



(RESEARCH ARTICLE)



Task assignment based on improved contract net in dynamic environment

Wang Mingqian ^{1,2,*} and Cao Shuai ^{1,2}

¹ School of Information Engineering, Changzhou Vocational Institute of Mechatronic Technology, Changzhou, Jiangsu Province, P.R.China.

² Jiangsu Internet of Things and Manufacturing Information Engineering Technology Research and Development Center, Changzhou, Jiangsu Province, P.R.China.

Publication history: Received on 15 March 2020; accepted on 21 March 2020

Article DOI: <https://doi.org/10.30574/gjeta.2020.2.3.0018>

Abstract

Task assignment algorithm research is an important direction in multi-agent research. The traditional contract-based task allocation algorithm has a low efficiency problem, and the problem will be more prominent in a dynamic environment. In the dynamic environment, an improved contract network method is proposed for the task allocation algorithm of multi-agent system. This method considers task familiarity and load balance, and improves the efficiency of multi-agent system allocation. Simulation results show that this paper shows that the method improves the system revenue.

Keywords: Dynamic environment; Improved contract net; Task familiarity; Task assignment

1. Introduction

Task assignment is the basis of MAS operation, which is related to how the system completes tasks, how to coordinate between agents, etc., which has a decisive influence on system goals or performance. The purpose of task assignment is to assign the tasks of the system to the appropriate Agent to optimize the performance of the task execution.

Different MAS control architectures also generate different task assignment methods. The typical control architectures in MAS are fully centralized, distributed, decentralized, and federated. Ferber classifies the task assignment problem of MAS [7-8], which is divided into two types: contracted allocation and emergent allocation.

The idea of emergent task assignment comes from the social organization of social organisms in nature. For example, ants, bees, fishes and other lower animals can form a stable structure through limited cognitive ability and simple local interaction. An effective division of labor is carried out to accomplish the overall task that the individual cannot accomplish alone, reflecting the emergence of the system. The emergent distribution method achieves coordination through the interaction of Agents' interactions with each other and the response to the environment. Since robots do not need explicit direct communication between them, the requirements for robot intelligence are not high, so it is more suitable for tasks and agents. Big situation. However, the use of the emerging task assignment method cannot accurately predict the behavior of the Agent in the system, so the analysis is very difficult and cannot guarantee efficiency.

In the agreed-upon allocation method, explicit communication and information exchange are required between Agents, which is more suitable for the actual tasks that humans arrange for Agents to complete. Based on the agreed-type allocation, it can be divided into centralized distribution and distributed distribution. The centralized distribution can be further divided into mandatory allocation and trader allocation. Distributed distribution can be divided into

* Corresponding author: Wang Mingqian

acquaintance network allocation and contract network allocation. The centralized task allocation method is simple to implement and easy to generate the global optimal solution. However, the method is concentrated in communication, which is easy to cause communication congestion, and the algorithm has high computational complexity, which is suitable for task assignment in small scale and certain environment. Distributed task allocation, can quickly adjust the program, Agent can dynamically join or exit, and the communication is more dispersed, can avoid congestion, the disadvantage is that it is easy to fall into local optimum, and when the task is adjusted, it will increase the communication burden of the system, suitable for Systems in medium to large scale systems and dynamic environments.

For specific application problems, the appropriate task allocation method should be selected according to the MAS control architecture and the specific application environment.

The dynamic environment is a complex situation. In [6], the dynamic conditions appearing in the MAS are prioritized, in order: (a) the original performer exits; (b) the new task joins; (c) the new performer joins (d) changes in threats; (e) cancellation of original tasks; (f) changes in the capabilities of the original performers; (g) changes in the capabilities of the performers required for the original tasks. In a dynamic environment, the redistribution of tasks is triggered. The task redistribution in the centralized control mode is to take the dynamically changed conditions as the initial state, and then complete the allocation again, so as to ensure that the optimal solution of the system can be obtained, and the disadvantage is that the traffic and the calculation amount are too large; Distributed task redistribution generally re-adjusts the tasks of local changes on the basis of the original, reduces the traffic, and improves the response speed of the system.

Contract network technology is a kind of distributed task allocation method. There are many researches at present, and its algorithm flow is simple and easy to implement. This paper studies the contract network allocation algorithm, gives the deficiency of the traditional contract network in the dynamic large-scale environment, and proposes an improved contract network algorithm.

2. Introduction of Task Allocation based on Contract Network

The contract network was proposed by Smith in the study of distributed problem solving in 1980 [3], which is a commonly used negotiation strategy in distributed task allocation. The agent nodes in the contract network model can be divided into: (a) the bidder: the owner of the task, responsible for the assignment and management of the task; (b) the bidder: according to their own status and evaluation of the task, choose to participate in the bidding, The tenderer issues a mission request; (c) the successful bidder: the winner of the final bid is responsible for completing the mission.

The general steps of the contract network include the following steps:

2.1. Tendering

After receiving a task, the task manager splits the large task into several subtasks according to certain rules, and then selects the decomposed task for allocation. At this point, wait for the participant's response information until the deadline for bidding.

2.2. Bid

After receiving the bid, other Agents will submit bid responses to the task initiator according to their own ability limits and evaluation of the tasks, including "request" and "reject". At the same time, the participants also monitor the release of new tasks. If there is a new task, the participants will continue to evaluate and decide whether to bid to the sponsor.

2.3. Win the bid

The initiator evaluates the bidder of the task, then selects the best participant and sends a task execution invitation to it. If the participant accepts the task execution invitation and sends the acceptance message to the initiator, the initiator issues a task to perform an "Accept" confirmation and sends a "Reject" message to the other participants. Otherwise, the initiator needs to re-tender the task. 4) Execution. The participant performs the task, and the task tenderer monitors the execution of the task. The successful bidder needs to submit the task execution result to the initiator: "failed", "rejected" or "completed".

Although the above negotiation process can realize the task assignment, there are still many shortcomings in the traditional contract network protocol, which affects the actual negotiation process and the efficiency of task assignment. These deficiencies include the following aspects:

2.4. Announcement of tenders

In the contract network bidding process, the announcement of the mission is broadcast, so that all agents can receive the bidding documents. However, this not only increases the burden of network communication, but also the task manager must evaluate all the bids and waste a lot of system resources.

2.5. Coexistence of multiple tenderers and confirmation of contracts

In the bidding process of the task participation, the bidder may receive another bidding document while waiting for the contract confirmation. At this time, if the agent waits for the bidder to send the confirmation message, the new task needs to wait for a certain period of time to reply. Make the whole allocation process slower; if the agent continues to bid, there may be cases of simultaneous bidding, so the bidding strategy needs to be improved. In another case, after the tenderer issues the contract to the successful bidder, he still needs to wait for the confirmation of the successful bidder. Before the confirmation information arrives, the manager does not know whether the successful bidder will accept the contract. At this time, the contract constraint is not established. If the successful bidder refuses to accept the contract, the tenderer will need to re-select the successful bidder.

2.6. Communication requirements

After the contract is formed, there is a constant communication between the tenderer and the successful bidder to maintain contact during the execution of the mission, and the tenderer supervises the successful bidder, which increases the requirements for the communication network.

For the selection of the winning agent, comprehensive consideration and trade-off from the three aspects of the load, capability and trust of the bidding agent [3], a dynamic task allocation algorithm based on multi-attribute evaluation winning strategy, thus effectively improving the task assignment. And the efficiency of execution. Its ability and trust are determined by the completion of past tasks.

In the literature [4], based on the classic contract network protocol, by introducing the strategies of mutual trust, acquaintance trust and devaluation, a computer generated force (CGF) suitable for multi-agent system (MAS) is proposed. Collaborative integrated contract network protocol, which can reduce the scope of bidding, effectively reduce the communication cost of collaboration, and improve the system's completion task indicators.

In the literature [5], according to the dynamic situation between Agents, the distribution of resources and the physical and logical communication distances are considered at the same time, so as to determine the specific allocation of tasks, thereby improving system performance.

In the literature [6], after considering the load balancing degree, the Agent task is allocated and adjusted to achieve uniform agent load.

3. Contract network improvement

The task assignment problem mainly involves the following: the characteristics of the assigned task, the workload of the bidding agent, the ability and the trust degree. First, the task assignment is formalized with a six-tuple: $\langle T, A, L, C_A, C_T, S \rangle$.

Where T is a set of task sequences to be assigned $T = \{T_1, T_2, \dots, T_N\}$, each subtask can be independently executed by a single agent. A is a set of Agents, which is $A = \{A_1, A_2, \dots, A_M\}$. L is the workload of the current Agent $L = \{L_1, L_2, \dots, L_M\}$. C_A is a $M \times N$ matrix that C_A^{ij} represents the ability of the Agent A_i to complete the type tasks T_j . C_T is a N -dimensional vector that C_T^j represents the ability required to complete the task T_j . S is also a $M \times N$ matrix that S_{ij} represents the familiarity of the Agent A_i and the task T_j .

The purpose of task assignment is to minimize the cost of the system to complete the task, while achieving a balance of workload between agents. The agent's workload is mainly reflected in the task distance, the amount of time required for the task to complete, and the number of tasks. Define the workload of Agent m in the system as L_m . Average workload:

$$\bar{L} = \frac{\sum_{i=1}^M L_i}{M} \quad (1)$$

Where M is the number of agents. Then the task load factor of Agent i is $WL_i = (L_i - \bar{L})/\bar{L}$. The load factor is expressed as: when $WL_i > 0$, the indicated task load is higher than the average level of the Agent in the system. When $WL_i < 0$, the indicated task load is lower than the average level of the Agent in the system.

For the execution of the Agent, after receiving the task invitation, it will evaluate its own capabilities and effectiveness. The task performance is the cost of the agent to complete the task minus the corresponding cost. The income after completing the task T_j is determined by the value of the task and the cost that the agent can perform. The costs of UCAV's execution tasks include path cost (fuel cost and time cost) T_j and risk cost (probability of being discovered and attacked by enemy threats). After the execution agent receives the bidding document, it evaluates itself. If the requirements of the task are met, the bidding will be carried out, and the bid value is its own capacity and distance consumption value.

Because in the dynamic environment, the ability of the agent is constantly changing, and the assigned tasks may be at a certain time, due to the change of the capabilities of the agent, the efficiency of the task execution is reduced. For this reason, the management agent needs to know the law of the change of the agent capability. That is to know the ability of the Agent at the time of final execution. If the final agent performs a task, its ability is lower than the minimum capacity requirement of the task, the system's revenue will be relatively low, and when it is greater than, the task can be completed normally, and there will be higher returns.

Defining the trust degree of the management agent for the bidding agent, the value ranges from 0 to 1. If it is '1', the ability value of the agent can be known exactly. On the contrary, the ability of the management agent to identify the bidding agent is inaccurate within a certain range. Value, there is a certain difference from the real ability.

The degree of trust is used to bid through the trust. The trust is derived from the completion of the previous tasks. It means that one Agent i believes that another Agent j can successfully complete the task and is the trust of the Agent. When there are more bidding objects, you can limit the degree of trust to control the scope of bidding. When the agent successfully completes the task, the manager will increase the trust level of the task, which is expressed as $S_{ij} = \min\{1, S_{ij} + \delta\}$; when the agent fails to complete the task as required, the trust value is lowered. For the initial time, the trust value of the task for each agent is zero. At this time, the unfamiliar agent is optional, and the task is broadcasted at this time; if the agent capability changes, the task cannot find a suitable agent from the trust list. When commissioning, you also need to broadcast the task.

4. Experimental Results and Analysis

Considering the example of a multi-UAV system UCAV, in a region with a length of 200 Width 150, there are 30 tasks, and 8 drones perform task execution. The positions of 30 tasks are randomly generated, and 8 agents are randomly generated on one boundary of the region.

In the experiment, the load of the task only considers the length of the flight route. When the drone reaches the mission point, it indicates that the mission is completed. At the same time, the revenue of the mission completion is related to the capability of the drone. The higher the capability, the greater the benefit. In this experiment, if the agent's ability is greater than the required capacity of the task, the gain is that the agent's ability is more than the other part, and the difference is doubled.

The task is generated as follows, the blue dot represents the task, and the red is the agent.

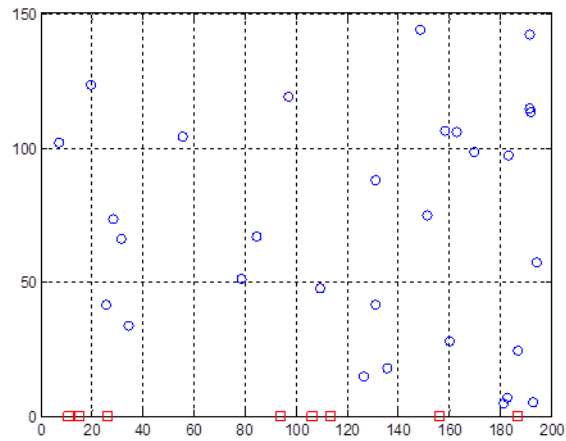


Figure 1 Task generation

Without considering load balancing, just simple random assignment of tasks, the resulting graphics are as follows:

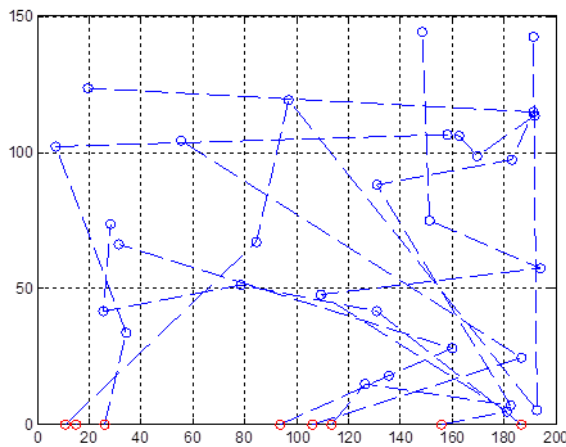


Figure 2 Task random assignment result of initial configuration

It can be seen that there are many connections in each point in the figure, the total load is relatively large, and it is not balanced, and the revenue at this time is relatively large. When considering the load balance degree, at this time, the task assignment result is as follows: the revenue at this time is 1.4119, and the network balance is also relatively large.

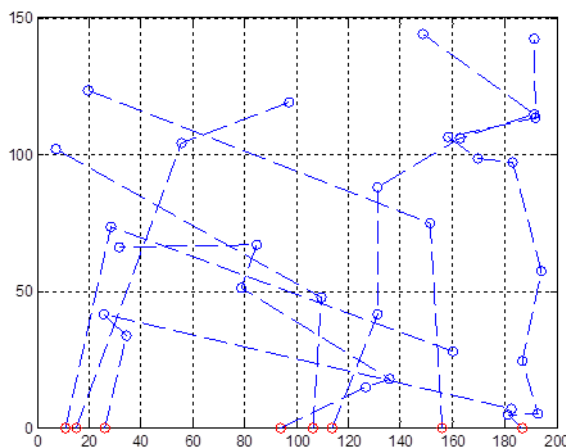


Figure 3 Task assignment results after considering load balancing

When the trust is increased, the distribution result of the network at this time is as follows: the efficiency is 4.3937.

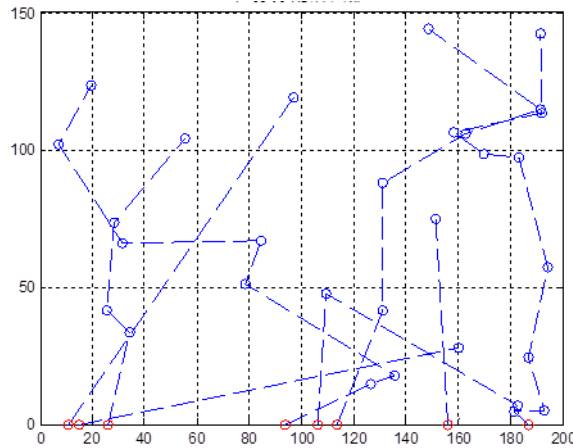


Figure 4 Task assignment result after increasing trust

4.1. Redistribution of tasks in a dynamic environment

4.1.1. Consider only the task load. If the green node in the figure is invalid, then the subsequent tasks need to be re-divided. The result of the division is as shown in the following figure: the benefit of the latter half is -2.7903.

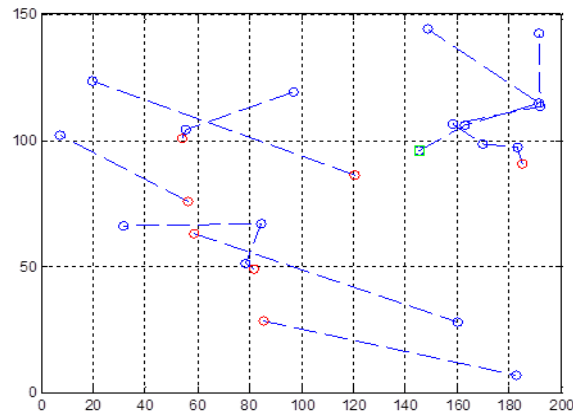


Figure 5 Considering UAV failure during task execution

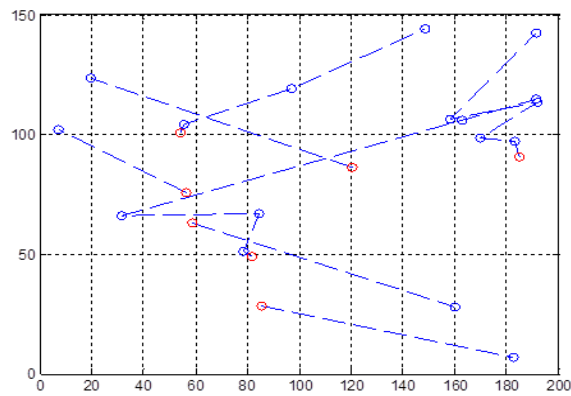


Figure 6 Task redistribution result after removing failed nodes

4.1.2. When considering the ability to perform familiarity, if one of the agents fails, the green node in the figure is invalid. As a result, the remaining mission revenue was 1.5800.

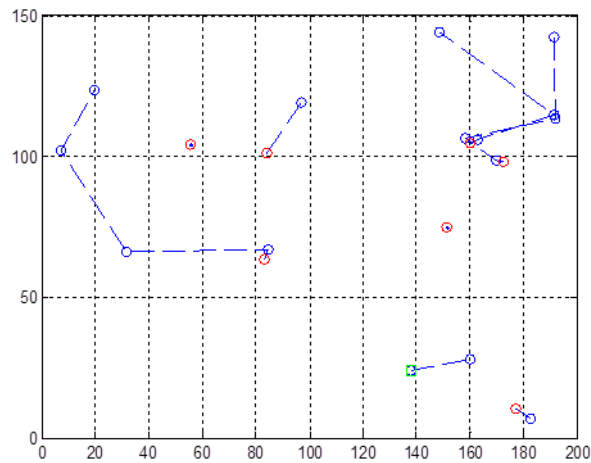


Figure 7 Considering the failure node that performs capability familiarity

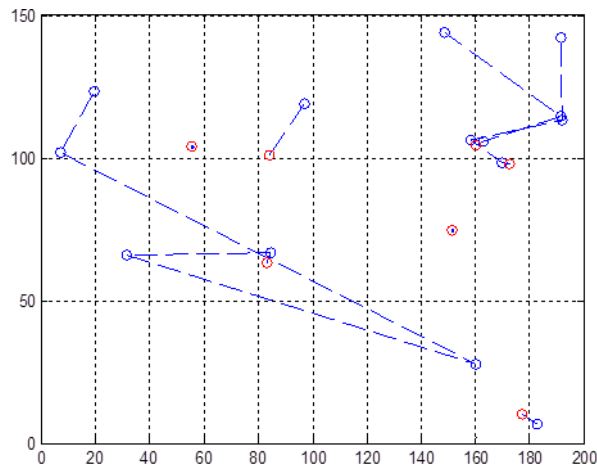


Figure 8 Task redistribution result after removing failed nodes

It can be seen that in static or in a dynamic environment, when considering the familiarity, the overall benefit of the system operation is relatively large, and the effect and performance obtained by considering only the load degree are better.

5. Conclusion

Based on the traditional contract network, this paper considers the changes of Agent and task ability under dynamic application, and defines the familiarity of the task. Through simulation experiments, it can be seen that the final system benefits are improved. Suitable for dynamic environments.

Compliance with ethical standards

Acknowledgments

The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

This work is supported by the Natural Science Foundation of the Higher Education Institutions of Jiangsu Province under Grant no.19KJB510019, Innovation and Entrepreneurship Training Program for College Students of Jiangsu Province under Grant no. 201913114004Y, Changzhou Key Laboratory of Industrial Internet and Data Intelligence under Grant no.CM20183002 and the Project of Changzhou Vocational Institute of Mechatronic Technology under Grant no. 2019-YBKJ-05.

Disclosure of conflict of interest

The two authors, Wang Mingqian and Cao Shuai, all declare that there is no conflict of interest regarding the publication of this paper.

References

- [1] Li Haigang and Wu Qidi. (2003). Review of Multi-Agent System Research [J]. Journal of Tongji University, 31(6), 728-732.
- [2] Hu W, Liu L and Feng G. (2017). Consensus of Linear Multi-Agent Systems by Distributed Event-Triggered Strategy [J]. IEEE Transactions on Cybernetics, 46(1), 148-157.

- [3] Qiu Hangping, Qin Ang et al. (2012). Research on Task Assignment Based on Improved Contract Net Model in Multi-Agent System [J]. *Computer Science*, 39(6A), 279-282.
- [4] Chen Jian, Liao Shouyi and Deng Fanglin. (2010). A Multi-Agent Based Computer Generated Force Collaboration Method[J]. *Computer Simulation*, 27(2), 113-117.
- [5] Jiang Yichuan. (2012). Task Assignment of Network Structured Multi-Agent System [J]. *Pattern Recognition and Artificial Intelligence*, 25(2), 262-272.
- [6] Li C J and Liu GP. (2018). Data-driven consensus for non-linear networked multi-agent systems with switching topology and time-varying delays [J]. *Iet Control Theory & Applications*, 12(12), 1773-1779.
- [7] Jin Zhouaba, Small M and Lu J A. (2017). Flocking of multi-agent dynamical systems based on pseudo-leader mechanism [J]. *Systems & Control Letters*, 61(1), 195-202.
- [8] Liu W, Wei G, Sheng W, et al. (2017). Decentralized Multi-Agent System-Based Cooperative Frequency Control for Autonomous Microgrids With Communication Constraints [J]. *IEEE Transactions on Sustainable Energy*, 5(2), 446-456.

How to cite this article

Wang M and Cao S. (2020). Task assignment based on improved contract net in dynamic environment. *Global Journal of Engineering and Technology Advances*, 2(3), 45-53.
