IOP Conf. Series: Journal of Physics: Conf. Series **1487** (2020) 012034 doi:10.1088/1742-6596/1487/1/012034

Collective Intelligent Decision Making Method Based on Rationality Negotiation

Luo Wentao[1], Feng Pingfa[1,2], Zhang Jianfu[1], Yu Dingwen[1], Wu Zhijun[1] and Liu Haochen[3]

[1.] Department of Mechanical Engineering, Tsinghua University, Beijing 100084;
[2.]Graduate School at ShenZhen, Tsinghua University, Shenzhen 518000
[3.] School of Automation Science and Electrical Engineering, Beihang University, Beijing 100191

Corresponding author Email: zhjf@tsinghua.edu.cn

Abstract. Collective intelligence without considering the rationality of the decision-making process often leads to unreasonable decisions, which brings immeasurable losses to intelligent system. To solve it, a collective intelligent decision-making method based on rationality negotiation is proposed. Each individual with decision-making right is endowed with wisdom by the adapted Q learning algorithm, so that each individual has the perceived ability and individual motivation. Rationality Negotiation Model algorithm is proposed to select a rational decision of group, which reflects collective motivation of group. Then the decision result generated by the collective motivation is transformed into knowledge feedback to each individual, so that the individual also learn the collective motivation. Finally with the repeated iterative optimization, individual action becomes reasonable in the long run. A case is studied based on industrial enterprises making decision in the face of unexpected situations influence production process. The actual verification shows that the proposed decision-making scheme for intelligent system is reasonable.

1. Introduction

From the individual-level intelligence to the social-level intelligent transformation process, collective intelligence often ignores the rationality of the consequence when completing the specific tasks, resulting in the group's overall goal losing its practical significance. Moreover, it could never be rational if one expert makes decision with his or her emotion[1]. In conclusion, the consistency of individual goals dominates the irrationality of the overall goal, which reflects the logical issue of negotiation and cooperation in collective decision.

There is often an orthogonal relationship between individual goals and overall goals[2]. Similar to human group decision-making[3], if the artificial intelligence group lacks an objective and fair negotiation mechanism, the decision will also be infected by one person, resulting in the loss of rationality of the overall goal eventually[4].

In this paper, a collective intelligent decision-making method based on rational negotiation is proposed to improve the credibility and the generalization ability of decision making. Rationality Negotiation Model is presented to illustrate how to select a rational decision among multiple decision schemes. Then an adapted decision-making paradigm based on Rationality Negotiation Model is established to describe how multi-agents make rational decisions from a long-term perspective. To verify the validation of the proposed method, an engineering application case of equipment action plan is given.

2. Rationality Negotiation Model

Rational decision-making is a robust decision made in long-term thinking. In general, the existing negotiation method like Contract Network Agreement[5] only requires agents to participate in one-time negotiation according to the given agreement. There is no repeated discretion and check between agents, which makes decision-making lose rational. A new rational negotiation model is proposed to ensure that the multi-agents participating in the decision-making can cooperate and play repeatedly, making the final robust result more convincing.

2.1. Input preprocessing

Suppose there is a task in the system needs to be executed. The action vector of the agent is expressed as $\boldsymbol{A} = \{0 \leq a_j \leq 1, 1 \leq j \leq N\}$, a_j stands for the probability that the j-th agent will perform the task, the probability value depends on the processing capacity and the completion time. The shorter the completion time, the larger the probability value. If the agent has no capacity to deal with the task, the probability is 0. The probability value could be defined by a linear function in formula (1).

$$a_j = C_{capacity} \times (h + \frac{1-h}{r}) \tag{1}$$

Where $C_{capacity}$ is a binary variable, $C_{capacity}$=1 indicates that agent_j can handle the task, $C_{capacity}$=0 means agent_j has no ability to handle the task; r is an integer variable, indicating the rank level of the completion time of all agents, the shorter the completion time, the higher the ranking. The value of h ranges from [0, 1]. The larger the weight value, the smaller the agent acceptance probability is affected by the ranking.

2.2. Model

The dynamic recursive mechanism of the rationality negotiation model is expressed in formula (2).

$$\begin{aligned}
&A_{n+1,j} = w_{a,j}A_{n,j} + \sum_i \sum_k w_{b,ik}y_{n,i} \\
&\mathrm{w}_{a,j} = \mathrm{w}_{a,j} + \Delta w_{a,j}, \Delta w_{a,j} = -\eta \frac{\partial \phi(A_{n+1})}{\partial w_{a,j}} \\
&\mathrm{w}_{b,ik} = \mathrm{w}_{b,ik} + \Delta w_{b,ik}, \Delta w_{b,ik} = -\eta \frac{\partial \phi(A_{n+1})}{\partial w_{b,ik}} \\
&y_n = MC(Action_n), \mathrm{y}_n = \{y_{n,i} \in binary | 1 \leq i \leq N\} \\
&\qquad\qquad (\mathrm{i, j, k} = 1,2, \cdots, \mathrm{n})
\end{aligned} \tag{2}$$

Where, $A_{n+1,j}$ represents the j-th element of A at the n+1th iteration,$w_{a,j}$ represents the weight between the j-th element of the A and the j-th agent,$w_{b,ik}$ represents the connection weight between the i-th element of $y_n$and the k-th agent $\emptyset(\cdot)$ is an optimization function is consistent with corporate goals. MC($\cdot$) is a Monte Carlo map, the output data $\mathrm{y} = \{y_i \in binary | 1 \leq i \leq N\}$ of MC($\cdot$) is a binary type (0 or 1), which is obtained by simulating the input action probability value, where $y_i = 1$ indicates that the i-th agent issues an execution command request, $y_i = 0$ indicates that the i-th agent does not issue an execution request; w_a, w_b are weight vector of the input action vector and feedback action vector respectively. Each connection in the model represents a weight variable. The weight variable is used to adjust the probability of the action vector in the agent.

The weight vectors are an effective way for agents to cooperate and play together. η is the learning rate parameter, which is valued in [0, 1]. The larger the value is, the faster the weight vector gradient decreases in the model, but it may fall into the local optimal solution. Therefore, the appropriate value should be selected according to the actual situation.

2.3. Rationality judgment

After the execution request scheme y is generated, the rationality judgment is performed firstly, and y that satisfies the rationality judgment condition is the final scheme. The rationality judgment condition can be defined by the constraint theory, which is denoted as follows:

$$\begin{gathered} \emptyset(\mathbf{A_n}) \geq \text{threshold} \\ \psi(\boldsymbol{y_n})\ is\ constrained \end{gathered} \tag{3}$$

Where, ψ is the constraint function group, which is set according to the current resources of the enterprise, for example, $\sum_{i=1}^{N} y_i = 1$ indicates that only one agent is allowed to take the task. If the judgment condition is not satisfied, the decision-making scheme is changed by utilizing gradient descent method to adjust the weight of the feedback node and input action node.

3. Collective Intelligent Decision-making Method Based On Rationality Negotiation

According to the rationality negotiation model, the collective intelligent decision-making method based on rationality negotiation (CIDMRN) is established. In order to describe the rational collective intelligent decision-making mechanism, the running process is as shown in figure 1. The Rationality Negotiation Model algorithm is added to evaluate and affect the behaviour of each agent.

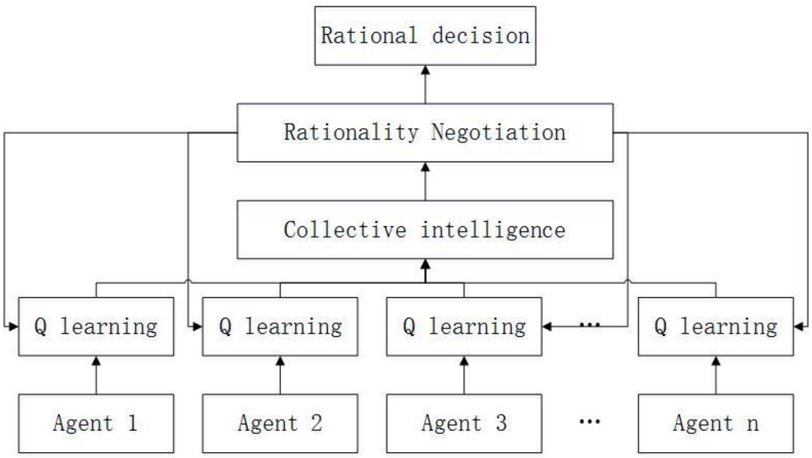

Figure 1. CIDMRN concept chart

3.1. Environment definition

Clarify the dynamic changes in external environment that do damage to the system, and treat these changes as a problem need to be solved. Then, Case-based reasoning (CBS) method[6] is applied to dig out the solution to the problem and the solution could be decomposed into individual-level subtasks. The set of subtasks can be represented by $\boldsymbol{T_{ask}}=(t_1, t_2, \cdots, t_g)$, where g stands for the number of subtasks. Finally, functional information of all individuals in the system are collected, which could be denoted as $\boldsymbol{F_A} = (f_{A1}, f_{A2}, \cdots, f_{Am})$, where m is the total number of all individual functions. The task-function matrix $\boldsymbol{T_F}$ could be constructed based on set $\boldsymbol{T_{ask}}$ and $\boldsymbol{F_A}$.

The element of matrix $\boldsymbol{T_F}$ takes a value of 0 or 1. $C_{tg,fAm} = 0$ means that completing subtask t_g does not require the functionf_{Am}, and $C_{tg,fAm} = 1$ indicates that completing the subtask t_g requires the function f_{Am}. The judgment conditions that whether t_i could be solved by the system are shown as equation(4).

$$T_F = \begin{bmatrix} C_{t1,fA1} & a_{t1,fA2} & \cdots & a_{t1,fAm} \\ C_{t2,fA1} & C_{t2,fA2} & \cdots & C_{t2,fAm} \\ \vdots & \vdots & \ddots & \vdots \\ C_{tg,fA1} & C_{tg,fA2} & \cdots & C_{tg,fAm} \end{bmatrix}$$

$$\sum_{j=1}^{m} C_{ti,fAj} > 0, 1 \le i \le g \tag{4}$$

If condition (4) could be met, it means there are individuals in the system that obsessing the function of performing the task i. If there is a task i ($1 \le i \le g$) that no individual in the system can complete, the system can reject the entire task or seek out-of-system company for assistance.

3.2. Individual smart decision

Adapted reinforcement learning algorithm is used to empower each individual with independent decision making ability.

The tuple method is utilized to express the intelligent behaviour of the agent, which is as follow:

$$\mathrm{agent_i} = (\mathrm{S_i}, \mathrm{A_i}, \mathrm{F_i}, \mathrm{P_{i,s\to a}}, \tau_i, \mathrm{R_i})$$

Where, S_i is the state set of the i-th agent. $\boldsymbol{A}$ is the action set of the i-th agent. $\boldsymbol{F_i} = (f_{i,1}, f_{i,2}, \cdots, f_{i,m})$ is the function set, m is the total number of functions of all agents, and $f_{i,m} = 0$ means the i-th agent do not have the m-th function, $f_{i,m} = 1$ means that the i-th agent has the m-th function. $P_{i,s\to a}$ is the probability that $agent_i$ takes action a in state S. $\mathrm{R_i}$ is a long-term reward function. τ_i is the discount factor. The state space $\boldsymbol{S_i} = \{t_s, t_e\}$ contains the time span of $agent_i$ from now to the next busy time denoted as t_s and processing end time denoted as t_e. t_e could be defined as $\langle t_{basic}, h\rangle$, where$t_{basic}$ is the basic completion time of the $agent_i$, which is the fixed value. h is the work time consumed prior to this time step, then $t_e = t_{basic} - h$.

The action space can be expressed as $A_i = \{0,1\}$, 0 in the action space indicates that the agent is not enabled, 1 indicates that the agent is enabled. $P_{i,s\to a}$ is specified in formula (5).

$$P_{i,s\to a} = \begin{cases} 1 - \dfrac{1}{1+\exp(D(t_s - t_{basic}) + E)}, t_s \ge t_{basic} \ and \ (T_F)_n \times F_i^T = \displaystyle\sum_{j=1}^{m} f_{i,j} \\ 0 \qquad\qquad , t_s < t_{basic} \ or \ (T_F)_n \times F_i^T \ne \displaystyle\sum_{i=1}^{m} f_{i,j} \end{cases} \tag{5}$$

Where, t_s is the state value corresponding to the current time of the state space, $(T_F)_n$ indicates the task-function set of the n-th subtask. The meaning of the above formula is that the more idle the i-th agent is on the condition that it has the function to deal with subtask n, the higher the probability that the agent is enabled.

If the agent is busy or it has no corresponding function to tackle subtask n, the agent couldn't be enabled. Next, a training set is collected to estimate parameters D and E in equation (5) by recording the probability $P_{i,s\to a}$ in different t_s and t_{basic}. The parameters D and E are solved by the maximum likelihood estimation of the constructed training set. The solution process is formula (6).

$$\min_{z=(A,B)} F(z) = -\sum_{i=1}^{M} \left(\frac{1 + t_s - t_{\mathrm{basic}}}{2} \ln(p_i) + \frac{1 - t_s + t_{basic}}{2} \ln(1 - p_i) \right) \tag{6}$$

The function of the long-term return function R_i of i-th agent in the t-th state is expressed in formula (7).

$$R_{\mathrm{t}} = \sum_{i=t}^{\infty} \tau^{(i-t)} \mathrm{r}_t(s_i, a_i)$$
$$\mathrm{r}_t(s, a) = \begin{cases} \dfrac{e^{t_s} - e^{-t_s}}{e^{t_s} + e^{-t_s}}, a_t = 1 \\ 0 \quad , a_t = 0 \end{cases} \tag{7}$$

Where, $r_t(s_i, a_i)$ stands for reward acquired by taking action a_i at the time step t under state of s_i. a_t is the action taken at time t, τ is the discount factor, ranging from [0, 1].

$Q^\mu = E[R_t|s_t, a_t]$ is utility expectation function. By searching for different (s_t, a_t), a time series-based action space vector corresponding to the maximum Q value of the i-th agent is obtained which could be described as $A_i=[a_{i,0}, a_{i,1}, \cdots, a_{i,L}]$, where, L means the time interval from the current time step, $a_{i,L}$ represents the action scheme of the i-th agent in the L-th time step in the future , and $a_{i,L}$=0 indicates i-th agent will not be enabled at L-th time step. $a_{i,L}$=1 indicates that i-th agent will be enabled at L-th time step.

According to the above model, a single agent can independently select a solution. When an emergency occurs, the agent firstly evaluates whether it has the ability to handle the event. If agent could deal with the emergency, then it will request to handle the emergency because agents gain reward only by taking action. The decision-making planning of a single agent is completed.

3.3. Collective rational decision

The decision-making plan of all agents is summarized in previous section, and the most reasonable decision-making plan is selected by Rationality Negotiation Model. Then, all agent action space vectors mentioned in previous section are aggregated to get the group action space matrix A_G.

$$A_G = \begin{bmatrix} a_{1,0} & a_{1,1} & \cdots & a_{1,L} \\ a_{2,0} & a_{2,1} & \cdots & a_{2,L} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n,0} & a_{n,1} & \cdots & a_{n,L} \end{bmatrix}$$

Where, $a_{n,L}$ represents the action value taken by the n-th agent at the L-th time step.

The purpose of Rationality Negotiation Model is to ensure the rationality of decision and avoid the decision mistakes owing to individual wisdom dominating the swarm wisdom. The state space of the Rationality Negotiation Model at time step t can be expressed as $\boldsymbol{S_Q^t} = \{n_{count}^t, T_{min}^t\}$, where n_{count}^t is number of enabled agents at time step t in collective decision making, T_{min} is minimum completion time of the enabled agents. The group action matrix A_G consists of the optimal decision-making actions of each independent agent based on adapted Q-learning.

The input action vector should be calculated by equation (1). Rank value is get by comparing all agents with the state space obtained in Individual smart decision process for each time step. The agent which is free and more efficient will have a higher rank value. Capacity value is obtained by action space of all agents.

The purpose function could be set according to real situation, for example, if it's beneficial for company to deal with emergency as soon as possible, the minimal of total completion time is purpose function, which can be expressed in formula (8): The adjustment of weight vector could be applied through gradient descent method in formula (2).

Rationality judgment condition is set, which reflects the rationality index from the perspective of collective level, that is, the action plan that satisfies the constraint is considered reasonable. Rationality judgment condition can be expressed in formula (9).

$$\phi(A_{n+1}) = (w_{a,j}A_{n,j} + \sum_i \sum_k w_{b,ik} y_{n,i}) \times t_{basic,j} \tag{8}$$

$$C = \begin{cases} n_{count} = 1 \\ t_{i,s} > t_{i,basic},\ i = \arg\min_i \{t_{i,\text{basic}} = T_{\min}\} \end{cases} \tag{9}$$

Formula (9) means number of enabled agents is 1, and the enabled agent should be the one with the shortest completion time among the agents that meet the enabling requirement.

If the rationality judgment condition is not satisfied, the behavior adjustment is performed by the dynamic recursive formula group (2). The optimal action space results in the collective decision-making scheme are fed back to each agent action space by weight vector. Then each agent action space and state value are updated, so that a single agent learns how to make decision with global thought. Next the updated single agent decision-making scheme continues to be summarized to Rationality Negotiation Model. During iterative optimization, the most rational decision plan could be acquired when the action plan of the collective decision is stable.

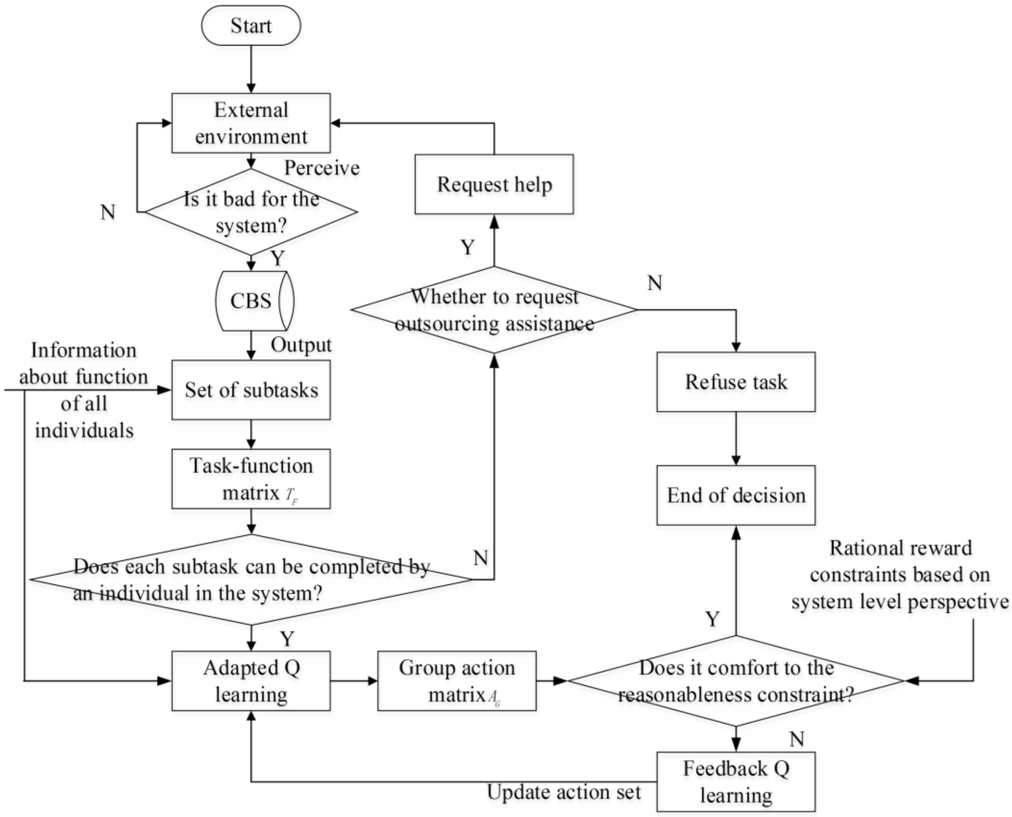

Figure 2. The flow chart of CIDMRN

The basic flow diagram of the above model is shown in figure 2. Firstly, decision-making system perceives the variation of the external environment, and judges whether the changes of the external environment have an impact on the decision-making system. If there is no impact on it, the decision-making system will not take action. But if there is an impact on system, a response solution will be made by CBR , which then will be transformed and decomposes into sub-tasks. Next, the decision system calculates whether the individual in the system has the ability to complete sub-tasks, if not, system will request assistance from outsource system, otherwise these sub-tasks are allocated to each agent to complete. The allocation method is to give the agent the ability of intelligent decision-making, so that it can apply actively according to its own situation and reward mechanism, and the intelligent behavior is given to be implemented through the Adapted Q-learning algorithm. After the task assignment is completed, a Group action matrix is formed, and the established rationality constraint is used to judge whether the matrix meets the condition. If the condition is met, the decision behavior is considered to be reasonable, and the decision is terminated. Otherwise, the intelligent behavior of the agent is updated by using the Feedback Q learning algorithm, which is to update a new task assignment scheme. Feedback Q learning's update principle is intelligent, enabling it to be updated in a way that makes group decisions more sensible.

In summary, the model of this paper gives the system two intelligent thinking skills, 1. task assignment behavior is intelligent; 2. task program update behavior is smart. The first kind of intelligence is to realize the rationality of individual behavior, and the second kind of intelligence is realizing the rationality of collective decision. The collective intelligent decision-making model based on rationality negotiation could respond to the dynamic environment of the enterprise in a timely manner, reflecting the flexibility, adaptability and collective intelligence of decision-making behavior.

4. Case Study

The whole algorithm model is demonstrated with a case study from smart factory.Suppose the factory has five smart processing equipment. The company's goal is to minimize the total time spent on emergency orders under normal production conditions. The equipment information table at time step t=0 is shown in Table 1.

Suppose the emergent processing orders appear at t=0s, and subtasks are generated based on CBS, which are $\boldsymbol{T} = \{T_1, T_2, T_3\}$. The function set of each device can be seen from Table 1, which is expressed as follows:

Table 1. Equipment information table (unit: s)

Device No / Parameter	1	2	3	4	5
t_{basic}	7	6	8	5	9
t_s	0	5	25	3	∞
t_e	5	0	0	0	0
Function	Milling	Assembly	Cutting	Milling	Cutting

Table 2. Contract network mechanism decision

Time step	Device 1	Device 2	Device 3	Device 4	Device 5
t1	0	0	0	0	0
t2	0	0	0	0	1
t3	1	0	0	1	0
t4	0	1	0	0	0

Where t_{basic} is processing time, t_s is the distance from present to the start of the next processing task, t_e is remaining completion time of the equipment being processed. Particularly, When t_s=∞ or t_e=0, it means that the device has no processing task in the current time step.

$$F = \begin{array}{c} \\ D_1 \\ D_2 \\ D_3 \\ D_4 \\ D_5 \end{array} \begin{array}{c} \begin{array}{ccc} F_1 & F_2 & F_3 \end{array} \\ \begin{bmatrix} 1 & & \\ & & 1 \\ & 1 & \\ 1 & & \\ & 1 & \end{bmatrix} \end{array} \qquad T_F = \begin{array}{c} \\ T_1 \\ T_2 \\ T_3 \end{array} \begin{array}{c} \begin{array}{ccc} F_1 & F_2 & F_3 \end{array} \\ \begin{bmatrix} & 1 & \\ 1 & & \\ & & 1 \end{bmatrix} \end{array}$$

Where, the column vectors of F from left to right stand for Milling(F_1), Cutting(F_2), Assembly(F_3), Row vectors of F indicate device serial number. Assume that the task-function matrix can be expressed as above.

T_F satisfies condition (1), which means the devices in the system have the ability to handle all subtasks under this task. Assume the subtasks are arranged to be took at three time points of t=2s, t=10s, and t=15s respectively, therefore, the whole operation process is divided into four time steps, and the order arrival time is used as the time step separation.

After calculating by CIDMRN, the optimal solution has been obtained. The collective decision iteration result is consistent with the first one, indicating that the decision has been stabilized, which means the optimal group decision-making scheme is obtained.

In the same case, the traditional contract network protocol is used for decision making, and the decision results obtained are as Table 2.

Since there are multiple agents with processing capability at the arrival of the task, the task assignment is performed according to the amount of processing capability of the device, but in the first sub-task, the fifth device has higher processing capability (has no order in the future), but processing time is longer than the third device, so the allocation is unreasonable. Moreover machines have the same processing capability for sub-task 2, so conflicts are formed .In summary, CIDMRN is more rational than the contract network mechanism because the contract network mechanism does not consider the long-term benefits and the ability to negotiate conflicts.

5. Conclusion

The model proposed in this paper can realize the phenomenon that the multi-person/machine decision is unreasonable due to different preferences, emotions and goals. The model is forward-looking and is applied to the digital factory environment with intelligent decision-making ability. It could also be applied to intelligent decision-making systems and collective decision-making models. It solves the problem of the existing decision-making system in algorithm implementation and ensures that the decision-making process is reasonable and reliable.

Acknowledgement

We gratefully acknowledge the financial support from the National Natural Science Foundation of China (Grant No. 51575301) and Shenzhen Foundational Research Project (Subject Layout) (Grant No.JCYJ20160428181916222).

References

[1] Yager, R. R 2000 Fuzzy modeling for intelligent decision making under uncertainty. IEEE Transactions on Systems Man & Cybernetics Part B Cybernetics A Publication of the IEEE Systems Man & Cybernetics Society, 30 (1), 60.
[2] Murphy, J 2018 Artificial Intelligence, Rationality, and the World Wide Web. IEEE Intelligent Systems, 33 (1), 98-103.
[3] Vincenzo, I. D.; Massari, G. F.; Giannoccaro, I.; Carbone, G.; Grigolini, P 2018 Mimicking the collective intelligence of human groups as an optimization tool for complex problems. Chaos Solitons & Fractals, 110, 259-266.
[4] Singh, V. K.; Gupta, A. K 2009 In From artificial to collective intelligence: Perspectives and implications, International Symposium on Applied Computational Intelligence and Informatics, pp 545-550.
[5] Zhang, H.; Shi, Z 2004 Dynamic Contract Net Protocol. Computer Engineering.
[6] Kolodner, J 1993 Case-Based Reasoning. Morgan Kaufmann Publishers Inc. p 3–8.