A novel approach for multi-agent cooperative pursuit to capture grouped evaders

Muhammad Zuhair Qadir\textsuperscript{1} • Songhao Piao\textsuperscript{1} • Haiyang Jiang\textsuperscript{2} • Mohammed El Habib Souidi\textsuperscript{3}

Published online: 12 September 2018 © Springer Science+Business Media, LLC, part of Springer Nature 2018

Abstract

An approach of mobile multi-agent pursuit based on application of self-organizing feature map (SOFM) and along with that reinforcement learning based on agent group role membership function (AGRMF) model is proposed. This method promotes dynamic organization of the pursuers’ groups and also makes pursuers’ group evader according to their desire based on SOFM and AGRMF techniques. This helps to overcome the shortcomings of the pursuers that they cannot fully reorganize when the goal is too independent in process of AGRMF models operation. Besides, we also discuss a new reward function. After the formation of the group, reinforcement learning is applied to get the optimal solution for each agent. The results of each step in capturing process will finally affect the AGR membership function to speed up the convergence of the competitive neural network. The experiments result shows that this approach is more effective for the mobile agents to capture evaders.

This study was supported by National Natural Science Foundation of China [Grant Number 61375081] and a special fund project of Harbin science and technology innovation talents research [Grant Number RC2013XXK010002].

Songhao Piao piaosh@hit.edu.cn
Muhammad Zuhair Qadir mzuhairqadir@gmail.com
Haiyang Jiang jhy786839364@163.com
Mohammed El Habib Souidi mohamed.souidi@hit.edu.cn

\textsuperscript{1} Department of Computer Science and Technology, Harbin Institute of Technology, Harbin, China
\textsuperscript{2} Department of Mathematics, Harbin Institute of Technology, Harbin, China
\textsuperscript{3} Department of Computer Science, University of Khenchela, Khenchela, Algeria
Keywords  Pursuit evasion · Multi-agent pursuit · SOFM · AGRMF · Reinforcement learning

1 Introduction

Multi-agent systems can work together to solve different complex problems in a given environment. These agents can take the shape of a robot and can help in various situations. In the mid-1940s, William Grey Walter, in the study of the world’s first artificial intelligent robot—the tortoise—found that these simple robots could perform some actions called “complex social behaviour” cooperatively [3]. Since then, significant progress has been made in the aspect theory and application of multi-agent systems. Multi-agent cooperative pursuit evasion has been a popular research area when it comes to games and other related problems [9]. Game theory provides simplest approach to solve it by using value and reward system to guide the agents towards its goal [7]. It provides coordination among them through mutual rationality of the multi-agents [8]. Moreover, Amigoni et al. [2] proposed a way to compute the optimal pursuer’s strategy that maximizes chances of target’s capturing in grid environment. It is based on theoretical and mathematical modelling. In [16], author highlights the necessary conditions and optimal pursuit strategy for successful chase by surrounding the evaders based on a mathematical model and geometrical principles.

Cooperation helps in robust and quick goal achievement of tasks ad goals [5, 6, 10]. In another study, Durham et al. [4] developed a strategy to handle clearing problem in which agents cooperatively guarantee to clear an environment given enough numbers of searchers are available.

With these researches, we were able inspire and propose our own algorithm which can better handle the problem with the help of reorganization to improve capture time.

2 Related work

Cooperative pursuit evasion is a known multi-agent problem which has attracted a lot of attention in the field of multi-agent robotics with a wide variety of applications such as search and rescue in disaster area and patrol on a larger scale, to name a few. It also has applications in home care, military combat and mobile sensor networks. Pursuit evasion is widely researched in differential games as well [8]. There are two key points in the implementation of coordination of multi-agent pursuit: one is organization and the other one is reorganization [14]. Organization means that each pursuer can find its own task by itself or it can just be assigned by control system according to its group’s situation and all of their social behaviour can be determined by their own ability and condition, the task of the whole group and the role played in the group. Reorganization is a capability of multi-agent system which promotes itself to move to a different organization which will be more adaptive to the new environment and hence can make the process of executing the task more effective than a previous organization. It is also referred as flexibility. Many useful and novel approaches are applied to improve cooperative multi-agent systems (MAS); some researchers focus on interaction between
the agents and the environment such as graph theory [1], polygonal environment [11], cellular environment [10], and others researchers focus on mechanisms of cooperation among human individuals. For instance, Chen et al. proposed a new hunting strategy based on optimized quick surrounding and quick-capture direction [16]. In order to simplify the process of communication of agents, Ze-Su et al. ameliorated a kind of task negotiation process based on multi-unit combinatorial auction [18]. Wang et al. [17] presented an alliance generation algorithm which can determine the cooperation intention according to the agent’s emotional factors, so that pursuers with the negative emotions can be prevented from participating in the group in case of a negative impact on the alliance. Souidi et al. [12] proposed an approach which can generate agents’ membership function called AGRMF which consists of their own credit and confidence just like human being, with the result that MAS can get optimized collaboration in grid environment making use of fuzzy logic.

These methods and researches have dramatically improved the state of the art of multi-agent cooperation pursuit. One of the most important problems that we are facing in this field is that pursuers cannot fully reorganize when evaders are too independent. This paper addresses this problem by proposing an algorithm for mobile multi-agent pursuit based on application of SOFM and reinforcement learning on AGR model; this method promotes dynamic organization of the pursuers groups and also makes the pursuers group evaders according to their will based on SOFM and AGRMF. It also helps the pursuers to be more dispersed and spread out as the process of chase goes on. In addition, a new reward function is discussed. Once the formation of the group is made, reinforcement learning technique is applied to get the optimal solution for each agent including the evaders. After each step, capture process becomes easier as it will affect the AGR membership function and also speed up the convergence of the competitive neural network. Finally, we analyse the effectiveness of the algorithm from the macro- and micro-simulation results.

3 Description of multi-agent pursuit problem

3.1 Problem description of pursuit evasion

The multi-agent systems for pursuit evasion problem based on AGR model consist of pursuers and evaders. The set of pursuers is denoted by P, and the set of evaders is denoted by E: \( P = \{p_1, p_2, \ldots p_n\} \) and \( E = \{e_1, e_2, \ldots e_n\} \). One of the features of the AGRMF model is that it gives pursuers two parameters that describe their capabilities learned from the process of pursuit: one is Confidence Conf: \( \forall \text{Conf} \in [0.1, 1] : \max(0.1, \frac{C_s}{C_t}) \) and another one is Credit Cred. \( \forall \text{Credit} \in [0.1, 1] : \min(1, 1 - \frac{C_b}{C_b + C_t}) \). Here, \( C_s \) is the number of tasks that has been accomplished successfully and \( C_t \) is the number of tasks in which the agent has participated. \( C_b \) is the number of the evaders abandoned by the agent. Each pursuer has its own pursuit range which is denoted by set \( R: R = \{r_1, r_2, \ldots r_n\} \); the evaders which come into \( r_n \) will be given priority to by \( p_n \). One of the pursuers will be selected to be the organizer, which will assign the pursuers to groups and cluster similar evaders to the same group according to AGRMF.
Pursuit difficulty means that how many pursuers are needed to catch that evader; the set of pursuit difficulties is denoted by $D = \{d_1, d_2, \ldots d_n\}$. Besides, each evader has its own priority, which is dependent on how many pursuers are near to that evader and secondly, inversely proportional to difficulty in catching that evader. So more pursuers close to that evader and lower the difficulty, the higher will be the priority assigned to that evader. Priority of an evader is denoted by $pr_i$:

$$pr_i = \frac{1 + P(e_i)}{d_i} \quad (1)$$

Here, $P(e_i)$ represents the number of pursuers whose pursuit range is invaded by evader $e_i$. One group consists of one or more pursuers and evaders, and the pursuers will try to pursue the evaders in that group. One evader will be considered to have been arrested by pursuers and dispersed from environment if there are enough pursuers or obstacles around him blocking his way as Eq. 2:

$$dn \leq \sum_{x \in \text{neibor}(dn)} P(x) + O(x) \quad (2)$$

Here, $\text{neibor}(dn)$ means the adjacent cell. $P(x)$ means the grid cell $x$ is whether occupied by pursuers or not, and $O(x)$ means whether that is an obstacle or not.

### 3.2 Agent group role membership function

Agent group role membership function (AGRMF) of pursuer $P$ to $E$ after the modification based on the expression proposed in the literature [11] is shown in Eq. 3 as:

$$\mu_e(p) = \frac{\text{Coef}_1 \ast \text{Dist}_{e,p} + \text{Coef}_2 \ast \text{Conf} + \text{Coef}_3 \ast \text{Credit}}{\sum_{i=1}^{3} \text{Coef}_i}, \quad (3)$$

where $\text{Dist}_{e,p}$ means Cartesian distance between $e$ and $p$. Ordinary $\mu_e(p)$ represents connection between pursuer $p$ and evader $e$. Coef1 to Coef3 are the main ability indicators of pursuer $p$ which are used to determine whether it can play a role in capturing $e$. The basic algorithm for coalition formation tries to greedily select enough pursuers with the highest $\mu_e(p)$ to join the group containing $e$ based on AGRMF[11].

### 4 Grouping evaders and path planning

The ordinary AGR model as extended with AGRMF creates one group for each evader which also means one group consists only one evader and the pursuers’ group only focuses on that evader letting alone others. This is the reason why the pursuers and evaders are relatively more discrete; therefore, the self-organization becomes less flexible when there are more evaders and in the bigger grid environment. This situation
is shown in Sect. 5. This subsection discusses a novel approach based on AGRMF which applies SOFM to generate one group consisting of more than one evaders.

The vector of AGRMF of evader \( n \) is denoted by \( U_{en} = \{ U_{en}(p_1), U_{en}(p_2), U_{en}(p_3) \ldots U_{en}(p_m) \} \). The set of \( U_{en} \) contains the chosen representation of evaders and is fed to SOFM to promote and learn how to group similar evaders. The basic structure of SOFM is inspired by AGRMF. The proposed algorithm we have used for training SOFM is also inspired by AGRMF, and through this, we get final group as an output. The algorithm works by first initializing the parameters and SOFM; then, the learning process of SOFM begins and is inspired by [13, 15]. Finally, the output is the indexes of groups each evader belongs to. The evader who gets the same output from SOFM will be assigned to the same group which will be discussed in detail in Sect. 5.

### 4.1 Path planning of multi-agent

Once the various pursuer alliances are formed, the next step is to determine the plan of motion. For this, we apply reinforcement learning as an agents’ motion strategy. Reinforcement learning is a classical approach which is famous for solving motion optimization problem in the environment where agents can get delayed reward. In this method, we developed a novel reward function to transform the theoretical model into its application in the mobile agent’s field. In reinforcement learning, agents can perceive the different states of the environment set \( S \), and the different actions they can execute set \( A \). For every discrete step, agents perceive the current state \( s_t \) and select an action \( a_t \) to execute it. Environment responds to this agent and returns reward \( r_t = (s_t, a_t) \); here, \( r \) is an immediate payoff function and a successor state \( s_{t+1} = \sigma(s_t, a) \) is then generated.

### 4.2 Immediate payoff function

The \( Re \) of evaders which determines the reward for the pursuers having caught it can get is equivalent to \( D \), \( Re_i \) is the reward in a certain region subject to normal distribution instead of occupying only one cell. The immediate payoff function of each cell which one pursuer gets from the organizer is sum of \( Re \) of evaders of the same group which that pursuer belongs to. The immediate payoff function of group is shown as Eq. 4 as below:

\[
  r_{\text{group}}(x, y) = \sum_{i \in \text{group}} Re_i pr_i e^{-\frac{1}{2}(x-x_i)^2+(y-y_i)^2},
\]

where \((x_i, y_i)\) is the Cartesian coordinates of the \( e_i \). This kind of immediate reward function can increase the attraction of the pursuers towards the evader along with its neighbours and centre of the group just as shown in Fig. 1. Peaks in the graph show the overlapping region of the evaders, targeting a region where the chances of capturing are increased and help surround the group effectively. The pursuers prefer to surround
all of them instead of focusing on only one of them, making it a very effective way to capture them all in groups.

4.3 Optimal strategy

Agent’s task is to learn a strategy $\pi : s \rightarrow a$ which selects the next action $a_t$ based on the current state $s_t$ and enables the agent to get the maximum of discounted cumulative reward $V_\pi = \sum_{t=0}^{\infty} \lambda^t r_{t+i}$ where $\lambda$ is the discount factor; agents uses the principle of $Q$ learning to complete the task, and the function of agents getting to their next action is shown in algorithm 1. In which from step 2 to step 12, $Q(s, a)$ is continuously changing until all $Q(s, a)$ becomes optimal.

5 Working of our algorithm

This section shows the sequences of our algorithm to generate coalitions and execute the chase which is inspired by the strategies and functions based on previous sections. This algorithm flowchart is shown in Fig. 2 which is interpreted as follows: the organizer evaluates the parameters including position and reward of each evader found in the environment by all pursuers and puts them in list of evaders; then, it broadcasts the message about each evader agent to all of the pursuers and waits for the response from pursuers. The pursuers send the response to organizer after getting enquiry. Next, the organizer calculates the membership degree of each pursuer for each evader. The evaders with same output from the SOFM which is fed by their AGRMF vectors are divided into the same group; pursuer group stores the list of pursuers which are responsible for each evader sequentially. Then, the organizer will append all the pursuers who are ready to catch the evaders of the same group to the same coalition. Each alliance has a limited lifetime called life to avoid infinite catching situation and produce chances for reorganization, for each pursuer to follow the optimized strategy.
they calculate according to the immediate payoff matrix during that period. All of the pursuers’ capability will result as a reward if one evader of the same group is caught. On the contrary, they will be punished for the rest of the evaders after lifetime. This whole scenario will continue until all evaders are captured.

6 Simulation experiments

We did experiments to show and reveal that how important is clustering similar evaders. We also compare the performance of our algorithm with other famous techniques. We implemented and experimented using grid environment which is widely used by scientific community to test and prove the effectiveness of the multi-agent systems [1, 2, 11]. Real world can easily be mapped into grid environment by using small units. In our case, we use the grid environment to carry out these simulations in a rectangular field of 100 by 100 grids, in which there are 33 pursuers and nine evaders whose difficulty varies between 2 and 4. The action set of each agent is defined as \( A = \{a_{\text{up}}, a_{\text{down}}, a_{\text{right}}, a_{\text{left}}\} \). All of the agents have the same speed (one grid for each iteration). Results show that the pursuers of AGRMF model inspired by SOFM prefer to get together and surround evaders which are close to each other than those of AGRMF model without SOFM.

We did experiments on two perspectives; first, two cases show how formation of pursuers is done but ignores evader clustering, and last three showcase how the evaders are clustered together as well. Different cases and notable outcomes are given below:

- Case AGR: formation of the groups of pursuers without any mechanism integration which is totally based on the AGR organizational model and ignores evader group formation.
- Case AGRMF: formation of the groups of pursuers with the AGRMF only before any new chase iteration, thereby correcting membership function during the first chase iteration and ignoring evader group formation.
A novel approach for multi-agent cooperative pursuit to…

Fig. 3 Average capturing time

- Case SOFM + AGRMF: formation of the groups of pursuers with our coalition algorithm based on AGRMF to cluster evaders as well.
- Case KMEAN + AGRMF: formation of the groups of pursuers with the application of KMEANS to cluster similar evaders with the most suitable parameters among the data set.
- Case DBSCAN + AGRMF: formation of the groups of pursuers with the application of DBSCAN to cluster similar evaders with the most suitable parameters among the data set.

The results shown in Fig. 3 are calculated after 100 experiments for each case. In the first case of AGR, the average of total chase iterations needed for capturing all evaders is 137.71. For the second case of AGRMF, capturing of evaders is completed in an average of 78.27 chase iterations and when it comes to case of SOFM, the capture of the evaders has only consumed an average of 47.64 chase iterations for capturing the evaders. On previous techniques, that is an average of 39.1% improvement over AGRMF time for capturing evaders.

We also did experiments to compare the flexibility degree which is defined as number of times the reorganization takes place once new formation is needed, in the case of AGRMF and case of SOFM + AGRMF. If at some point, one of the pursuers changes its target, its fellow group members need to reorganize and make a new coalition to effectively capture the target evader. Higher the flexibility, the better is their ability to reorganize and hence help in capturing effectively. The results are shown in Fig. 4. Our experiments show that our algorithm helps agents reorganize on average 43.54 times as compared to only 19.1 times in case of AGRMF alone which is an average almost more than double the times as a consequence reducing capturing time as well. Orange graph line shows significant improvement over AGRMF used without
SOFM shown in blue graph line. So it can reorganize more abruptly and much more times than without SOFM. Figure 5 shows the results of the development of pursuer’s reward in case of AGRMF and the case of SOFM + AGRMF. More rewards for better coalition and capturing the evader as the capture process continues and resulting in better reward graph as it approaches towards claiming all the evaders.
Comparison of clustering approaches for evaders is also shown. We compared popular KMEANS + AGRMF and DBSCAN + AGRMF approaches with ours. Results show that the algorithm inspired by SOFM performs on a par with others as shown in Fig. 6. Our algorithm SOFM + AGRMF takes an average of 47.64 iterations to capture all the evaders, whereas KMEANS takes 61.6 and the DBSCAN + AGRMF takes 69.98 iterations. Overall significant improvement of 22.66% over KMEANS. Hence, our algorithm improved capturing time and flexibility degree with the new reward function and coalition formation which clusters similar evaders as well.

7 Conclusion

In this paper, the cooperation mechanism of multi-agent pursuers based on SOFM and AGRMF is presented specifically focused on how to cluster similar evaders according to their membership function instead of just assigning group for each pursuer, besides a novel immediate reward function is proposed to increase the attraction of cell around the evaders. Results show that the pursuers of AGRMF model inspired by SOFM are more effective to get together and surround evaders which are close to each other than those of AGRMF model without SOFM. An improvement of 39.1% is achieved to capture all evaders with this new coalition formation. The main purpose of this algorithm is to improve the flexibility degree—the number of times agents adjust their coalition formation w.r.t. target—when the pursuers are in the situation of bigger environment and also more evaders in that environment. Our algorithm reorganizes almost double the times as compared to AGRMF. The simulation results show great improvement in reducing the total capturing time and improving the flexibility degree.
as compared to previous algorithms which do not cluster the evaders at all. Also it shows that it is adaptive to the change of environment.

Our algorithms work very well when evaders are randomly placed in an environment but are not much effective when evaders are too independent, for example are at corners of the environment, and will take same time as AGRMF. In the future, we would like to tackle this problem. Other aspects are to consider camera on agent as input with limited range of view and variable speed of agents in pursuit evasion.

References

1. Ames B et al (2015) A leapfrog strategy for pursuit-evasion in a polygonal environment. IJCGA 25(02):77–100. https://doi.org/10.1142/s0218195915500065
2. Amigoni F et al (2012) A game theoretical approach to finding optimal strategies for pursuit evasion in grid environments. IEEE (ICRA). https://doi.org/10.1109/icra.2012.6224924
3. Dorf RC et al (1990) Concise international encyclopedia of robotics: applications and automation. Wiley, New York, pp 50–59
4. Durham JW, Franchi A, Bullo F (2012) Distributed pursuit-evasion without mapping or global localization via local frontiers. Auton Robot 32:81–95
5. Fang B et al (2017) Personality driven task allocation for emotional robot team. Int J Mach Learn Cybern. https://doi.org/10.1007/s13042-017-0679-3
6. Florian R et al (2012) Muroco: a framework for capability—and situation-aware coalition formation in cooperative MRS. J Intell Robot Syst 67:339–370
7. Ghazikhani A et al (2010) A novel algorithm for coalition formation in multi-agent systems using cooperative game theory. In: 18th (ICEE’10), Isfahan, Iran, pp 512–516
8. Hespanha JP, Prandini M, Sastry S (2000) Probabilistic pursuit-evasion games: a one-step Nash approach. In: Proceedings of the 39th IEEE Conference on Decision and Control, vol 3, pp 2272–2277, Sydney, Australia
9. Kuo JY et al (2015) Multiagent cooperative learning strategies for pursuit-evasion games. Math Probl Eng, Article ID 964871
10. Li J et al (2009) Multi-robot cooperative pursuit based on association rule data mining. In: 6th Conference on FSKD. 7(4):303–308. https://doi.org/10.1109/fskd.2009.403
11. Soudi ME, Piao S (2016) A new decentralized approach of multiagent cooperative pursuit based on the iterated elimination of dominated strategies model. Math Probl Eng 2016:1–11. https://doi.org/10.1155/2016/5192423
12. Soudi ME, Songhao P et al (2015) Multi-agent cooperation pursuit based on an extension of AALAADIN organisational model. J Exp Theor Artif Intell 28(6):1075–1088. https://doi.org/10.1080/0952813x.2015.1056241
13. Stephanakis IM et al (2015) Anomaly detection in secure cloud environments using a SOFM model for clustering sets. In: 16th (INNS). https://doi.org/10.1145/2797143.2797145
14. Stiﬄer NM, Okane JM (2014) A complete algorithm for visibility-based pursuit-evasion with multiple pursuers. In: 2014 IEEE International Conference on Robotics and Automation (ICRA). https://doi.org/10.1109/icra.2014.6907074
15. Vesanto J, Alhoniemi E (2000) Clustering of the self-organizing map. IEEE Trans Neural Netw 11(3):586–600. https://doi.org/10.1109/72.846731
16. Wang C et al (2013) A new approach of multi-robot cooperative pursuit. In: Proceedings of the 32nd Chinese Control Conference, Xi’an; ISBN: 978-9-8815-6383-5
17. Wang H et al (2014) An alliance generation algo based on modified particle swarm optimization for multiple emotional robots pursuit-evader problem. 11th (FSKD)
18. Ze-Su C et al (2008) Multi-robot cooperation pursuit based on combinatorial auction mechanism under dynamic environment. In: 2nd Conference on (ISSCAA), pp 1–6