

Research Article

A New Decentralized Approach of Multiagent Cooperative Pursuit Based on the Iterated Elimination of Dominated Strategies Model

Mohammed El Habib Souidi^{1,2} and Songhao Piao¹

¹Harbin Institute of Technology, Computer Science and Technology, Harbin 150001, China

²Department of Computer Science, University of Khenchela, 40000 Khenchela, Algeria

Correspondence should be addressed to Mohammed El Habib Souidi; mohamedsouidi1989@hotmail.com

Received 20 June 2016; Revised 7 September 2016; Accepted 25 September 2016

Academic Editor: Vladimir Turetsky

Copyright © 2016 M. E. H. Souidi and S. Piao. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Game Theory is a promising approach to acquire coalition formations in multiagent systems. This paper is focused on the importance of the distributed computation and the dynamic formation and reformation of pursuit groups in pursuit-evasion problems. In order to address this task, we propose a decentralized coalition formation algorithm based on the Iterated Elimination of Dominated Strategies (IEDS). This Game Theory process is common to solve problems requiring the withdrawal of dominated strategies iteratively. Furthermore, we have used the Markov Decision Process (MDP) principles to control the motion strategy of the agents in the environment. The simulation results demonstrate the feasibility and the validity of the given approach in comparison with different decentralized methods.

1. Introduction

Multiagent System (MAS) is the cooperation of an organized set of intelligent agents situated in an environment in order to coordinate their performances and resolve complex problems. Coalition formation, considered as a major focus in social systems, is an important method of cooperation. In general, coalition formation between the agents is goal-directed and short-lived. Coalitions are formed to achieve a specific objective and dissolve when it is accomplished. Coalition formation has received a considerable amount of attention in recent research [1, 2]. Further, some research activities are based on the notion of Organization that allows the coalition of the agents in the form of groups as well as the cooperation between their members. Consequently, Ferber et al. [3] proposed AALAADIN organizational model based on three principal axes: Agent, Group, and the Role used simultaneously to describe the concrete agents' organizations (AGR).

Multiagent cooperative pursuit is a known multiagent problem [4, 5]. Based on identical conditions between

pursuers and evaders, the pursuit-evasion problem can be classified into single object pursuit and multiobject pursuit. Also, this problem has been considered in many references and its applications were inspired by an equally diverse set of approaches and useful techniques, such as Organization [6], in which we have used the principles of AGR organizational model to propose a pursuit coalition formation algorithm. Also, in order to equip each pursuit group with a dynamic access mechanism, we have introduced a flexible organizational model extended from AGR through the application of fuzzy logic principles that determines the membership degree of each pursuer in relation to each group [7].

Furthermore, Cai et al. introduced an economical auction mechanism (MPMEGBTBA) [8], where an advanced task negotiation process based on Task Bundle Auction was proposed in order to allocate the task dynamically through dynamic coalition formation of multiple agents. Also, we can find several researches treating the pursuit problem through different principles as Graph Theory [9], Polygonal Environment [10, 11], Data Mining [12], and Reinforcement Learning [13]. In this kind of problem, the pursuers and

evaders are presented in different kinds of types. The type of a pursuer denotes its pursuit capacity. Otherwise, the type of an evader reflects the number and type of pursuers required to perform its capture. The value of an evader indicates the expected rewards that should be returned to the relevant pursuers after the achievement of the capture.

Game Theory can be considered as the simplest way to model situations of conflict and studies the interactions between interested agents. The classic question relating game theory to multiagent systems is “what is the best action that an agent can perform?” This principle has been widely used in multiagent pursuit problems [14–17]. The negotiation based on Game Theory is focused on the value and rewards of each agent, which appropriately reflect the objective of agent’s negotiation (satisfying the goal of each agent). The main advantage provided by Game Theory algorithms in MAS is the coordination appearing through the hypothesis of mutual rationality of the agents. Therefore, The Game Theory algorithms are used to coordinate autonomous rational agents without the use of coordination mechanism explicitly integrated in the model of agents. Also, they provide different methods that define the optimal agents’ coalitions in several types of problems. Otherwise, the disadvantages of these algorithms concern the agents which are frequently considered as perfect rationales. Moreover, Game Theory algorithms are focused on the value of the optimal solution and overlook the most efficient method to achieve it.

In this paper we have focused on the Iterated Elimination of Dominated Strategies (IEDS), Game Theory technique to propose coalition formation algorithm as part of the pursuit-evasion problems. A strategy is the complete specification of the agent’s behavior in any situation (in the case of an extensive form game it means what behavior the agent must undertake according to the set of information provided). Moreover, we have used Markov Decision Process (MDP) principles in order to control the motion strategy of each agent. This process (MDP) provides formalism of model and resolves planning and learning problems under uncertainty [18, 19].

The paper is organized as follows: in Section 2, we discuss the main related works based on the same principles used in this paper. In Section 3, we focus on the pursuit-evasion problem by submitting a detailed explanation of the simulation environment and its different contents. Also, we approach the environmental agents by the definition and the clarification of the principal characteristics of pursuers and evaders. In Section 4, the Iterated Elimination of Dominated Strategies principle is described and detailed by an application’s example of this Game Theory process. In Section 5, the basic principles of Markov Decision Process are motivated. Therefore, we clarify the principles of the primary functions better known as the reward and transition functions. In Section 6, we introduce the distributed coalition formation algorithm with a detailed clarification of the coalition progress. A simulation of the pursuit-evasion game example is shown in Section 7; in this part we describe our simulation environment in a specific manner. Also, we present the results achieved in comparison with other outcomes based on other theories. Finally, Section 8 contains concluding remarks about this paper.

2. Related Work

There exist a lot of works based on game theoretic principles regarding the PE problem such as in [14] where the author described the control of an autonomous agents’ team tracking an intelligent evader in a nonaccurately mapped terrain based on a method to calculate the Nash equilibrium policies by resolving an equivalent zero-sum matrix game. In this example, among all Nash equilibrium, the evader selects the one which optimizes its deterministic distance to the pursuers’ team. In order to resolve the problems often encountered in the algorithms of pursuit-evasion games such as computational complexity and the lack of universality, Dong et al. [15] propose a hybrid algorithm founded on improved dynamic artificial potential field and differential game, where Nash equilibrium solution is optimal for both pursuer and evader in barrier-free zone in pursuit-evasion game, and in accordance with environment changes around the pursuit elements the algorithm is applied with flexibility. Moreover in [16], Amigoni and Basilico have presented an approach to calculate the optimal pursuer’s strategy that maximizes the probability of the target’s capture in a given environment. This approach is based on the definition of the pursuit-evasion game theoretic model as well as on its resolution through the mathematical programming.

More recently, Lin et al. [20] proposed a pursuit-evasion differential game based on Nash strategies involving limited observations. On the one hand, the evader undertakes the standard feedback Nash strategy. On the other hand, the pursuers undertake Nash strategies based on the novel concept of best achievable performance indices proposed. This model has potential applications in cases where several weakly equipped pursuing vehicles are tracking well-equipped unmanned vehicle.

In relation to PE, MDP is usually used to provide the motion planning for the mobile pursuers through the maximization of the rewards obtained during the pursuit. In [21], a Partially Observable Markov Decision Process (POMDP) algorithm is used to search the mobile target in a known graph. The main objective is to ensure the capture of the targets via the clearing of the graph in minimal time. In [22], the authors propose a new approach Continuous Time Markov Decision Process (CTMDP) to address the PE problem. In relation to MDP, CTMDP takes into account the impact of the transition time between the states involving a strong robustness against changes in transition probability. In [23] the authors proposed an innovative approach totally based on MDP with the aim of resolving sequential multiagent decision problems by allowing agents to reason explicitly about specific coordination mechanisms. In other words, they determined a value iteration algorithm to compute the optimal policies that recognizes and reasons about Coordination Problems.

Furthermore, we can consider other works based on MDP and IEDS such as [24], in which an exact dynamic programming algorithm for partially observable stochastic games (POSGs) is developed. Also, it is proven that the algorithm iteratively eliminates very weakly dominated strategies without first forming a normal form representation of the

game when it is applied to finite-horizon POSGs. Otherwise, several types of coordination mechanisms are currently used such as Stochastic Clustering Auctions (SCAs) [25, 26] which represent a class of cooperative auction methods based on the modified Swendsen-Wang method. It permits each robot to reconstitute the tasks that have been linked and applies to heterogeneous teams. Other mechanisms are market-based as TraderBots [27] applied on greedy agents in order to provide a detailed analysis of the requirements for robust and efficient multirobot coordination in dynamic environments. From the point of view of Game Theory, some research activities [28] investigated the optimal coordination approach for multiagent foraging, indeed, they built the equivalence between the optimal solution of MAS and the equilibrium of the game according to the same case, and then they introduced evolutionarily stable strategy to be of service in resolving the equilibrium selection problem of traditional Game Theory.

3. Problem Description

In this section, we focus on the cooperation problem in which n pursuers situated in a liminary toroidal grid environment X have to capture m evaders of different types. The expressions of $P = \{P_1, \dots, P_n\}$ and $E = \{E_1, \dots, E_m\}$ represent the collection of n pursuers and m evaders, respectively. Pursuers and evaders represent the roles that the agents can play. Each evader is characterized by a type Re , with $Re \in \{I, II, III, IV\}$ to indicate how many pursuers are required to capture it. Here we suppose that the pursuers can evaluate the evaders' types after the localization. There exist some fixed obstacles with different shapes and sizes in the environment X . The position could be destined for the mapping mp :

$$X \rightarrow \{0, 1\}, \text{ such as } \forall x \in X: mp(x) = 1, \text{ well then } x \text{ is an obstacle.}$$

In our proposal, the strategies of each pursuer are guided by determining factors that reflect the individual development of the pursuer during the execution of the assigned tasks. These factors are detailed as follows.

Self-Confidence Degree. In multiagent systems, each agent must be able to execute the services requested by the other agents. The self-confidence degree is the assessment of the agent's success in relation to the assigned tasks. It is denoted and computed in the following way:

$$\forall \text{Conf} \in [0.1, 1] : \text{Conf} = \max\left(0.1, \frac{C_s}{C_t}\right). \quad (1)$$

C_s is the number of tasks that the agent has accomplished. C_t is the number of tasks in which the agent has participated.

The Credit. In the case where the agent cannot perform a task, then its credit will be affected. The credit of an agent is designated and calculated as follows:

$$\forall \text{Credit} \in [0, 1] : \text{Credit} = \min\left(1, 1 - \frac{C_b}{C_t - C_s}\right). \quad (2)$$

C_b is the number of the abandoned tasks by the agent.

Environment Position. The position of the agent in the environment is a crucial criterion for the pursuit sequences, because the capture will be easier if the pursuer is closer to the evader. The position Pos is computed as follows:

$$\text{Pos} = \text{Dist}(S_p, S_E). \quad (3)$$

S_p is the state (cell) of the pursuer. S_E is the state (cell) of the evader. Dist is the distance between the pursuer and the evader.

$$\text{Dist}(S_p, S_E) = \sqrt{(CC_{P_i} - CC_{E_i})^2 + (CC_{P_j} - CC_{E_j})^2}. \quad (4)$$

(CC_{P_i}, CC_{P_j}) is the Cartesian coordinates of the pursuer. (CC_{E_i}, CC_{E_j}) is the Cartesian coordinates of the evader.

In order to distinguish the different coalitions, each pursuer belonging to a coalition calculates the value returned to itself through this strategy. This computation is based on the factors characterizing the pursuers. For example, a pursuer (P_1) belongs to the coalition (Co); the value of this coalition in relation to this pursuer is calculated as follows:

$$\begin{aligned} \text{Co}_{(\text{val}, P_1)} &= \frac{\text{Coef}_1 \times \text{Conf}_1 + \text{Coef}_2 \times \text{Credit}_1 + \text{Coef}_3 \times \text{Pos}_1}{\sum_{k=1}^3 \text{Coef}_k} \\ &+ \sum_{i=2}^{\text{Re}} \frac{\text{Coef}_1 \times \text{Conf}_i + \text{Coef}_2 \times \text{Credit}_i + \text{Coef}_3 \times \text{Pos}_i}{\text{Re} \times \sum_{k=1}^3 \text{Coef}_k}. \end{aligned} \quad (5)$$

Coef is coefficient of each factor.

On the basis of these values and using IEDS method, our mechanism will be able to select the optimal pursuit coalition for each evader detected as detailed in Section 6.

4. The Iterated Elimination of Dominated Strategies (IEDS)

The coalition is a set of pursuers required to capture the evaders detected. In the coalition each pursuer must correspond to a specific strategy. In our proposal, a pure strategy s_i defines a specific pursuit group's integration that the pursuer will follow in every possible and attainable situation during the pursuit. Such coalitions may not be random or drawn from a distribution, as in the case of mixed strategies. A strategy str_i dominates another strategy str'_i if and only if, for every potential combination of the other players' actions str_{-i} ,

$$\mu_i(\text{str}_i, \text{str}_{-i}) \geq \mu_i(\text{str}'_i, \text{str}_{-i}). \quad (6)$$

μ is function that returns the results obtained through the application of a specific strategy.

Consider the strategic game shown in Table 1, where the column player has three pure strategies and the row player has only two (a). Knowing that, the values shown in each case represent the expected payoffs returned to the players in the case of selecting the current strategy. Playing the Center is always better than playing the Right side for the column

TABLE I: Application of IEDS technique.

(a)			
	Left	Center	Right
Up	5, 4	3, 8	1, 5
Down	6, 6	6, 0	-3, -1
(b)			
	Left	Center	Right
Up	5, 4	3, 8	1, 5
Down	6, 6	6, 0	-3, -1
(c)			
	Left	Center	Right
Up	5, 4	3, 8	1, 5
Down	6, 6	6, 0	-3, -1
(d)			
	Left	Center	Right
Up	5, 4	3, 8	1, 5
Down	6, 6	6, 0	-3, -1

Bold fonts reflect how the dominated strategies are deleted.

player. Consequently, we can assume he will eventually stop playing Right side because it is a dominated strategy (b). So, we can ignore the Right side column after its elimination. Now row player has a dominated strategy, UP. Eventually row player stops playing UP; then, row-UP gets eliminated (c). Finally, we have two remaining choices Down-Left and Down-Center and column player notices that it can only win by playing Left (d). So, we can deduce that the IEDS solution is (Down, Left) with the following payoff: (6, 6).

5. Markov Decision Process Principles

Markov Decision Processes (MDPs) provide a mathematical framework to model decision making in situations where outcomes are somewhat random and partially under the control of a decision maker. In cooperative multiagent systems, MDP allows the formalization of sequential decision problems. This process only models the cooperative systems in which the reward function is shared by all players. MDP is defined by $\langle N, S, A, T, R \rangle$ as follows:

N is the number of agents Ag_i in the system, $i \in \{1, \dots, N\}$.

S corresponds to the set of agents' states s .

A : $A = A_1 \times A_2 \times \dots \times A_N$ defines the set of joint actions of the agents, where A_i is the set of local actions of the agent Ag_i .

T is the transition function. It returns the probability $T(s, a, s')$ meaning that the agent goes into the state s' if it runs the joint action $a \in A$ from state s .

R defines the reward function. $R(s, a, s')$ represents the reward obtained by the agent when it transits from

the state s to the state s' by the execution of the action a .

5.1. Reward Function. In MDP problem, the next states selected are the states returning maximum definitive reward. In our proposal, we have used Heuristic functions in order to calculate the immediate reward of each state. The reward function defines the goals that the pursuers have to achieve and identifies the environmental obstacles. To calculate this function, we relied on the agents' environment position detailed in Section 3, which allows the distribution of the rewards on the environmental cells fairly. The calculation of the rewards in each state s concerned is effectuated as follows:

$$R(s, a) = \begin{cases} \gamma & \text{if } E_i \subseteq s \\ 0 & \text{mp}(x) = 1 \\ \gamma - \text{Val}(\text{Dist}(CC_P, CC_E)) & \text{else.} \end{cases} \quad (7)$$

γ is the maximum reward. $\text{Val}(\text{Dist}(CC_P, CC_E))$ represents the distance value.

Regarding the distribution of the rewards in the standard cells, we note that the reward function is inversely proportional to the distance function.

Figure 1 illustrates a part of our simulation environment detailed in Section 7. The values displayed in the different cells $[V_1, V_2, V_3]$ represent the gains generated by the reward function. The dynamic rewards will be awarded to any pursuer situated in the cell concerned during the pursuit.

V_1 : the reward could be obtained if the pursuer concerned tracks the first evader.

V_2 : the reward could be obtained if the pursuer concerned tracks the second evader.

V_3 is the index of the cell (occupied or free).

5.2. Transition Function. The transition probabilities (ρ) describe the dynamism of the environment. They play the role of the next-state function in a problem-solving search, knowing that every state could be the possible next state according to the action undertaken in the actual state. Our approach is developed in grid of cells environment where each agent can move in four different states: s_{up} , s_{down} , s_{left} and s_{right} .

The transition probabilities of the pursuers are based on the reward degree as shown:

$$\begin{aligned} \sum_{s'} \rho(s' | s, a) &= 1, \\ \rho(s' | s, a) &= \frac{R(s', a)}{\gamma}, \\ \rho(s' | s, a) &= \max(\rho(s | s, a), \rho(s_{\text{up}} | s, a), \\ &\quad \rho(s_{\text{down}} | s, a), \rho(s_{\text{right}} | s, a), \rho(s_{\text{left}} | s, a)), \\ &\quad \forall s, a. \end{aligned} \quad (8)$$

								
[41 97 0]	[40 98 1]	[39 97 0]	[38 96 0]	[37 95 0]	[36 94 0]	[35 93 0]	[34 92 0]	[33 91 1]
[42 96 0]	[41 97 0]	[40 96 0]	[39 95 0]	[38 94 0]	[37 93 0]	[36 92 0]	[35 91 0]	[34 90 0]
[43 95 0]	[42 96 0]	[41 95 0]		[39 93 0]	[38 92 0]		[36 90 0]	[35 89 0]
[44 94 0]	[43 95 0]		[41 93 0]	[40 92 0]	[39 91 0]	[38 90 0]	[37 89 0]	[36 88 0]

FIGURE 1: Reward function applied to the grid environment. Cells with a red frame: the selected states; blue agents: pursuers; green agents: evaders; black cells: cells containing obstacles.

The linkages between the evader and each pursuer shown in Figure 2 reflect the optimal trajectories provided by the application of the method proposed in this section during each different pursuit step.

6. Coalition Formation Algorithm Based on IEDS

A number of coalition formation algorithms have been developed to define which of the potential coalitions should actually be formed. To do so, they typically compute a value for each coalition, known as the coalition value, which provides an indication of the expected results that could be derived if this coalition is constituted. Then, having calculated all the coalitional values, the decision about the optimal coalition to form can be selected. We employ an iterative algorithm in order to determine the optimal coalitions of agents. It begins with a complete set of coalitions (agent-strategy combinations) and iteratively eliminates the coalitions that have lower contribution values to MAS efficiency. The pseudocode of our algorithm is shown in Algorithm 1.

First, the algorithm calculates all the possible coalitions (Nbr_{cl}) that the pursuers can form before their filtration as needed. The expected number of the possible coalitions to form is calculated according to the following:

$$\begin{aligned}
 Nbr_{cl} &= \frac{n!}{(n - Re_1)!Re_1!} \times \frac{n - Re_1!}{(n - (Re_1 + Re_2))!Re_2!} \\
 &\times \cdots \times \frac{n - (Re_1 + \cdots + Re_{N-1})!}{(n - (Re_1 + Re_2 + \cdots + Re_N))!Re_N!} \quad (9) \\
 &= \prod_{j=1}^N \frac{(n - \sum_{k=0}^{k=j-1} Re_k)!}{(n - \sum_{k=0}^{k=j} Re_k)!Re_j!}
 \end{aligned}$$

n is the number of pursuers in the environment. N is the number of evaders detected. $Re_0 = 0$.

In order to distribute the calculation of the possible coalitions among the pursuers, the possible general coalitions

```

n: The number of pursuers
i = 0
k = 0
j = indicator of the chase iteration.
Calculate the possible coalitions;
While ( $C_{life} > 0$ ) do
    Calculate the value of each coalition;
    While (number of coalitions > 1) do
        Eliminate the dominated strategy of  $P_i$ ;
         $i \leftarrow i \bmod n + 1$ ;
    end while
    Assign the pursuers' roles according to the
    Selected coalition;
    Chase iteration;
End while
If (capture = true) then
    While ( $k \leq n$ )
        Update ( $Reward_{P_k}$ )
         $K++$ ;
    end while
Else
    The guilty pursuers pay some fines;
end if
    
```

ALGORITHM 1

(Ω) will be calculated. A general coalition enrolls all the pursuers required to capture the set of evaders detected.

$$\Omega = \frac{n!}{(n - \lambda)! \lambda!} \quad (10)$$

$\lambda = (Re_1 + Re_2 + \cdots + Re_N)$.

The general coalitions generated will be equitably distributed among the agents playing the role Pursuer. Specifically, each general coalition will be composed of N pursuit groups. From each general coalition generated through

[53 88 0]	[52 89 0]	[51 90 0]	[50 91 0]	[49 92 0]	[48 93 0]	[47 94 0]	[46 95 0]	[45 94 0]	[44 93 0]	[43 92 0]	[42 91 1]	[41 90 0]	[40 89 0]	[39 88 0]
[54 91 1]		[52 91 0]	[51 92 0]	[50 93 0]	[49 94 0]	[48 95 0]	[47 96 0]	[46 95 0]	[45 94 0]	[44 93 0]		[42 91 0]	[41 90 0]	
[55 90 0]	[54 91 0]	[53 92 0]	[52 93 0]	[51 94 0]	[50 95 0]	[49 96 0]	[48 97 0]	[47 96 0]	[46 95 0]	[45 94 0]	[44 93 0]	[43 92 0]	[42 91 0]	[41 90 0]
[56 91 0]	[55 92 0]	[54 93 0]	[53 94 0]	[52 95 0]	[51 96 0]	[50 97 0]	[49 98 1]	[48 97 0]	[47 96 0]	[46 95 0]	[45 94 0]	[44 93 0]	[43 92 0]	[42 91 0]
[57 90 0]	[56 91 0]	[55 92 0]	[54 93 0]	[53 94 0]	[52 95 0]	[51 96 0]	[50 97 0]	[49 96 0]	[48 95 0]	[47 94 0]	[46 93 0]	[45 92 0]	[44 91 0]	[43 90 0]
[58 89 0]	[57 90 0]	[56 91 0]	[55 92 0]	[54 93 0]	[53 94 0]	[52 95 0]	[51 96 0]	[50 95 0]	[49 94 0]	[48 93 0]	[47 92 0]	[46 91 0]	[45 90 0]	[44 89 0]
[59 88 0]		[57 90 0]	[56 91 0]	[55 92 0]	[54 93 0]		[52 95 0]	[51 94 0]	[50 93 0]	[49 92 0]		[47 90 0]	[46 89 0]	[45 88 0]
[60 87 0]	[59 88 0]	[58 89 0]	[57 90 0]	[56 91 0]	[55 92 0]	[54 93 0]	[53 94 0]	[52 93 0]	[51 92 0]	[50 91 0]	[49 90 0]	[48 89 0]	[47 88 0]	[46 87 0]
[61 86 0]	[60 87 0]	[59 88 0]	[58 89 0]	[57 90 0]	[56 91 0]	[55 92 0]	[54 93 0]	[53 92 0]	[52 91 0]	[51 90 0]	[50 89 0]	[49 88 0]	[48 87 0]	[47 86 0]
[62 85 0]	[61 86 0]	[60 87 0]	[59 88 0]	[58 89 0]	[57 90 0]		[55 92 0]	[54 91 0]	[53 90 0]	[52 89 0]	[51 88 0]	[50 87 0]	[49 86 0]	
[63 84 0]	[62 85 0]	[61 86 0]	[60 87 0]	[59 88 0]	[58 89 0]	[57 90 1]	[56 91 0]	[55 90 0]	[54 89 0]	[53 88 0]	[52 87 0]	[51 86 0]	[50 85 0]	[49 84 1]

FIGURE 2: Pursuers' behaviors prediction after the transition function application.

precedent calculation equation (10), a number of possible coalition formations (\mathcal{U}) will be computed:

$$\begin{aligned} \mathcal{U} &= \frac{\lambda!}{(\lambda - \text{Re}_1)! \text{Re}_1!} \times \frac{\lambda - \text{Re}_1!}{(\lambda - (\text{Re}_1 + \text{Re}_2))! \text{Re}_2!} \\ &\quad \times \cdots \times \frac{\lambda - (\text{Re}_1 + \cdots + \text{Re}_{N-1})!}{(\lambda - (\text{Re}_1 + \text{Re}_2 + \cdots + \text{Re}_N))! \text{Re}_N!} \quad (11) \\ &= \prod_{j=1}^N \frac{(\lambda - \sum_{k=0}^{j-1} \text{Re}_k)!}{(\lambda - \sum_{k=0}^j \text{Re}_k)! \text{Re}_j!}, \end{aligned}$$

$$\text{Nbr}_{\text{cl}} = \Omega \times \mathcal{U}$$

$$= \frac{n!}{(n - \lambda)! \lambda!} \times \prod_{j=1}^N \frac{(\lambda - \sum_{k=0}^{j-1} \text{Re}_k)!}{(\lambda - \sum_{k=0}^j \text{Re}_k)! \text{Re}_j!}. \quad (12)$$

This decentralized technique aims to balance the computation of the possible coalition formations among the pursuers. Furthermore, this method is more detailed in Section 7 via its application to the case study. Noting that, the value of each coalition generated in relation to each pursuer contained will be calculated according to (5). Each pursuer shares the coalitions calculated with the others to start the coalition selection

process. Secondly, we apply the Iterated Elimination of Dominated Strategies principle with the aim of finding the optimal coalition through this process. Knowing that, each strategy is represented by a possible coalition formation. Alternately, each pursuer eliminates the coalition with the lower value in relation to itself and sends the update to the next pursuer concerned. Pursuers are assigned in accordance with the selected coalition. Each pursuer performs only one chase iteration. The algorithm repeats these instructions until the end of the chase life. When $C_{\text{life}} = 0$ and the captures are accomplished, some rewards will be attributed to each one of the participating pursuers; the rewards are determined as follows:

$$\text{Rewards}_p = \frac{R(s, a)}{L}. \quad (13)$$

L is the number of the coalition's members.

Otherwise, in the case of capture failure, the guilty pursuers must pay some fines to the rest of the coalition's members. These fines are calculated as the following manner:

$$\begin{aligned} \gamma &= (s_0, a_1, s_1, a_2, s_2, \dots, s_h, a_h), \\ \text{Fines} &= \sum_{i=w}^{h-1} R(s_i, a_{i+1}). \quad (14) \end{aligned}$$

TABLE 2: The distribution of the possible coalitions' computation.

Pursuers	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8	P_9	P_{10}
General coalitions	5	5	5	5	5	4	4	4	4	4
Possible coalitions generated	350	350	350	350	350	280	280	280	280	280

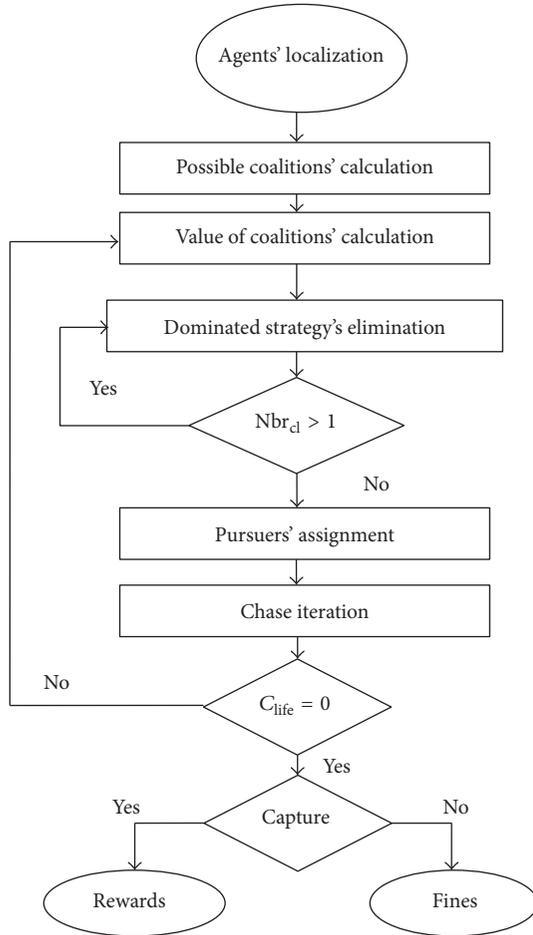


FIGURE 3: Flow chart of the algorithm.

γ is the set of states regarding the guilty pursuer. $0 \leq w \leq h$, where w represents the index of coalition's beginning.

Figure 3 reflects the flow chart of this pursuit algorithm resuming the different steps explained in this section from the detection to the capture of the existing evaders.

7. Simulation Experiments

In order to evaluate the approach presented in this paper, we realize our pursuit-evasion game on an example taking place in a rectangular two-dimensional grid with 100×100 cells. Also, we can find some obstacles characterized by the constancy and the solidity. As regards the environmental agents, our simulations are based on ten (10) pursuers and two (02) evaders of type $Re = IV$. As shown in Figure 4, it is specifically detailed how a pursuer of this type can be

captured. Each agent is marked with an ID number. Both pursuers and evaders have a similar speed (one cell per iteration) and an excellent communication system. The pursuers' teams are totally capable of determining their actual positions, and the evaders disappeared after the capture accomplishment. If the capture of the evader is performed, the coalition created to improve this pursuit will be automatically dissolved.

Table 2 resumes the results obtained after the application of the decentralized computation of the possible coalitions on this case study according to the process explained in Section 6. In this case and according to (10) the possible general coalitions (Ω) are equal to 45 coalitions, which will be distributed on the existing pursuers as shown in Table 2. From each general coalition, a number of coalitions will be generated ($\bar{U} = 70$) according to (11).

Moreover, in this case, we have studied the number of possible coalitions generated in parallel by the pursuers in relation to the number of the existing pursuers as shown in Figure 5. In relation to the centralized method in which only one pursuer computes the possible coalitions, the decentralized method decreases significantly the time concerning this computation through its division on the number of the existing pursuers.

In order to vary the types of coordination mechanisms used in our simulations, we have seen the usefulness to compare this work with our recent pursuit-evasion research activity based on AGR organizational model [6]. We have also seen the usefulness to compare our results with the results achieved after the application of an auction mechanism illustrated in Case-C- [8]. Noting that, these two methods are based on decentralized coalition formation.

Case-A- is pursuit based on (AGR) organizational model [6].

Case-B- is our new approach based on the Iterated Elimination of Dominated Strategies (IEDS) principle.

Case-C- is a pursuit based on an economical auction mechanism (MPMEGBTBA) [8].

The results shown in Figure 6 represent the average capturing time achieved during forty (40) different simulation case studies (episodes) from the beginning to the end of each one. In order to showcase the difference between the different cases, we have seen the usefulness to take into consideration the iteration concept, which determines the number of state changes regarding each agent during the pursuits.

In the first case (AGR), the average capturing time obtained equals 144.225 iterations. Furthermore, we note an interesting decrease until 100.57 iterations after the application of MPMEGBTBA due to the appropriate roles' attribution provided by this auction mechanism. However, the results that occurred through the application of IEDS

[49 94 0]	[48 95 0]	[47 96 0]	[46 95 0]	[45 94 0]	[44 93 0]		[42 91 0]
		▼					
[50 95 0]	[49 96 0]	[48 97 1]	[47 96 0]	[46 95 0]	[45 94 0]	[44 93 0]	[43 92 0]
	▶	▶	▶				
[51 96 0]	[50 97 1]	[49 98 1]	[48 97 1]	[47 96 0]	[46 95 0]	[45 94 0]	[44 93 0]
		▲					
[52 95 0]	[51 96 0]	[50 97 1]	[49 96 0]	[48 95 0]	[47 94 0]	[46 93 0]	[45 92 0]
[53 94 0]	[52 95 0]	[51 96 0]	[50 95 0]	[49 94 0]	[48 93 0]	[47 92 0]	[46 91 0]
[54 93 0]		[52 95 0]	[51 94 0]	[50 93 0]	[49 92 0]		[47 90 0]

FIGURE 4: Example: evader of the type Re equals IV after the capture.

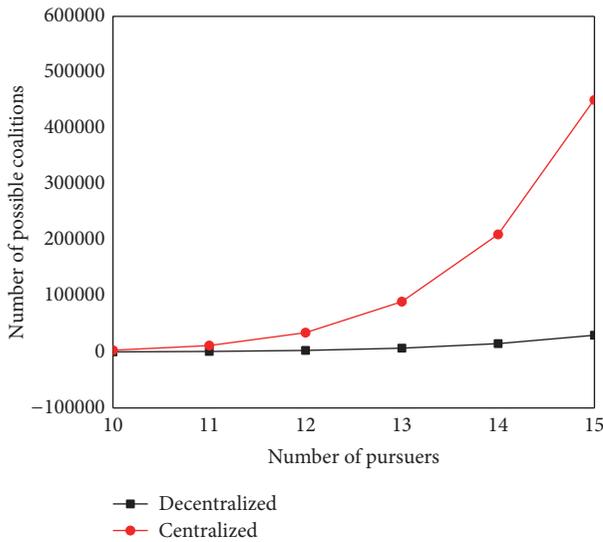


FIGURE 5: Centralized and decentralized coalitions' computation in relation to the number of pursuers.

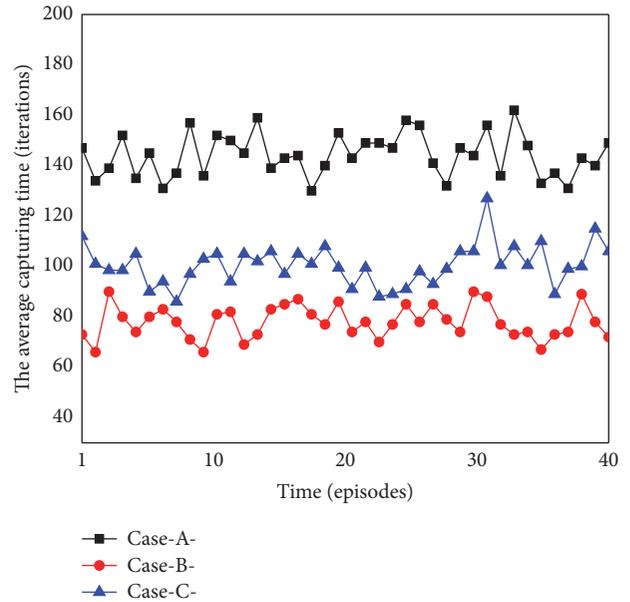


FIGURE 6: Average capturing time after (40) different pursuits.

coalition formation algorithm revealed an average capturing time of 78 iterations.

Figure 7 shows the development of the pursuers' reward function during the same pursuit period of the different cases and the outcomes reflect the improvement brought by the dynamic formations and reformations of the pursuit teams.

Finally, we have focused on the study of the average pursuers' rewards obtained in each case of chase iteration during full pursuit. In Figure 8 the x -axis represents the value of rewards achieved by a pursuer and each unit y -axis represents chase iterations. The results shown in this figure reveal a certain similarity between AGR and MPMEGBTBA,

in which the average pursuer's rewards achieved reach 0.59 and 0.507, respectively. Otherwise in IEDS, the average result increases until 0.88.

The results shown in Figure 9 represent the internal learning development (self-confidence development) of the pursuers during the pursuit applied to the three cases. The positivity of the results is due to the grouping and the equitable task sharing between the different pursuit groups imposed by the different coordination mechanisms applied. Moreover, we can note the superiority of the results obtained through IEDS in relation to the other cases provoked by the

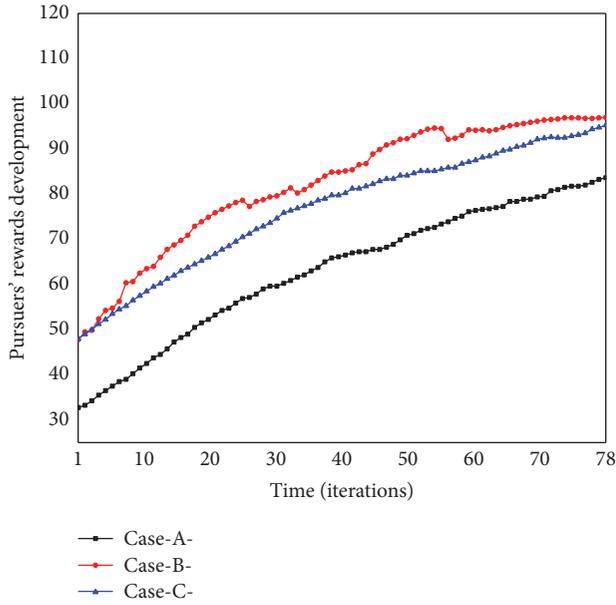


FIGURE 7: The pursuers' rewards development.

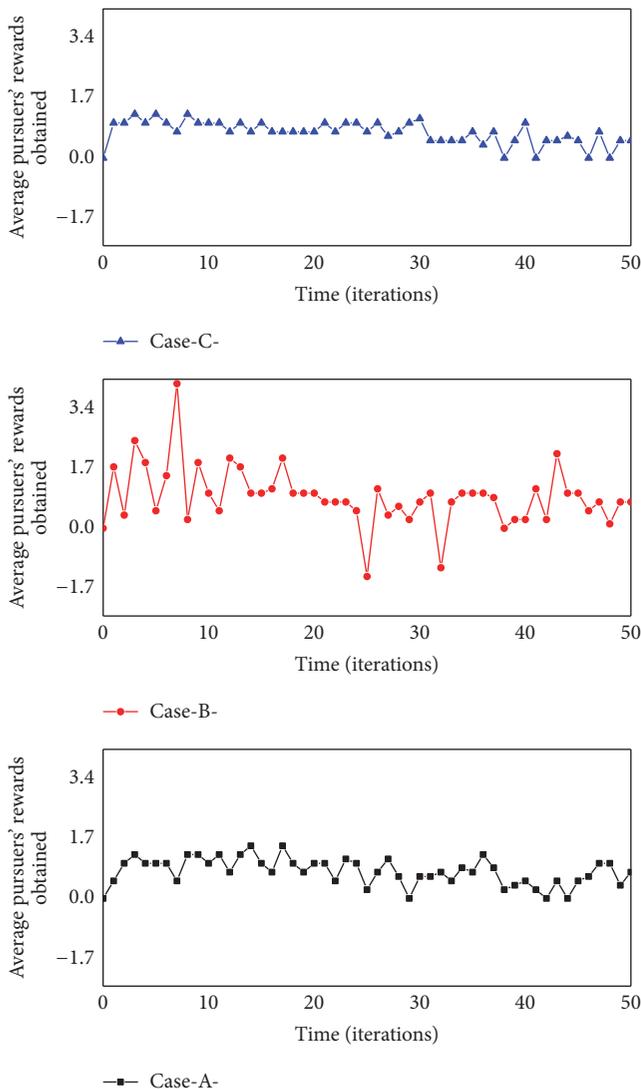


FIGURE 8: Average pursuers' reward per iteration.

TABLE 3: Pursuit result.

	AGR	IEDS	MPMEGBTBA
Average capturing time (iteration)	144.225	78	100.57
Average pursuers' rewards obtained by iteration	0.59	0.88	0.507
Average pursuers' self-confidence development	0.408	0.533	0.451

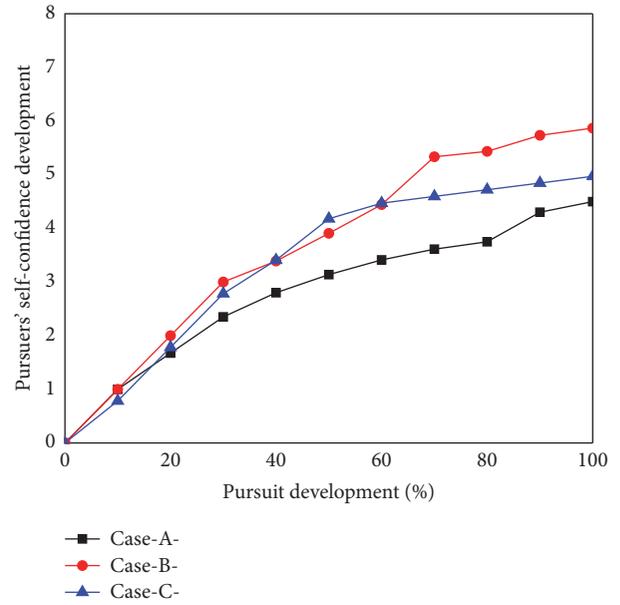


FIGURE 9: Pursuers' learning development during the pursuit.

dynamism of the coalition formations and the optimality of task sharing provided by our algorithm.

Table 3 summarizes the main results achieved; we deduce that the pursuit algorithm based on the Iterative Elimination of Dominated Strategies (IEDS) is better than the algorithm based on AGR organizational model as well as the auction mechanism based on MPMEGBTBA regarding the reward's development as well as the capturing time. The leading cause of this fact is the dynamism of our coalitional groups. This flexible mechanism improves the intelligence of the pursuers concerning the displacements and the rewards acquisition, knowing that team reward is optimal in the case where each pursuer undertakes the best path.

8. Conclusion

This paper presents a kind of a decentralized coalition method based on Game Theory principles for different types of pursuit; the proposed method demonstrates the positive impact imposed by the dynamism of the coalition formations. Firstly, we have extended our coalition algorithm from the Iterated Elimination of Dominated Strategies. This process allows us to determine the optimal pursuit coalition strategy according to the Game Theory principles. Secondly, we have focused on the Markov Decision Process as a motion

strategy of our pursuers in the environment (grid of cells). To highlight our proposal we have developed a comparative study between our algorithm and a decentralized strategy of coalition based on AGR organizational model as well as an auction mechanism based on MPMEGBTBA. Simulation results shown in this paper demonstrate that the algorithm based on IEDS is feasible and effective.

Competing Interests

The authors declare that they have no competing interests.

Acknowledgments

This paper is supported by National Natural Science Foundation of China (no. 61375081) and a special fund project of Harbin science and technology innovation talents research (no. RC2013XK010002).

References

- [1] A. Ghazikhani, H. R. Mashadi, and R. Monsefi, "A novel algorithm for coalition formation in multi-agent systems using cooperative game theory," in *Proceedings of the 18th Iranian Conference on Electrical Engineering (ICEE '10)*, pp. 512–516, Isfahan, Iran, May 2010.
- [2] L. Boongasame, "Preference coalition formation algorithm for buyer coalition," in *Proceedings of the 9th International Joint Conference on Computer Science and Software Engineering (JCSSSE '12)*, pp. 225–230, Bangkok, Thailand, May 2012.
- [3] J. Ferber, O. Gutknecht, and F. Michel, "From agents to organizations: an organizational view of multi-agent systems," in *Agent-Oriented Software Engineering IV: 4th International Workshop, AOSE 2003, Melbourne, Australia, July 15, 2003. Revised Papers*, P. Giorgini, J. Muller, and J. Odell, Eds., vol. 2935 of *Lecture Notes in Computer Science*, pp. 214–230, Springer, Berlin, Germany, 2004.
- [4] J. Y. Kuo, H.-F. Yu, K. F.-R. Liu, and F.-W. Lee, "Multiagent cooperative learning strategies for pursuit-evasion games," *Mathematical Problems in Engineering*, vol. 2015, Article ID 964871, 13 pages, 2015.
- [5] G. I. Ibragimov and M. Salimi, "Pursuit-evasion differential game with many inertial players," *Mathematical Problems in Engineering*, vol. 2009, Article ID 653723, 15 pages, 2009.
- [6] M. Souidi, S. Piao, G. Li, and L. Chang, "Coalition formation algorithm based on organization and Markov decision process for multi-player pursuit evasion," *International Journal of Multiagent and Grid Systems*, vol. 11, no. 1, pp. 1–13, 2015.
- [7] M. E.-H. Souidi, P. Songhao, L. Guo, and C. Lin, "Multi-agent cooperation pursuit based on an extension of AALAADIN organisational model," *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 28, no. 6, pp. 1075–1088, 2016.
- [8] Z.-S. Cai, L.-N. Sun, H.-B. Gao, P.-C. Zhou, S.-H. Piao, and Q.-C. Huang, "Multi-robot cooperative pursuit based on task bundle auctions," in *Intelligent Robotics and Applications*, C. Xiong, Y. Huang, Y. Xiong, and H. Liu, Eds., vol. 5314 of *Lecture Notes in Computer Science*, pp. 235–244, Springer, Berlin, Germany, 2008.
- [9] B. Goode, A. Kurdila, and M. Roan, "A graph theoretical approach toward a switched feedback controller for pursuit-evasion scenarios," in *Proceedings of the American Control Conference (ACC '11)*, pp. 4804–4809, San Francisco, Calif, USA, June 2011.
- [10] V. Isler, S. Kannan, and S. Khanna, "Randomized pursuit-evasion in a polygonal environment," *IEEE Transactions on Robotics*, vol. 21, no. 5, pp. 875–884, 2005.
- [11] J. Thunberg, P. Ögren, and X. Hu, "A Boolean Control Network approach to pursuit evasion problems in polygonal environments," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '11)*, pp. 4506–4511, May 2011.
- [12] J. Li, Q. Pan, and B. Hong, "A new approach of multi-robot cooperative pursuit based on association rule data mining," *International Journal of Advanced Robotic Systems*, vol. 7, no. 3, pp. 165–172, 2010.
- [13] J. Liu, S. Liu, H. Wu, and Y. Zhang, "A pursuit-evasion algorithm based on hierarchical reinforcement learning," in *Proceedings of the International Conference on Measuring Technology and Mechatronics Automation (ICMTMA '09)*, vol. 2, pp. 482–486, IEEE, Hunan, China, April 2009.
- [14] J. P. Hespanha, M. Prandini, and S. Sastry, "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, December 2000.
- [15] J. Dong, X. Zhang, and X. Jia, "Strategies of pursuit-evasion game based on improved potential field and differential game theory for mobile robots," in *Proceedings of the 2nd International Conference on Instrumentation, Measurement, Computer, Communication and Control (IMCCC '12)*, pp. 1452–1456, IEEE, Harbin, China, December 2012.
- [16] F. Amigoni and N. Basilico, "A game theoretical approach to finding optimal strategies for pursuit evasion in grid environments," in *Proceedings of the IEEE International Conference on Robotics and Automation River Centre*, pp. 2155–2162, Saint Paul, Minn, USA, May 2012.
- [17] R. Liu and Z.-S. Cai, "A novel approach based on Evolutionary Game Theoretic model for multi-player pursuit evasion," in *Proceedings of the International Conference on Computer, Mechatronics, Control and Electronic Engineering (CMCE '10)*, vol. 1, pp. 107–110, August 2010.
- [18] B. Khosravifar, F. Bouchet, R. Feyzi-Behnagh, R. Azevedo, and J. M. Harley, "Using intelligent multi-agent systems to model and foster self-regulated learning: a theoretically-based approach using Markov decision process," in *Proceedings of the 27th IEEE International Conference on Advanced Information Networking and Applications (AINA '13)*, pp. 413–420, IEEE, Barcelona, Spain, March 2013.
- [19] L. Ting, Z. Cheng, and Z. Weiming, "Planning for target system striking based on Markov decision process," in *Proceedings of the IEEE International Conference on Service Operations and Logistics, and Informatics (SOLI '13)*, pp. 154–159, IEEE, Dongguan, China, July 2013.
- [20] W. Lin, Z. Qu, and M. A. Simaan, "Nash strategies for pursuit-evasion differential games involving limited observations," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 51, no. 2, pp. 1347–1356, 2015.
- [21] E. Ehsan and F. Kunwar, "Probabilistic search and pursuit evasion on a graph," *Transactions on Machine Learning and Artificial Intelligence*, vol. 3, no. 3, pp. 57–65, 2015.
- [22] S. Jia, X. Wang, and L. Shen, "A continuous-time markov decision process-based method with application in a pursuit-evasion example," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 46, no. 9, pp. 1215–1225, 2016.

- [23] C. Boutilier, "Sequential optimality and coordination in multiagent systems," in *Proceedings of the 16th International Joint Conference on Artificial Intelligence (IJCAI '99)*, vol. 1, pp. 478–485, Stockholm, Sweden, August 1999.
- [24] E. A. Hansen, D. S. Bernstein, and S. Zilberstein, "Dynamic programming for partially observable stochastic games," in *Proceedings of the 19th National Conference on Artificial Intelligence*, pp. 709–715, 2004.
- [25] K. Zhang, E. G. Collins Jr., and A. Barbu, "An efficient stochastic clustering auction for heterogeneous robotic collaborative teams," *Journal of Intelligent & Robotic Systems*, vol. 72, no. 3–4, pp. 541–558, 2013.
- [26] K. Zhang, E. G. Collins Jr., and D. Shi, "Centralized and distributed task allocation in multi-robot teams via a stochastic clustering auction," *ACM Transactions on Autonomous and Adaptive Systems*, vol. 7, no. 2, article 21, 2012.
- [27] M. B. Dias and T. Sandholm, *TraderBots: a new paradigm for robust and efficient multirobot coordination in dynamic environments [Ph.D. thesis]*, The Robotics Institute, Carnegie Mellon University, Pittsburgh, Pa, USA, 2004.
- [28] Y. Wang, *Evolutionary Game Theory Based Cooperation Algorithm in Multi-Agent System*, *Multiagent Systems*, InTech, Rijeka, Croatia, 2009.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

