [11, 12]. In this paper, we did further researches on agent coalition formation problem in dynamic environment and improved the speed of finding pareto optimal solution [13, 14]. A model of multi-objective agent coalition formation was proposed.

II. Backgrounds

A. Multi-objective Evolutionary Algorithm (MOEA)

Several multi-objective evolutionary algorithms (MOEA) were proposed from early 1990s. Such as

Manuscript received January 1, 2013; revised June 1, 2013; accepted July 1, 2013.

Copyright credit, corresponding author: Bo. Xu .

doi:10.4304/jsw.8.12.2991-2995

MOGA, NPGA, NSGA, SPGA, NSGA-II, SPEA2, etc[15,16]. Multi-objective optimization problem is a complex problem often encountered in engineering. Multi-objective evolutionary algorithm has been successfully applied in many fields such as electronics, environmental science, finance, economics, geometry, physics information and resources.

B. Model of Multi-objective Agent Coalition Formation

In this paper, we assume that: $A = \{A_1, A_2, ...A_n\}$ represents n agents and that each has r -dimension capability vector $B_i =< b_i^1, b_i^2, ...b_i^r >$, $b_i^j \geq 0$, ($1 \leq i \leq n$, $1 \leq j \leq r$), where each capability is a property that quantifies the ability to perform an action.

$T = \{t_1, t_2, ...t_m\}$ represents m tasks and that a set of capability requirement vector $B_{t_I} =< b_{t_I}^1, b_{t_I}^2, ...b_{t_I}^r >$. $CS = \{C_1, C_2, ...C_m\}$ represents agent coalition. C_i is a group of agents that decide to cooperate to perform a task and each coalition performs a single task t_i . C_i has r -dimensional capability vector B_c , B_c represents the sum of the capabilities. C_i can perform a task t_i only if its capability vector satisfies $B_{t_i}^j < B_{C_i}^j$, for each C_i , there exist coalition cost $Cost_{c_i}$ and coalition value $Value_{c_i}$. $Min(Cost(T))$ represents minimization of critical cost, $Max(\Pr ofit(T))$ represents maximization of total of all profit.

The multi-objective agent coalition is to find a set of solutions that are superior among all the solutions when all objectives are considered. The agent coalition is a multi-objective optimization problem and can be depicted as follows:

$$Max(\Pr ofit(T))$$
$$Min(Cost(T))$$
$$Min(Time(T))$$
$$\text{restriction:} Time(T) \leq Max^T(T) \qquad (1)$$

Where $Time(T)$ is the time that all tasks of system are completed.

III. Algorithm Design

A. Initialization and Mutation Operator Figures and Tables

The population size depends on the nature of the problem but typically contains several hundred or thousands of possible solutions. Traditionally, the population is generated randomly, covering the entire range of possible solutions (the search space). Occasionally, the solutions may be “seeded” in areas where optimal solutions are likely to be found.

Mutation operator is the core of the multi-objective evolutionary Algorithm. It will have a direct impact on the performance of the algorithm is shown. In IMOEA, due to chromosomal or entangled in a superposition state of the state, updates can not be used in the traditional GA mutation operation.

The specific methods are as follows: the variation of the probability increases while small chromosome individual, this way can maintain the diversity of the population, and search for a broader solution space to avoid falling into the local optimum. Decrease in mutation probability while large individual chromosome,, so that this way can reduce the amount of computation and ensure the diversity of groups. Therefore defined mutation probability:

$$P_m = 0.1(1-\alpha) \quad (2)$$

Mutation operation would also undermine the good general values of the individual; the mutation probability is smaller in biological evolution, multiplied by 0.1. P_m Knowable range is [0, 0.1].

B. The Selection of Global Optimal Solution and Local Optimal Solution

The key question of MOEA is the choice of the optimal solution and the global optimal solution. It is natural to be selected by objective function value in single objective problems. But the objective function value is difficult to be determined by which solution better in the multi-objective optimization problems, which can not be selected. So this approach is taken, for the selection of the global optimal solution, all the Pareto solutions in accordance with the sort of potential future value of F , the greatest potential value is the global optimum solution. For the history of each individual i select the optimal solution, using the following steps:

① If the current solution i dominated the history of the individual optimal solution, then update the history of the optimal solution.

② If the current solution did not dominate the history of individual it’s optimal solution, then compare the current optimal solution and the history of the $D(i)$ value, choose a smaller $D(i)$ the value of the solution as the history of the optimal solution.

C. The Procedure of IMOEA

Begin

a) $t = t + 1$;

b) Initial population;

While (not termination-condition) do

i) Evaluate $P(t)$, sample it once and get one non-dominated solution; Store the best non-dominated solution among $P(t)$ into $A(t)$

ii) Compare the non-dominated solutions with the current $A(t)$.

iii) Through non-dominated sorting and crowding distance sort rebuild the archive set $A(t)$;

$$A(t) \subseteq M_f(P(t) \cup A(t-1), \leq)$$

iv) generating target solution set $O(t)$,

$$O(t) \subseteq P(t) \cup A(t),$$

vi) Update $P(t)$;

c) If stopping condition is satisfied, then stop.

End

IV. Experimental Results and Analysis

We design experiments of NSGA [7], MOEA and IMOEA. The parameters of experiments are set as follows: tasks require the ability vector $B_l = < b_l^1, b_l^2, b_l^3, b_l^4 >$, agents establish the different ability vector $A_i = < b_i^1, b_i^2, b_i^3, b_i^4 >$, $Cost_{A_i}$ =1* b_i^1 +2* b_i^2 +3* b_i^3 + 4* b_i^4 ; $Profit_l$ =1* b_i^1 +2* b_i^2 +3* b_i^3 +4* b_i^4 . In experiments set the number of agent = 200, three typical application environments were the given for multi-task coalition.

Environment 1: $B_{all} = \sum_{i=1}^{n} B_{A_i} < B_l$, three algorithms can not generate agent coalition.

Environment 2: $B_{all} = \sum_{i=1}^{n} B_{A_i} \geq B_l$, three algorithms can generate agent coalition, but the quality of the results are different (see Table 1,2).

Environment 3: $B_{all} = \sum_{i=1}^{n} B_{A_i} >> B_l$, three algorithms can generate agent coalition, but the quality of the results are different (see Table 1,2).

TABLE 1

COMPARISON OF THREE ALGORITHMS FOR COALITION VALUE (STATISTICAL OF 100)

Environment	Number of tasks	Maximum coalition value		
		IMOEA	MOEA	NSGA
1	4	0.0	0.0	0.0
	8	0.0	0.0	0.0
	50	0.0	0.0	0.0
2	4	0.9889	0.9412	0.8332
	8	0.9756	0.9434	0.8011
	50	0.9659	0.9362	0.7832
3	4	0.9843	0.9491	0.8180
	8	0.9789	0.9344	0.7983
	50	0.9676	0.9201	0.7792

TABLE 2

COMPARISON OF MULTI-OBJECTIVE EVALUATION OF THE OPTIMAL COALITION (STATISTICAL OF 100)

Environment	number of tasks	$Time(T)$ (Seconds)			$Max(Profit(T))$			$Min(Cost(T))$		
		NSGA	MOEA	IMOEA	NSGA	MOEA	IMOEA	NSGA	MOEA	IMOEA
1	4	-	-	-	-	-	-	-	-	-
	8	-	-	-	-	-	-	-	-	-
	50	-	-	-	-	-	-	-	-	-
2	4	7.65	6.32	5.33	0.303	0.923	0.971	0.65	1.36	1.31
	8	12.55	10.13	9.82	0.374	0.903	0.962	0.62	1.38	1.33
	50	44.09	36.31	32.21	0.357	0.894	0.933	0.61	1.43	1.42
3	4	7.98	6.43	5.67	0.336	0.880	0.964	0.65	1.43	1.41
	8	12.02	10.31	9.71	0.383	0.872	0.953	0.63	1.51	1.45
	50	44.86	38.32	35.97	0.322	0.804	0.924	0.62	1.69	1.44

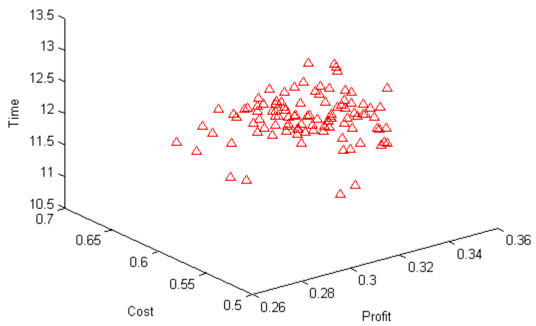

Figure 1 Pareto solution distribution based on NSGA (Task=8, Environment 2)

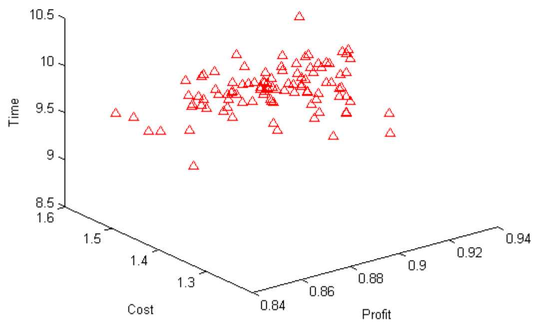

Figure 2 Pareto solution distribution based on MOEA (Task=8, Environment 2)

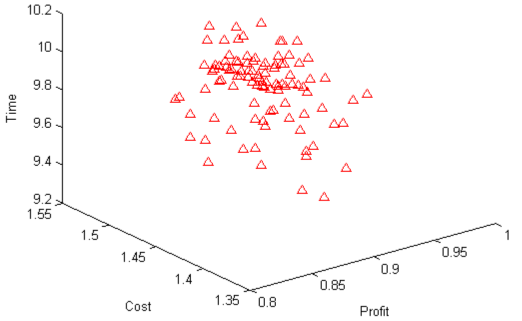

Figure 3 Pareto solution distribution based on IMOEA (Task=8, Environment 2)

Pareto solution distribution based on three algorithms is shown in Figure 1,2,3. The statistical comparisons are shown in Table 1 and Table 2. Table 1 shows the best coalition value found by MOEA, NSGA and IMOEA. Table 2 shows multi-objective evaluation of optimal coalition.

From Table 1, 2, it can be seen that IMOEA can get better results. We can see that in the environment 1, task can not be solved by the three algorithms. In the environment 2, three algorithms can generate agent coalition, but the quality of the results is different, the sum of coalition value of our algorithm is the largest, and our algorithm has the highest utilization rate of resources. And results are the best. In the environment 3, MOEA and NSGA have a great waste of resources, coalition value is very small, and the advantages of our method are obvious.

From Figure 1, 2, 3, we can see our algorithm also has the fastest convergence.

The above results show that our algorithm can generate multi-task coalition in typical conditions promptly and efficiently, and the solution quality is better than other two algorithms, and can avoid the deadlock of the coalition and waste of resources.

V. Conclusions

Agent coalition is a key issue of multi-agent system, there are still many areas to be studied, especially for dynamic environments. We face not only the complexity of the agent coalition formation, but also the actual application process including not only the combination of resources, task allocation, but also co-verification, co-simulation and other follow-up steps. Of course these are the next step work.

Acknowledgment

This work was supported by the opening Project of Universities of Guangdong Petrochemical Trouble shooting Control Engineering and Information Technology Development Center (No.512016), Maoming Municipal Science and Technology Project (No. 201202 63). Opening Project of Jiangsu High Technology Research Key Laboratory of Wireless Sensor Networks (WSNLK201102).

References

[1] Shehory O,Kraus S . Feasible Formation of Coalition among Autonomous Agents in Non-Super-Additive Environments [J]. *Computational Intelligence*,1999,15(3) :218-251

[2] Sandholm T W, Lesser V R. Coalition among Computationally Bounded Agents [J]. *Artificial Intelligence*, 1997, 94(1): 99-137

[3] A. De Vany. Information, bounded rationality, and the complexity of economic organization [J], *Taiwan. Journal of Political Economy*,1996,2(1):21-44

[4] Jingan Yang, Zhenghu Luo. Coalition formation mechanism in multi-agent systems based on genetic algorithms[J]. *Applied Soft Computing*,2007,7(2):561-568

[5] Jian-guo Jiang, Na Xia. Searching for Agent Coalition for Single Task Using Improved Ant Colony Algorithm [J]. *Journal of Computer Research and Development*,2005,42(5):734-739

[6] Guo-fu Zhang, Jian-guo Jiang, Na Xia, ect. Solutions of Complicated Coalition Generation Based on Discrete Particle Swarm Optimization [J]. *Acta Electronica Sinica* , 2007, 35(2):323-327

[7] Bo Xu, Qing Guan, Ke Chen. Multi-Agent Coalition Formation Based on Quantum-behaved Particle Swarm Optimization [J]. *Journal of Information and Computational Science*, 2010,7(5):1059-1064

[8] Guo-fu Zhang, Jian-guo Jiang, Zhao-pin Su, ect. Searching for overlapping coalitions in multiple virtual organizations [J]. *Information Sciences: an International Journal*, 2010,180(17):3140-3156

[9] Chao-feng Lin, Shan-li Hu, Sheng-fu Zheng, ect. Multi-task Coalition Generation Algorithm Based on An Improved Ant Colony Algorithm [J]. *Journal of Computer Research and Development*,2006,43(1):176-181

[10] Zhao-pin Su, Jian-guo Jiang, Na Xia, ect. Multi-task coalition parallel generation algorithm based on dimension partition strategy and immunity [J].*Systems Engineering-Theory & Practice*,2008, 28 (1):118-123

[11] Talal Rahwan, Sarvapali D. Ramchurn, Nicholas R. Jennings, ect. An Anytime Algorithm for Optimal Coalition Structure Generation[J].*Journal of Artificial Intelligence Research*,2009, 34(1): 521-567

[12] Travis C. Service, Julie A. Adams. Coalition formation for task allocation: theory and algorithms[J]. *Autonomous Agents and Multi-Agent Systems*,2011,22(2):225-248

[13] Xiaofang Guo, Yuping Wang. A Hybrid Simplex Multi-Objective Evolutionary Algorithm Based on A New Fitness Assignment Strategy [J]. *Journal of Computers*, 2013, 8(2):284-289

[14] Lovekesh Vig, Julie A. Adams. Coalition Formation: From Software Agents to Robots[J].*Journal of Intelligent and Robotic Systems*,2007,50(1):85-118

[15] Yajuan Yang,Wenxue Niu. Multi-Agents Model and Simulation for the Evolution of Industrial Clusters [J].*Journal of Computers*, 2013, 8(2):326-333

[16] Yu Chen, Xiufen Zou, Weicheng Xie. Convergence of multi-objective evolutionary algorithms to a uniformly distributed representation of the Pareto front [J]. *Information Sciences: an International Journal*, 2011, 181(16):3336-3355

Bo. Xu He is a Ph.D, CCF Senior Member, ACM Member, who now works at Department of Computer Science and Technology of Guangdong University of Petrochemical Technology. He was born in 1982. His

main research interests include computational intelligence and cloud computing.

Ruizhe Zhang He is a master, who now works Computer Science and Technic Academy Department, Pingdingshan University. He was born in 1971. His main research interests include network management architecture and management mechanism.

JianPing Yu He is a doctor, who now works at Hunan Normal University. He was born in 1979. His main research interests include computational intelligence.