Frontiers of Information Technology & Electronic Engineering
www.zju.edu.cn/jzus; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Coalition formation based on a task-oriented collaborative ability vector*

Hao FANG‡1,2, Shao-lei LU1,2, Jie CHEN1,2, Wen-jie CHEN1,2

(1School of Automation, Beijing Institute of Technology, Beijing 100081, China)

(2State Key Laboratory of Intelligent Control and Decision of Complex Systems, Beijing 100081, China)

E-mail: fangh@bit.edu.cn; lu_shaolei@126.com; chenjie@bit.edu.cn; chen.wenjie@163.com

Received Oct. 9, 2016; Revision accepted Dec. 23, 2016; Crosschecked Dec. 29, 2016

Abstract: Coalition formation is an important coordination problem in multi-agent systems, and a proper description of collaborative abilities for agents is the basic and key precondition in handling this problem. In this paper, a model of task-oriented collaborative abilities is established, where five task-oriented abilities are extracted to form a collaborative ability vector. A task demand vector is also described. In addition, a method of coalition formation with stochastic mechanism is proposed to reduce excessive competitions. An artificial intelligent algorithm is proposed to compensate for the difference between the expected and actual task requirements, which could improve the cognitive capabilities of agents for human commands. Simulations show the effectiveness of the proposed model and the distributed artificial intelligent algorithm.

Key words: Collaborative vector; Task allocation; Multi-agent system; Coalition formation; Artificial intelligence
http://dx.doi.org/10.1631/FITEE.1601608 **CLC number:** TP182

1 Introduction

Coalition formation is an important coordination problem in multi-agent systems (MASs). In some situations, a single agent is unable to implement a given task or cannot implement it effectively. Agents need to coordinate with others to finish the given task, and a coalition is formed to achieve this goal.

Forming a coalition is an important way to cooperate with others and many methods have been proposed (Ketchpel, 1994; Shehory and Kraus, 1996; Sandhlom and Lesser, 1997; Sellner *et al.*, 2006). Among these methods, game theory and social reasoning are used primarily, as seen in Shehory and Kraus (1998) and Sichman *et al.* (1998), respectively. Game theory is used mainly to compare the effectiveness of the formed coalition, rather than providing an algorithm to create it (Shehory and Kraus, 1998). Social reasoning based algorithms, such as those in Sichman *et al.* (1998), An *et al.* (2007), and Auer *et al.* (2015), use the ability of an 'intelligent agent' to maintain an external description—goals, actions, plans—of other agents and form a coalition accordingly. Some methods of forming coalitions are inspired by phenomena in the nature, and can be found in Bonabeau *et al.* (1997), Haque and Egerstedt (2009), Du *et al.* (2010), and Haque *et al.* (2013).

Coalition formation has been applied in many fields, such as smart grids (Gensollen *et al.*, 2015; Ye *et al.*, 2015), search, and rescue (Zhao *et al.*, 2016). Ye *et al.* (2015) investigated the problem of dispatch of distributed energy resources in smart grids. They solved this problem via multi-agent coalition forma-

‡ Corresponding author

* Project supported by the National Natural Science Foundation of China (Nos. 61573062, 61120106010, and 61321002), the Beijing Outstanding Ph.D. Program Mentor (No. 20131000704), and the Beijing Advanced Innovation Center for Intelligent Robots and Systems (Beijing Institute of Technology)

ORCID: Hao FANG, http://orcid.org/0000-0002-9627-0325

tion with negotiation. However, many parameters were hand-tuned, so it does not suit dynamic systems. Zhao *et al.* (2016) proposed an algorithm consisting of a task inclusion phase and a task removal phase. Tasks are assigned to an agent by sorting the costs and making a consensus. Base on this, Whitbrook *et al.* (2015) made a development and extension to enhance the performance of a distributed heuristic algorithm using the novel concept of performance impact (PI).

The task requirement in task allocation is usually estimated by a human or evaluation system, and there may be a difference between the task requirement given by the human and the actual task requirement. Thus, an artificial intelligent algorithm is needed to improve the agents' cognition about the commands from human beings. Therefore, a cognitive compensation mechanism is proposed combined with a could model (Li and Du, 2014).

In this paper, first, task-oriented collaborative capabilities are established to reflect their characteristics towards the tasks. Second, a stochastic mechanism is proposed to reduce excessive competitions. Third, the concept of cognitive inertia is proposed which indicates the stable preference of the human. A cognitive compensation mechanism is proposed to reduce the difference between the estimated and the actual task requirements.

2 Model description

2.1 Graph theory

The graph theory used in this study is introduced. An undirected graph $\mathcal{G}$ is defined by a set of elements called vertices, $\mathcal{V}(\mathcal{G})$, a set of elements called edges, $\mathcal{E}(\mathcal{G})$. Associated with each edge are either one or two vertices called its ends. $N+1$ agents in a multi-agent system are regarded as the vertices $\mathcal{V}=(0,1,\cdots,N)$ of graph $\mathcal{G}$. v_i denotes agent i or vertex i in graph $\mathcal{G}$. v_{ij} denotes the direction from vertex i to vertex j. The set of neighbours of vertex i is denoted by N_i. $|N_i|$ is the cardinality of N_i, which means the degree of agent i or vertex i in graph $\mathcal{G}$.

2.2 Collaborative ability vector

Two tasks, reconnaissance mission and combat mission, are considered here and the values of abilities are tightly connected to tasks. Five abilities are given for each task: communication, reconnaissance, combat, motor, and energy, recorded as Cm, R, Ct, M, and E for short, respectively. Two task-oriented ability vectors are modeled as follows.

2.2.1 Reconnaissance mission oriented collaborative ability vector

Ability could be affected by many factors. The relationships between abilities and factors in the reconnaissance mission are shown in Fig. 1.

According to these factors, five ability models are given in the following.

Communication ability: The communication ability of an agent is affected by three factors, i.e., degree of the agent, quality of the communication, and mobility of the agent.

Degree factor: A larger degree means a stronger cooperative communication ability. The degree is set to be 7 in wireless communication. The degree factor of agent i is given as follows:

$$\mathrm{Deg}_{v_i}=|N_i|/7. \tag{1}$$

Quality factor: Received signal strength indicator (RSSI) can be used here to evaluate the quality of the communications between two agents. So, the communication quality of agent i can be modeled as follows:

$$\mathrm{Qua}_{v_i}=\frac{1}{|N_i|}\sum_{j\in N_i}s*\min\left(\mathrm{RSSI}\left(v_{ij}\right),\mathrm{RSSI}\left(v_{ji}\right)\right). \tag{2}$$

Here, s denotes sensitivity, ranging from 0 to 1, and $\mathrm{RSSI}(v_{ij})$ means the received signal strength of agent i from agent j, ranging from 0 to 1. $\mathrm{RSSI}(v_{ji})$ is defined analogously.

Mobility factor: RSSI can probably reflect the distance between two agents. Based on $\mathrm{RSSI}(v_{ij})$, the mobility definition of agent i can be given as follows:

$$\mathrm{Mob}_{v_{ij}}=\ln\frac{\mathrm{RSSI}^t_{v_{ij}}}{\mathrm{RSSI}^{t-1}_{v_{ij}}}, \tag{3}$$

where t denotes the current time specified in seconds. The mobility of an agent can be obtained by calculating the average of the relevant mobility among the agent and its neighbors, which is given by

$$\mathrm{Mob}_{v_i}=\frac{1}{|N_i|}\sum_{j\in N_i}\mathrm{Mob}_{v_{ij}}. \tag{4}$$

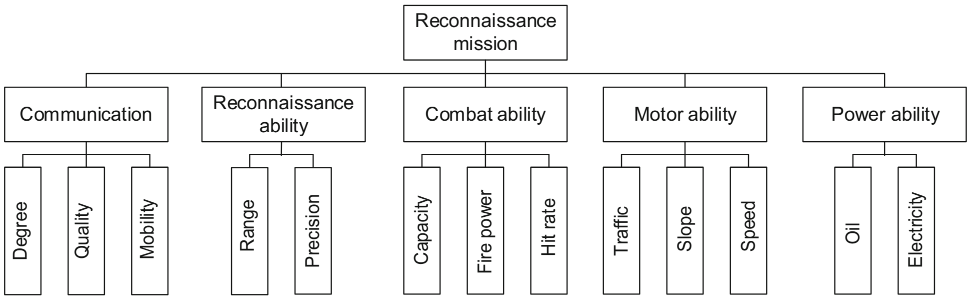

Fig. 1 Relationships between abilities and factors

Table 1 Parameters of the detectors

Device	Range	Largest search angle	Probability of finding targets	Maximum number of traced targets	Location precision	Solution
SAR	Middle	Middle	Middle	Middle	Middle	Middle, high
CCD	Short	Small	Low	Small	Middle, high	High
ERD	Long	Large	High	Large	Low	Low

Table 2 Digital parameters of the detectors

Device	Range	Largest search angle	Probability of finding targets	Maximum number of traced targets (normalized)	Location precision	Solution
SAR	0.5	0.5	0.5	0.5	0.5	0.8
CCD	0.3	0.3	0.3	0.3	0.8	1.0
ERD	1.0	1.0	1.0	1.0	0.3	0.3

Table 3 Paired comparisons

Parameter	Range	Largest search angle	Probability of finding targets	Maximum number of traced targets	Location precision	Solution
Range	1	1/3	1/5	1/3	3	3
Largest search angle	3	1	1/3	3	3	3
Probability of finding targets	5	3	1	5	5	5
Maximum number of traced targets	3	1/3	1/5	1	1	1
Location precision	1/3	1/3	1/5	1	1	1
Solution	1/3	1/3	1/5	1	1	1

Therefore, the value of the communication ability can be calculated as follows:

$$A_{\mathrm{c}} = \omega_1 \cdot \mathrm{Deg}_{v_i} + \omega_2 \cdot \mathrm{Qua}_{v_i} + \omega_3 \cdot \mathrm{Mob}_{v_i}, \quad (5)$$

where ω_1, ω_2, and ω_3 are positive and $\omega_1+\omega_2+\omega_3=1$. Based on experience, ω_1, ω_2, and ω_3 are roughly given as 0.3, 0.5, and 0.2, respectively.

Reconnaissance ability: The reconnaissance ability of the agent is affected by two factors, i.e., range of the detector and precision of the detector. Three detectors are considered: synthetic aperture radar (SAR), charge-coupled device (CCD), and electronic reconnaissance device (ERD). The parameters of these devices are listed in Table 1.

The reconnaissance ability of the agent is defined as follows:

$$A_{\mathrm{r}} = D^{\mathrm{sar}} \cdot \xi^{\mathrm{sar}} + D^{\mathrm{ccd}} \cdot \xi^{\mathrm{ccd}} + D^{\mathrm{erd}} \cdot \xi^{\mathrm{erd}}. \quad (6)$$

Here, D^{sar}, D^{ccd}, and D^{erd} denote the ability of three devices, and the sum of the three non-negative numbers is equal to 1. ξ denotes the electrical interference coefficient, indicating the degree of interference by the electronic equipment, and $\xi \in [0,1]$. To calculate the value of the reconnaissance ability, the analytic hierarchy process (AHP) method (Saaty, 1990; 2008) is applied here. First, parameters in Table 1 are transferred into the accurate numbers shown in Table 2.

Paired comparisons of the qualitative parameters are given in Table 3.

According to the AHP method, consistency checking is necessary. The largest eigenvalue $\lambda_{\max}$ of matrix $\boldsymbol{B}$ formed by Table 3 is 6.5576. The consistency index (CI) and consistency ratio (CR) are calculated as follows:

$$\mathrm{CI}=\frac{\lambda_{\max}-n}{n-1}=0.111, \tag{7}$$

$$\mathrm{CR}=\frac{\mathrm{CI}}{\mathrm{RI}}=\frac{0.111}{1.24}=0.0895<0.1. \tag{8}$$

Here, n denotes the order of the matrix, set to 6 here, and the random index (RI) is given in Table 4.

Table 4 Values of the random index (RI)

n	1	2	3	4	5	6	7	8
RI	0	0	0.58	0.90	1.12	1.24	1.32	1.41

The eigenvector, denoted by $\boldsymbol{C}$, associated with the maximum eigenvalue $\lambda_{\max}$, is normalized. The result after normalization is shown as follows:

$$\boldsymbol{C}=[0.1073, 0.2157, 0.4327, 0.1105, 0.0699, 0.0699]. \tag{9}$$

Then, the abilities of detectors, denoted by $\boldsymbol{D}$, can be obtained by calculating the product of $\boldsymbol{B}$ and $\boldsymbol{C}$. After normalization, $\boldsymbol{D}$ is shown as follows:

$$\boldsymbol{D}=[D^{\mathrm{sar}}, D^{\mathrm{ccd}}, D^{\mathrm{erd}}]=[0.3, 0.2, 0.5]. \tag{10}$$

Combat ability: The combat ability is affected by three factors, i.e., fire power, ammo capacity, and probability of a first-round hit. Then the combat ability of an agent is defined as follows:

$$A_{\mathrm{t}}=\frac{f_{\max}}{F_{\max}}\cdot f_{\mathrm{left}}\cdot\rho. \tag{11}$$

Here, $f_{\max}$ denotes the maximum fire power of the current agent, $F_{\max}$ denotes the largest one of all $f_{\max}$, f_{left} denotes the allowance for the ammunition, and ρ denotes the probability of a first-round hit.

Motor ability: The motor ability is affected by three factors, i.e., maximum speed, slope coefficient, and traffic coefficient. Motor ability is defined as follows:

$$A_{\mathrm{m}}=\frac{v_{\max}}{V_{\max}}\cdot\alpha\cdot\beta. \tag{12}$$

Here, $v_{\max}$ denotes the maximum speed of the current agent, $V_{\max}$ denotes the largest one of all $v_{\max}$, and α denotes the slope coefficient, defined by

$$\alpha=\begin{cases}1-\sqrt{3}\tan\theta, & \theta<\pi/6,\\ 0.001, & \text{else},\end{cases} \tag{13}$$

where θ denotes the slope between the agent and the ground, and it is less than $\pi/6$ here.

β in Eq. (12) denotes the traffic coefficient (Table 5).

Table 5 Values of the traffic coefficient

Landform	Road	Dirt road	Grass land	Hill land	Sand land	Wood land
Level	0	1	2	3	4	5
Coeff.	1	0.8	0.6	0.4	0.3	0.8

Coeff.: coefficient

Energy ability: The energy ability is affected by two factors, remaining oil and remaining electronic power. The energy ability is defined as follows:

$$A_{\mathrm{e}}=0.5\frac{l_{\mathrm{left}}}{L_{\max}}+0.5\frac{e_{\mathrm{left}}}{E_{\max}}. \tag{14}$$

Here, l_{left} denotes the remaining oil and $L_{\max}$ the longest distance that one agent can run. Therefore, the reconnaissance mission oriented feature vector is

$$\boldsymbol{A}=[A_{\mathrm{c}}, A_{\mathrm{r}}, A_{\mathrm{t}}, A_{\mathrm{m}}, A_{\mathrm{e}}]. \tag{15}$$

2.2.2 Combat mission oriented collaborative ability vector

The distance between the agent and the target needs to be considered in the combat mission. The relationship between features and factors in the combat mission is shown in Fig. 2.

The communication ability and motor ability are the same as those in the reconnaissance mission. Reconnaissance, combat, and energy abilities will be shown in the following part.

Reconnaissance ability: Considering the factor of distance, reconnaissance ability is defined as follows:

$$A_{\mathrm{r}}'=p^{\mathrm{sar}}\cdot\xi^{\mathrm{sar}}+p^{\mathrm{ccd}}\cdot\xi^{\mathrm{ccd}}+p^{\mathrm{erd}}\cdot\xi^{\mathrm{erd}}, \tag{16}$$

where p denotes the probability of finding targets. p^{sar} is defined as follows:

$$p^{\mathrm{sar}}=\exp\left(-\frac{d}{d_0}\right), \tag{17}$$

where d denotes the distance between the agent and the target, d_0 indicates the maximum distance over which a device can find the target, and p^{ccd} and p^{erd} are defined analogously.

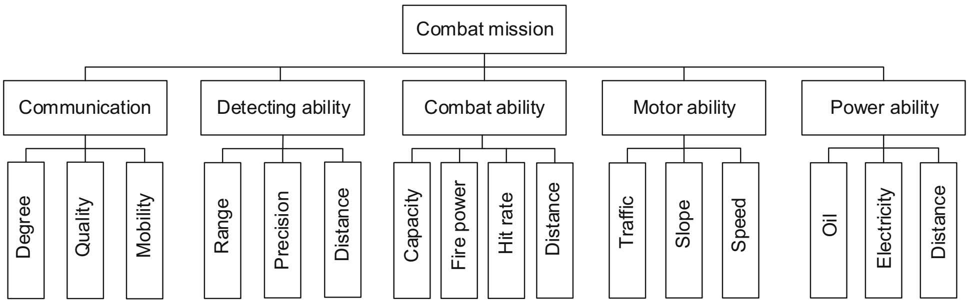

Fig. 2 Relationships between abilities and factors

Combat ability: Considering distance (Diao *et al.*, 2014), combat ability is defined as follows:

$$A'_{\mathrm{t}} = \frac{f_{\max}}{F_{\max}} \cdot f_{\mathrm{left}} \cdot \rho \cdot f_{\mathrm{dist}}, \tag{18}$$

where f_{dist} is given by

$$f_{\mathrm{dist}} = \begin{cases} 1, & d_{\mathrm{t}} < d_{\mathrm{mk}}, \\ 2^{-\frac{d_{\mathrm{t}} - d_{\mathrm{mk}}}{d_{\mathrm{m}} - d_{\mathrm{mk}}}}, & d_{\mathrm{mk}} \le d_{\mathrm{t}} \le d_{\mathrm{m}}, \\ 0.5\mathrm{e}^{-\frac{d_{\mathrm{t}} - d_{\mathrm{m}}}{d_{\mathrm{m}}}}, & d_{\mathrm{t}} > d_{\mathrm{m}}, \end{cases} \tag{19}$$

where d_{t} denotes the distance between the agent and the target, d_{m} denotes the weapon range, and d_{mk} denotes the radius of the non-escaped zone.

Energy ability: Considering distance, energy ability is defined as follows:

$$A'_{\mathrm{e}} = \frac{l_{\mathrm{left}}}{l_{\mathrm{left}} + d_{\mathrm{e}}}. \tag{20}$$

The model is introduced from Bonabeau *et al.* (1997). Here, d_{e} denotes the distance between the agent and the target. Therefore, the combat mission oriented feature vector is

$$\boldsymbol{A}' = [A'_{\mathrm{c}}, A'_{\mathrm{d}}, A'_{\mathrm{t}}, A'_{\mathrm{m}}, A'_{\mathrm{e}}]. \tag{21}$$

2.3 Task demand vector

Task demands for agent abilities vary. A task demand vector is established using AHP to represent a task. Agents form a coalition to reach the task demand, which means that the coalition is able to accomplish this task. Normalized task demand vectors for two missions, the reconnaissance mission and the combat mission, are established as follows.

Reconnaissance mission demand vector: In the reconnaissance mission, paired comparisons between abilities are as given in Table 6.

Table 6 Paired comparisons

Ability	Cm	R	Ct	M	E
Cm	1	1/5	2	1/2	1
R	5	1	3	3	2
Ct	1/2	1/3	1	1	1/2
M	2	1/3	1	1	1
E	1	1/2	2	1	1

According to the AHP algorithm, the normalized task demand vector is obtained as follows:

$$\boldsymbol{W}_{\mathrm{d}} = [0.13, 0.43, 0.11, 0.16, 0.17]. \tag{22}$$

Combat mission demand vector: In the reconnaissance mission, using the AHP method, paired comparisons between abilities are as given in Table 7.

Table 7 Paired comparisons

Ability	Cm	R	Ct	M	E
Cm	1	1	1/5	1/2	1/3
R	1	1	1/3	1	1/3
Ct	5	3	1	3	2
M	2	1	1/3	1	1/2
E	3	3	1/2	2	1

Following the same calculation as above, we can obtain the normalized demand vector:

$$\boldsymbol{W}_{\mathrm{t}} = [0.08, 0.11, 0.41, 0.14, 0.26]^{\mathrm{T}}. \tag{23}$$

Then the ability or desire to execute the two tasks will be calculated as follows:

$$\boldsymbol{P}_{\mathrm{d}} = \boldsymbol{A} \cdot \boldsymbol{W}_{\mathrm{d}}, \quad \boldsymbol{P}_{\mathrm{t}} = \boldsymbol{A}' \cdot \boldsymbol{W}_{\mathrm{t}}. \tag{24}$$

The reality task demands for two missions are denoted by $\mathbf{RW}_{\mathrm{d}} = h \cdot \boldsymbol{W}_{\mathrm{d}}$ and $\mathbf{RW}_{\mathrm{t}} = k \cdot \boldsymbol{W}_{\mathrm{t}}$. h and k are given by the task publisher. The ability of the coalition needs to reach the corresponding reality task demand, such as $\mathbf{RW}_{\mathrm{d}}$ and $\mathbf{RW}_{\mathrm{t}}$.

3 Coalition formation algorithm with stochastic mechanism

The distributed coalition formation algorithm in Lu and Fang (2016) is modified here. In Lu and Fang (2016), agent i chooses a coalition to execute a task, such as task k, based on the term 'task readiness' denoted by TR_k^i, which consists of task profit, and the term 'task fitness' denoted by TP_k^i and TF_k^i, respectively. When the coalition ability cannot reach the task demand, agents choose the other coalition. The learning automata algorithm was used in Lu and Fang (2016) to adapt the weights of TP_k^i and TF_k^i. TR_k^i is calculated in Lu and Fang (2016) as follows:

$$\mathrm{TR}_k^i = \omega_{k1}^i(t) \cdot \mathrm{TP}_k^i + \omega_{k2}^i(t) \cdot \mathrm{TF}_k^i, \qquad (25)$$

where $\omega_{k1}^i(t)$ and $\omega_{k2}^i(t)$ are calculated using the learning automata algorithm as follows:

$$\begin{cases} \omega_{k1}(t) = \omega_{k1}(t-1) + \alpha(1-\omega_{k1}(t-1)), \\ \omega_{k2}(t) = \omega_{k2}(t-1)(1-\alpha), \end{cases} \qquad (26)$$

where $\alpha \in (0,1)$ denotes the learning coefficient.

As can be seen, all agents make decisions at the same time interval, which leads to excessive competition. To reduce the competition, Eq. (26) is modified as follows:

$$\begin{cases} \omega_{k1}^i(t_s^i) = \omega_{k1}^i(t_{s-1}^i) + \alpha(1-\omega_{k1}^i(t_{s-1}^i)), \\ \omega_{k2}^i(t_s^i) = \omega_{k2}^i(t_{s-1}^i)(1-\alpha), \end{cases} \qquad (27)$$

where t_s^i denotes the start time of step s when an agent i makes a decision. t_s^i is calculated as follows:

$$t_s^i = \frac{1}{|N_i|+1} \sum_{j \in \{N_i, i\}} t_{s-1}^j + (1+\varepsilon), \qquad (28)$$

where ε is a stochastic number in $(0.01, 0.05)$, following the event known as a small probability event and acceptable here. The information about the coalitions needs to be updated when agent i makes a decision. Hence, TR_k^i is modified as follows:

$$\tilde{\mathrm{TR}}_k^i = (\omega_{k1}^i(t_s) \cdot \mathrm{TP}_k^i + \omega_{k2}(t_s) \cdot \mathrm{TF}_k^i) \frac{\mathrm{RT}_k - \mathrm{CA}_k(t_s)}{\mathrm{RT}_k}, \qquad (29)$$

where RT_k denotes the requirement of task k, and $\mathrm{CA}_k(t_s)$ denotes the capability of coalition k at time t_s.

Therefore, agent i makes a decision as follows:

$$k_{\max} = \arg\max_k \ \tilde{\mathrm{TR}}_k^i. \qquad (30)$$

Agent i chooses $C_{k_{\max}}$. As can be seen from Eq. (29), agent i makes a decision based on the real-time information of the coalitions. As the requirement of every task is limited, this procedure could reduce excessive competition, which enhances the probability to get into the final coalition for the agent. Besides, this procedure could benefit the next round of decision making. In addition, agents make decisions at almost the same time, as the ε in Eq. (28) is a vary small number. Thus, the total time cost will not be much larger in each iteration of decision making.

4 Cognitive inertia and cognitive compensation

The stochastic mechanism proposed in Section 3 could reduce the exclusive competition during coalition formation. However, there is still a problem of guaranteeing the accuracy of the task requirement estimation made by the human commanders, since there will always be differences between the actual task requirements and their estimates. This is a crowd intelligence system organized by agents, humans, and networks, as summarized in Pan (2016). Therefore, to reduce such differences, a cognitive compensation mechanism is proposed based on the cognitive inertia. The cognitive inertia is defined as follows:

Definition 1 The fact that a human always calculates the task requirement using his/her own evaluation standard which will not change to a large extent for a human being is called the cognitive inertia.

With the help of the cognitive inertia, cognitive compensation is used for the human. The requirement of task k is given as $\mathrm{RT}_k = \{\mathrm{RT}_k^1, \mathrm{RT}_k^2, \cdots, \mathrm{RT}_k^m\}$, where m denotes the total number of the capabilities in task k. The capabilities of a single agent i is given as $\mathrm{AC}_i = \{\mathrm{ac}_i^1, \mathrm{ac}_i^2, \cdots, \mathrm{ac}_i^m\}$. Usually, the coalition C_k that executes task k is formed when the following condition holds:

$$\sum_{i \in C_k} \mathrm{AC}_i^l \geq \mathrm{RT}_k^l, \quad \forall\, l \in [1, m]. \qquad (31)$$

To deal with the difference between the task requirement and the actual task requirement, a cognitive compensation mechanism is proposed to verify RT_k.

Every agent, such as agent i, has a cognitive compensation rate vector $\boldsymbol{\gamma}_i = [\gamma_i^1, \gamma_i^2, \cdots, \gamma_i^m]$, where γ_i^m denotes the cognitive compensation rate for the capability indexed by m in RT. The procedure of the compensation is carried out as follows. Initially, coalition C_k is formed when Eq. (28) holds, and then task k is executed. When the mission is over, agents calculate the compensation rate γ_i^l for the next commissions. The calculation of γ_i^l is as follows:

$$\gamma_i^l = \frac{\mathrm{RT}_k^l - (C_k^l - \tilde{C}_k^l)}{\mathrm{RT}_k^l}, \quad \forall\, l \in [1, m], \tag{32}$$

where $\tilde{C}_k^l$ denotes the capability indexed by l of coalition C_k when the mission is over, and C_k^l denotes the capability when the coalition is formed. Thus, RT_k is verified as

$$\tilde{\mathrm{RT}}_k^l = \mathrm{RT}_k^l(1 + \gamma_i^l), \quad \forall\, l \in [1, m]. \tag{33}$$

γ_i^l is task independent, and the agents could make it more reasonable by communicating with the agents of other tasks. Here the concept of the could model is used to improve the rationality. The digital characteristics of a cloud are $[\mathrm{Ex}, \mathrm{En}, \mathrm{He}]$, which denote expectation, entropy, and super entropy, respectively. Here γ_i^l is regarded as a drop, a concept in the could model. From the perspective of agent i, a cloud consists of drops $\{\gamma_{ij}^l | j \in N_i\}$. Ex_i is figured out through data fitting, and the fitting curve is as follows:

$$y_{ij}^l = \exp\left(\frac{-(\gamma_{ij}^l - \mathrm{Ex}_i^l)^2}{2{\mathrm{En}_i^l}^2}\right), \ \forall\, j \in N_i, \ \forall\, l \in [1, m], \tag{34}$$

where y_{ij}^l denotes the unified number of drops, which equals γ_{ij}^l, and $\max_{j \in N_i} y_{ij}^l = 1$.

En_i^l is the standard deviation of $\{\gamma_{ij}^l | j \in N_i\}$ and He_i^l is the standard deviation of $\{\tilde{\mathrm{En}}_{ij}^l | j \in N_i\}$, where

$$\tilde{\mathrm{En}}_{ij}^l = \sqrt{\frac{-(\gamma_{ij}^l - \mathrm{Ex}_i)^2}{\ln y_{ij}}}, \ \forall\, j \in N_i, \ \forall\, l \in [1, m]. \tag{35}$$

After $[\mathrm{Ex}_i^l, \mathrm{En}_i^l, \mathrm{He}_i^l]$ have been calculated, γ_i^l is updated as follows:

$$\begin{cases} \gamma_i^l = \mathrm{normrnd}(\mathrm{Ex}_i, \tilde{\mathrm{En}})_i^l, & \forall\, l \in [1, m], \\ \mathrm{En}_i^l = \mathrm{normrnd}(\mathrm{En}_i^l, \mathrm{He}_i^l), & \forall\, l \in [1, m], \end{cases} \tag{36}$$

where $\mathrm{normrnd}(x, y)$ returns a random number that obeys the normal distribution with x being the mean value and y the standard deviation.

5 Simulation

Three kinds of agents, 15 in total, are set here, denoted by H, M, and S, which are labeled by blue box, red box, and green box, respectively (references to color refer to the online version of the figure). The parameters of the agents are given in Table 8.

Table 8 Parameters of agents

Agent type	Detectors assembled	Maximum fire power	Probability of a first-round hit
H	SAR, CCD, ERD	0.5	0.85
M	SAR, CCD	1.0	0.77
S	CCD, ERD	0.7	0.82

Agents patrol in the area of the reconnaissance mission and then the combat mission. To distinguish the ability of an agent, the ability of an agent to execute a task is called desire here. The desire of an agent to execute the reconnaissance mission and the combat mission is shown in Figs. 3 and 4, respectively.

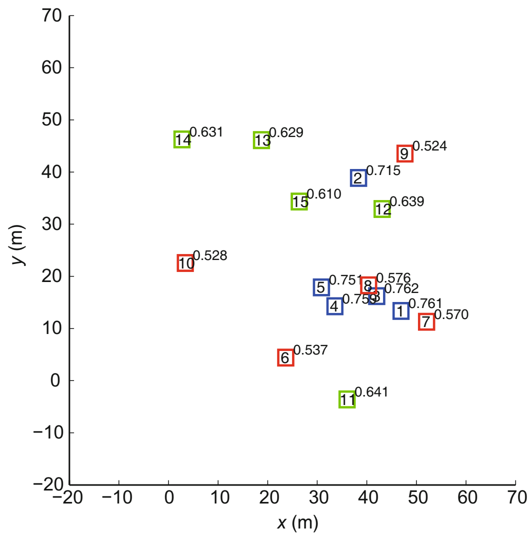

Fig. 3 Desire for the reconnaissance mission

Agents patrol in the area of 90 m × 90 m. The desire of an agent is shown at the top right of the box. The blue point in the center of Fig. 4 means the target, and the blue circle means the range of the fire power in relation to the target.

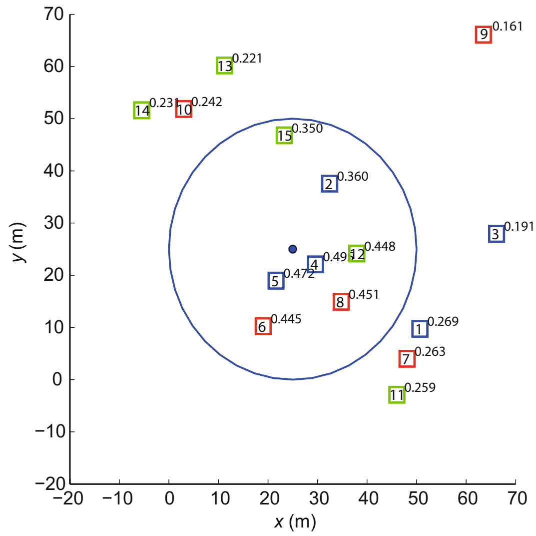

Fig. 4 Desire for the combat mission

The five abilities of agents in the multi-agent system associated with Figs. 1 and 2 are shown in Tables 9 and 10, respectively.

Table 9 Five abilities of agents for the reconnaissance mission

Agent	Cm	R	Ct	M	E
1	0.3949	1.0000	0.3345	0.6299	0.8380
2	0.3002	1.0000	0.3229	0.4287	0.8346
3	0.4133	1.0000	0.3278	0.6299	0.8341
4	0.3877	1.0000	0.3336	0.6299	0.8330
5	0.3334	1.0000	0.3237	0.6299	0.8338
6	0.0429	0.5000	0.5997	0.6788	0.8355
7	0.3487	0.5000	0.6049	0.6299	0.8366
8	0.4156	0.5000	0.5942	0.6299	0.8337
9	0.2464	0.5000	0.6054	0.4287	0.8340
10	0	0.5000	0.5677	0.6788	0.8370
11	0.0429	0.8000	0.4410	0.6299	0.8349
12	0.2684	0.8000	0.4514	0.4287	0.8317
13	0.0429	0.8000	0.4416	0.5557	0.8371
14	0.0429	0.8000	0.4628	0.5557	0.8339
15	0.0429	0.8000	0.4473	0.4287	0.8368

In Fig. 3 and Table 9, more neighbors and detectors enhance the desire of an agent to execute the reconnaissance mission. In Fig. 4 and Table 10, agent 8 and agent 2 are of almost the same distance from the target, while agent 8 has stronger abilities in terms of combat, motor, and energy (Table 10), and agent 8 is more able to execute the combat mission. Although agent 8 has a stronger combat ability than agent 2, and agent 2 is closer to the target than agent 8, which makes the energy sufficient. For the agents out of the blue circle, their combat ability is much weaker than that of the agents within the circle.

Table 10 Five abilities of agents for the combat mission

Agent	Cm	R	Ct	M	E
1	0.3112	0.1570	0.0333	0.6299	0.4818
2	0.0429	0.3465	0.2129	0.4287	0.6583
3	0.0429	0.1079	0.0212	0.4287	0.4109
4	0.3368	0.5948	0.2460	0.6299	0.8204
5	0.3011	0.5195	0.2254	0.6788	0.7804
6	0.2384	0.1152	0.3828	0.6788	0.6221
7	0.3593	0.0382	0.0519	0.6299	0.4657
8	0.2406	0.1356	0.3895	0.6299	0.6495
9	0.0429	0.0081	0.0230	0.4287	0.3380
10	0.2498	0.0357	0.0490	0.5557	0.4636
11	0.3217	0.1357	0.0357	0.6299	0.4438
12	0.2348	0.3519	0.3057	0.6299	0.6793
13	0.0429	0.1223	0.0324	0.5557	0.4338
14	0.2498	0.1120	0.0326	0.5557	0.4146
15	0.0429	0.2557	0.2151	0.5557	0.5840

In coalition formation simulation, 30 enemies are randomly distributed in three areas, and 30 agents are assigned in three coalitions to eliminate the enemies. All are set to three different kinds. The simulation is shown in Fig. 5.

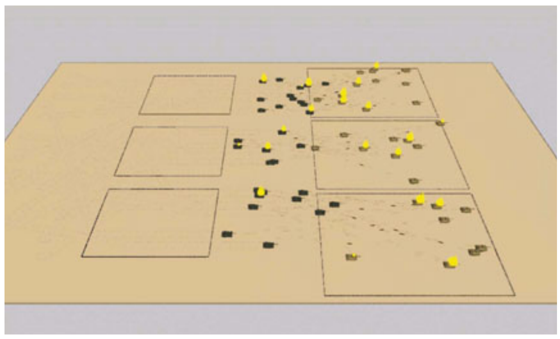

Fig. 5 Simulation of task execution

One hundred rounds of task allocation are simulated, where the energy abilities, detectors assembled, and first-round fit probability are given randomly in each round. The success rate of task execution is 97%. The strategy of destroying enemies is not studied here. Simulation results have shown the effectiveness of the established model and the distributed algorithm.

During each of these 100 rounds of task execution, each required capability for a task is estimated by a human being. The redundancy of the capability before and after compensation for five capabilities is shown in Figs. 6a–6e.

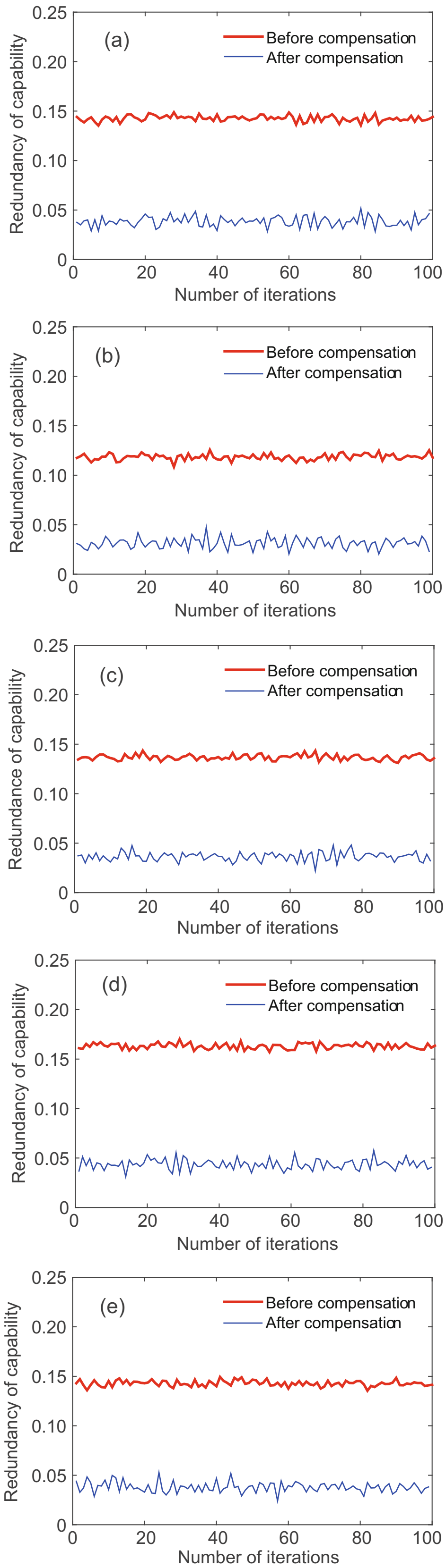

Fig. 6 Redundancy of the first (a), second (b), third (c), fourth (d), and fifth (e) capability before and after compensation

Redundancy is calculated as

$$\text{Redundancy} = \frac{\text{RT} - \text{ATR}}{\text{RT}}, \tag{37}$$

or

$$\text{Redundancy} = \frac{\tilde{\text{RT}} - \text{ATR}}{\tilde{\text{RT}}}. \tag{38}$$

Figs. 6a–6c show that this human being prefers to estimate higher and different values for different capabilities. The compensation mechanism can reduce the gap between the estimate and the actual value, avoiding 60% of redundancy.

6 Conclusions

The models of task-oriented collaborative abilities are designed for coalition formation. Exclusive competitions during coalition formation are reduced by proposing a stochastic mechanism. In addition, an artificial intelligent algorithm named cognitive compensation is proposed to help agents understand the commands from human beings. Numerical simulations show the validity of the proposed model and the distributed artificial intelligent algorithm.

References

An, B., Shen, Z.Q., Miao, C.Y., *et al.*, 2007. Algorithms for transitive dependence-based coalition formation. *IEEE Trans. Ind. Inform.*, **3**(3):234-245. http://dx.doi.org/10.1109/TII.2007.902255

Auer, S., Heitzig, J., Kornek, U., *et al.*, 2015. The dynamics of coalition formation on complex networks. *Sci. Rep.*, **5**, Article 13386. http://dx.doi.org/10.1038/srep13386

Bonabeau, E., Sobkowski, A., Theraulaz, G., *et al.*, 1997. Adaptive task allocation inspired by a model of division of labor in social insects. Proc. Biocomputing and Emergent Computation, p.36-45.

Diao, X.H., Fang, Y.W., Xiao, B.S., *et al.*, 2014. Task allocation in cooperative air combat based on multi-agent coalition. *J. Beijing Univ. Aeronaut. Astronaut.*, **40**(9):1268-1275 (in Chinese). http://dx.doi.org/10.13700/j.bh.1001-5965.2013.0564

Du, J.P., Zhou, L., Qu, P., *et al.*, 2010. Task allocation in multi-agent systems with swarm intelligence of social insects. 6th Int. Conf. on Natural Computation, p.4322-4326. http://dx.doi.org/10.1109/ICNC.2010.5583611

Gensollen, N., Becker, M., Gauthier, V., *et al.*, 2015. Coalition formation algorithm of prosumers in a smart grid environment. IEEE Int. Conf. on Communications, p.5896-5902. http://dx.doi.org/10.1109/ICC.2015.7249262

Haque, M.A., Egerstedt, M., 2009. Coalition formation in multi-agent systems based on bottlenose dolphin alliances. American Control Conf., p.3280-3285.

Haque, M., Egerstedt, M., Rahmani, A., 2013. Multilevel coalition formation strategy for suppression of enemy air defenses missions. *J. Aerosp. Inform. Syst.*, **10**(6):287-296. http://dx.doi.org/10.2514/1.53860

Ketchpel, S., 1994. Forming coalitions in the face of uncertain rewards. AAAI National Conf. on Artificial Intelligence, p.414-419.

Li, D.Y., Du, Y., 2014. Artificial Intelligence with Uncertainty (2nd Ed.). National Defence Industry Press, Beijing (in Chinese).

Lu, S.L., Fang, H., 2016. An improved distributed coalition formation algorithm in MAS. *Contr. Dec.*, in press.

Pan, Y.H., 2016. Heading toward artificial intelligence 2.0. *Engineering*, **2**(4):409-413. http://dx.doi.org/10.1016/J.ENG.2016.04.018

Saaty, T.L., 1990. How to make a decision: the analytic hierarchy process. *Eur. J. Oper. Res.*, **48**(1):9-26. http://dx.doi.org/10.1016/0377-2217(90)90057-I

Saaty, T.L., 2008. Decision making with the analytic hierarchy process. *Int. J. Serv. Sci.*, **1**(1):83-98.

Sandhlom, T.W., Lesser, V.R.T., 1997. Coalitions among computationally bounded agents. *Artif. Intell.*, **94**(1):99-137. http://dx.doi.org/10.1016/S0004-3702(97)00030-1

Sellner, B., Heger, F.W., Hiatt, L.M., *et al.*, 2006. Coordinated multi-agent teams and sliding autonomy for large-scale assembly. *Proc. IEEE*, **94**(7):1425-1444. http://dx.doi.org/10.1109/JPROC.2006.876966

Shehory, O., Kraus, S., 1996. A kernel-oriented model for coalition-formation in general environments: implementation and results. AAAI/IAAI, p.134-140.

Shehory, O., Kraus, S., 1998. Methods for task allocation via agent coalition formation. *Artif. Intell.*, **101**(1):165-200. http://dx.doi.org/10.1016/S0004-3702(98)00045-9

Sichman, J.S., Conte, R., Demazeau, Y., *et al.*, 1998. A social reasoning mechanism based on dependence networks. Proc. 11th European Conf. on Artificial Intelligence, p.416-420.

Whitbrook, A., Meng, Q.G., Chung, P.W.H., 2015. A novel distributed scheduling algorithm for time-critical multi-agent systems. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, p.6451-6458. http://dx.doi.org/10.1109/IROS.2015.7354299

Ye, D.Y., Zhang, M.J., Sutanto, D., 2015. Decentralised dispatch of distributed energy resources in smart grids via multi-agent coalition formation. *J. Parall. Distr. Comput.*, **83**:30-43. http://dx.doi.org/10.1016/j.jpdc.2015.04.004

Zhao, W.Q., Meng, Q.G., Chung, P.W.H., 2016. A heuristic distributed task allocation method for multivehicle multitask problems and its application to search and rescue scenario. *IEEE Trans. Cybern.*, **46**(4):902-915. http://dx.doi.org/10.1109/TCYB.2015.2418052