Conflict Resolution of Production-Marketing Collaborative Planning based on Multi-Agent Self-adaptation Negotiation

Hao Li, Ting Pang, Yuying Wu and Guorui Jiang

The Economics and Management school, Beijing University of Technology, No. 100, Pingleyuan, Chaoyang District, Beijing, China

Keywords: Production-Marketing Collaborative Conflict, Multi-Agent, Self-adaptation Negotiation, RBF Neural Network, Q-reinforcement Learning.

Abstract: In order to overcome the lack of adaptability and learning ability of traditional negotiation, we regard supply chain production-marketing collaborative planning negotiation as the research object, design one five-elements negotiation model, adopt a negotiation strategy based on Q-reinforcement learning, and optimize the negotiation strategy by the RBF neural network and predict the information of opponent for adjusting the concession extent. At last, we give a sample that verifies the negotiation strategy can enhance the ability of the negotiation Agents, reduce the negotiation times, and improve the efficiency of resolving the conflicts of production-marketing collaborative planning, comparing to the un-optimized Q-reinforcement learning.

1 INTRODUCTION

In order to meet the market requirements of the dynamic changes quickly and timely, the retailer and manufacturer on the supply chain establish contract early and draw up merchandise procurement plan. Now, the production-marketing planning changes from the simple trading to the consideration of general interests of supply chain. However, because the objectives of different enterprise are always various, their disagreements and conflicts appear usually.

Distributed Agent technology have characters such as interaction, autonomy and learning (Wang, 2013), it is not restricted by time and space, using the negotiation means based on multi-agent in the supply chain production-marketing collaborative planning, not only resolves conflicts but also solves the enterprise-decentralization problem. Many scholars have used the Agent technology in the negotiation of supply chain. For example，Kumar proposed a multi-agent system, selected the best supplier by automatic negotiation based on cost, distance and quality(Kumar, 2011); Sara used the multi-agent system to simulate multi-layer supply chain, controlled the inventory and cost by sharing information, predicting knowledge (Sara, 2012). In order to adapt to environment and the opponent's dynamic information, the scholars in the intelligent negotiation field began to introduce self-learning mechanism into negotiation. For instance, Cheng learned the opponent's utility function by using SVM (Cheng, 2009). Q-reinforcement learning is an algorithm which is independent of environment model, proposed by Watkins(Watkins, 1992), each action of Agent during negotiating has a return function value Q, then evaluates the present action of Agent and predicts the next action of Agent, accomplishes the proposed process by calculating Q(Sui, 2010). Many studies have introduced Q-reinforcement learning into collaborative negotiations (Shen, 2012; Ariel, 2013) to resolve conflicts effectively, optimize collaborative effect.

The studies above all improve the running effectiveness of supply chain, but still there are some shortages. Now there are fewer studies on supply chain negotiation adopting adaptive algorithms; self-learning ability and adaptability of negotiation Agents are relative poor; most of negotiation strategies based on Q-reinforcement learning are not self-adaptive, there are lesser studies on adjusting Q value depending on opponents' behavior, and the convergence speed is still slow. To solve the problems, we propose a multi-agent adaptive negotiation method for the problem of supply chain production-marketing collaborative planning conflict.

Li H., Pang T., Wu Y. and Jiang G..
Conflict Resolution of Production-marketing Collaborative Planning based on Multi-Agent Self-adaptation Negotiation.
DOI: 10.5220/0004830602090214
In *Proceedings of the 6th International Conference on Agents and Artificial Intelligence* (ICAART-2014), pages 209-214
ISBN: 978-989-758-016-1

In consideration of the multi-issues between retailers and manufacturers, we establish negotiation strategy based on Q-reinforcement learning to ensure both sides have different levels of concession, resolve conflicts and obtain satisfying results. At the same time, we use RBF neural network to optimize negotiation strategy, predict opponent's action and adjust Q value, improve the speed of convergence, reduce the negotiation times.

2 NEGOTIATION MODEL

Suppose the negotiation model is defined as $H=\{G,X,W,T,O\}$, the definition of each element are shown as followed:

(1) G represents the set of negotiation Agent, $G=\{A_m, A_r\}$, A_m represents manufacturer Agent, A_r represents retailer Agent;

(2) X represents the set of negotiation issues, suppose issue elements are quantitative, $X=\{x_1,x_2, \cdots ,x_n\}$, n represents the number of negotiation issues, the contents of issues may be price, quantity, date of delivery, defect rate and so on;

(3) W represents the set of negotiation issue weight, $W=\{\omega_1, \omega_2, \cdots, \omega_n\}$, representing the preference value of Agent on issue $x_i(1 \leqslant i \leqslant n)$, suppose both Agents have same preference value;

(4) T represents the set of negotiation time-limit, $T=\{T_m,T_r\}$, T_m and T_r respectively represent the maximum negotiation times that manufacturer Agent and retailer Agent set;

(5) O represents the set of issue boundary values , $O=\{O_m, O_r\}$, where $O_m=[O_{xi_m}^{\min}, O_{xi_m}^{\max}]$, denoting the range of values which Manufacturer Agent can accept on issue x_i , and $O_r=[O_{xi_r}^{\min}, O_{xi_r}^{\max}]$ the range of values which retailer Agent can accept on issue x_i.

Assuming t represents the t-th negotiation; $O_{x_m}^t, O_{x_r}^t$ respectively represents the comprehensive values on the whole issues which manufacturer Agent and retailer Agent proposed at the t-th times, as in (1), (2).

$$O_{x_m}^t = \sum_{i=1}^{n} (O_{xi_m}^t \times \omega_i) \tag{1}$$

$$O_{x_r}^t = \sum_{i=1}^{n} (O_{xi_r}^t \times \omega_i) \tag{2}$$

Where $O_{xi_m}^t, O_{xi_r}^t$ respectively represents the proposal values on the issue x_i that Manufacture Agent and Distributor Agent propose at the t-th time. With the increase of the number t of negotiation, $O_{xi_m}^t$ decreases within the acceptable range $[O_{xi_m}^{\min}, O_{xi_m}^{\max}]$, while $O_{xi_r}^t$ increases within the acceptable range $[O_{xi_r}^{\min}, O_{xi_r}^{\max}]$. When the absolute value of the difference between $O_{xi_m}^t$ and $O_{xi_r}^t$ is less than l (l presents a positive number which is less than 0.5), the negotiation succeeds, and the final trading value O_{xi}^t of x_i takes their average value, as in (3).

$$O_{xi}^t = \frac{O_{xi_m}^t + O_{xi_r}^t}{2} \tag{3}$$

3 THE NEGOTIATION STRATEGY BASED ON Q-REINFORCEMENT LEARNING

Q-reinforcement learning method performs action a_t in the state s_t by Q function, and performs the cumulative awarding values of discount it gains. (Shen, 2007) :

$$Q(s_t, a_t) = r_t + \gamma \max\{Q(s_{t+1}, a_{t+1})\}$$

Where r_t denotes the awarding value Agent accept after transferring from state s_t to state s_{t+1}, the value can be positive, negative or zero. γ the discount factor, $Q(s_{t+1}, a_{t+1})$ the expectations after Agent transfer to state s_{t+1}. During the Q-reinforcement learning , Agent have experienced a series of time steps, during every time step, the learning steps are: (1) To observe current state s_t; (2) To select and perform action a_t; (3) To observe the next state s_{t+1}; (4) To receive the reinforcement signal and adjust Q-expectation according to the established Q-value formula.

According to the thought of Q-reinforcement learning, add reinforcement learning Q-value to the process of negotiation proposal, then give Agent a awarding to make each Agent has a concession to some extent so that make sure both Agents reach agreements as soon as possible. Assuming the r_t is positive when the negotiation succeeds, r_t is negative when the negotiation fails, or r_t is 0 while the negotiation is in progress. During the negotiation the Q-value increases constantly, therefore current $\max\{Q(s_{t+1}, a_{t+1})\}$ is the Q-expectation from the last negotiation. Based on the above hypothesis, the Q-expectation of Agent during negotiation is $\gamma^{t-1}Q$. At first, defining the Q-expectation of Agent as shown in (4), (5).

The initial Q-expectation which retailer Agent proposed on the issue x_i at the t-th negotiation:

$$Q_r = \int_{O_{xi_r}^{\min}}^{O_{xi_r}^{\max}} (O_{xi_r}^t - O_{xi_r}^{\min}) d\, O_{xi_r}^t \tag{4}$$

The initial Q-expectation is that manufacturer Agent proposes on the issue x_i at the t-th negotiation:

$$Q_m = \int_{O_{xi_m}^{\min}}^{O_{xi_m}^{\max}} (O_{xi_m}^t - O_{xi_m}^{\min}) d\, O_{xi_m}^t \tag{5}$$

To control the growth speed of Q-value, we define Q-value as the average expectation as shown in (6), (7).

The average expectation of Q-value which retailer Agent proposes on the issue x_i at the t-th negotiation:

$$\overline{Q_{xi_r}^t} = \frac{\gamma^{t-1} Q_r}{t} \tag{6}$$

The average expectation of Q-value, which manufacturer Agent proposes on the issue x_i at the t-th negotiation:

$$\overline{Q_{xi_m}^t} = \frac{\gamma^{t-1} Q_m}{t} \tag{7}$$

The γ controls the changing speed of the reward value, also affects the concession degree of both Agents in Q-reinforcement learning.

The discount factor γ which retailer Agent proposes on the issue x_i at the t-th negotiation:

$$\gamma_r = 1 + \frac{\theta_{xi_r}^t - O_{xi_r}^{t-1}}{t} \tag{8}$$

Where $\theta_{xi_r}^t$ represents the proposal value that retailer Agent predicts manufacturer Agent on the issue x_i at the t-th negotiation.

The discount factor γ, which manufacturer Agent proposes on the issue x_i at the t-th negotiation:

$$\gamma_m = 1 + \frac{O_{xi_m}^{t-1} - \theta_{xi_m}^t}{t} \tag{9}$$

Where $\theta_{xi_m}^t$ represents the proposal value that manufacturer Agent predicts retailer Agent on the issue x_i at the t-th negotiation.

Agent will make concessions on the basis of the award value during every negotiation, concession degree and each proposal value are defined as shown in (10), (11).

The value of proposal that retailer Agent proposes on the issue x_i at the t-th negotiation:

$$O_{xi_r}^t = O_{xi_r}^{\min} + \overline{Q_{xi_r}^t} \tag{10}$$

The value of proposal that manufacturer Agent proposes on the issue x_i at the t-th negotiation:

$$O_{xi_m}^t = O_{xi_m}^{\min} + \overline{Q_{xi_m}^t} \tag{11}$$

4 NEGOTIATION STRATEGY OPTIMIZATION BASED ON RBF NEURAL NETWORK

4.1 Designing of Network Structure

To reach agreement as soon as possible, we optimize in Q-reinforcement learning and make reasonable concession by predicting opponent's proposal value using RBF neural network (Shi, 2009). Taking retailer Agent as example, to approach $\theta_{xi_r}^t$, we design a 3-layer feed forward network, as shown in Figure 3.

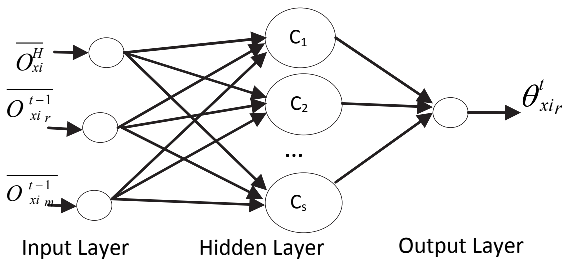

Figure 3: RBF neural network.

Input layer contains three nodes and input vector is O=[o_1,o_2,o_3]=[$\overline{O_{xi}^H}$, $\overline{O_{xi_r}^{t-1}}$, $\overline{O_{xi_m}^{t-1}}$]. $\overline{O_{xi}^H}$ is the average historical negotiation result of both Agents on issues x_i, as shown in (12); $\overline{O_{xi_r}^{t-1}}$ is the average t-1 round proposal result of retailer Agent on issues x_i, as shown in (13); $\overline{O_{xi_m}^{t-1}}$ is the average t-1 round proposal result of manufacturer Agent on issue x_i, as shown in (14). The hidden layer contains s nodes; $C_p=[c_{p1}, c_{p2}, c_{p3}]^T (1 \leqslant p \leqslant S)$ represents the data center of the p-th node with the same dimension as O's; $\Phi=[\Phi(O,C_1), \Phi(O,C_2), \ldots, \Phi(O,C_S)]$ is output matrix in hidden layer and $\Phi(\bullet)$ is radial basis function, achieving a direct mapping of input layer to hidden layer based on Gauss function, as shown in (15); $E=[e_1,e_2, \cdots, e_S]^{T}$ is output weight matrix. Output layer contains only one node, which is the simple linear weighted sum of the hidden layer output matrix, obtaining the possible proposal value $\theta_{xi_r}^{t}$ which retailer Agent predicts manufacturer Agent.

The average historical negotiation result

$$\overline{O_{xi}^{H}} = \sum_{j=1}^{k} \frac{O_{xi}^{j}}{k} \tag{12}$$

Where k is the number of x_i historical negotiation and O_{xi}^{j} is the result of j-th historical negotiation.

The average t-1 round proposal result of retailer Agent:

$$\overline{O_{xi_r}^{t-1}} = \sum_{j=1}^{t-1} \frac{O_{xi_r}^{j}}{t-1} \tag{13}$$

The average t-1 round proposal result of manufacturer Agent:

$$\overline{O_{xi_m}^{t-1}} = \sum_{j=1}^{t-1} \frac{O_{xi_m}^{j}}{t-1} \tag{14}$$

The Gaussian radial basis function:

$$\phi(O,C_{\mathrm{p}}) = \exp(-\frac{\|O-C_{\mathrm{p}}\|}{2\sigma_{\mathrm{p}}^{2}}),\ \mathrm{p}=1,\cdots,S \tag{15}$$

Where σ_{p} is the width of the hidden layer and its size determines the shape of the function.

The output value of the network output layer:

$$\theta_{xi_r}^{t} = \sum_{p=1}^{S} \phi(O,C_P) * E \tag{16}$$

4.2 Network Parameter Learning

(1) The data center parameter of hidden layer is updated based on K-means clustering algorithm, specific steps are as follows:

Step 1: Randomly select S data samples as initial data center C_{p} in hidden layer ($1 \leqslant p \leqslant S$);

Step 2: Group $O=[o_1,o_2,o_3]=[\overline{O_{xi}^{H}}, \overline{O_{xi_r}^{t-1}}, \overline{O_{xi_m}^{t-1}}]$ by the nearest cluster center, if $\|O-Cs\| = \min\|O-C_P\|$ ($1 \le s \le S$), the sample o_1,o_2,o_3 belong to class ψ_s;

Step 3: Calculate the average of the samples in class ψ_s, update it to the new data enter, $C_p = \frac{1}{N_p}\sum O$, N_{p} is the i-th node samples number;

Step 4: If the difference between the new cluster center and the original is less than ε, the obtained cluster center is final basis function center of RBF neural networks; and if more than ε, then return to step 2;

Step5: Confirm the width of the basic functions:

$$\sigma_p = \frac{1}{N_p}\|O - C_p\|^2 .$$

(2) Output layer weights learning based on Gradient-descent algorithm. Set the output error of samples as $D = \frac{1}{2}\left(b_i - \sum_{p=1}^{S}\varphi(O,C_p)*E\right)^2$, Where b_i is the expectation output. Updating weights, $E = E - \eta\frac{\partial D}{\partial E}$, Where η is the learning rate of Gradient-descent algorithm.

5 AN EXAMPLE OF SELF- ADAPTATION NEGOTIATION

We give an example to illustrate the feasibility of this strategy, the effectiveness of negotiation method. Assume in the manufacturing supply chain of one electronic product, retailer submits order plan to manufacturer, including price, quantity, but both sides have conflicts on collaborative plan, in order to avoid reaching an impasse, they start to negotiate using the negotiation strategy. The four items of this plan are regarded as issues, n=2, here lists a part of data. Suppose time-limit of negotiation T_r=20, T_m=25, threshold l=0.2, node S=5, learning rate $\eta = 0.2$, the set of issues boundary and weighting as shown in Table 1.

Table 1: Data of example.

X	$[O_{xi_r}^{min}, O_{xi_r}^{max}]$	$[O_{xi_m}^{min}, O_{xi_m}^{max}]$	W
x1(price)	$[15, 60]$	$[10, 55]$	0.4
x2(quantity)	$[20, 100]$	$[40, 120]$	0.2

Using the adaptive negotiation method proposed, we simulate and implement a conflict resolution of supply chain production-marketing collaborative planning. Simulating results of the proposals (here means price and quantity with time going) submitted by manufacturer Agent and retailer Agent are shown in Figure 4. The proposal of either Agent at each time is expressed as (price, quantity). It can be seen that the manufacturer and retailer continue making concessions. At the fifth time retailer Agent submits the proposal (38.00, 77) and manufacturer Agent submits (38.15, 77), the difference of comprehensive value of either Agent on the two issues is 0.09, which is less than 0.2, so the negotiation finishes. According to formula (3), the final result is (38.08, 77) after negotiating for 5 times, and achieve satisfactory results on both sides. Then we use the Q-reinforcement learning algorithm to simulate, the results are shown in Figure 5. The parameters of Figure 5 have the same means as the Figure 4. The final result is (43.35, 80) after negotiating for 10 times. By comparing Figure 4 with Figure 5, we can find that the Q-reinforcement learning algorithm optimized by RBF neural network can reduce the negotiation times and improve the efficiency of solving the production-marketing collaborative planning conflict.

6 CONCLUSIONS

Resolving the conflicts of production-marketing collaborative planning is an important guarantee of low cost and high-efficiency running of supply chain; it is an efficient way to resolve conflicts by multi-agent self-adaptive negotiation method. We construct a negotiation model, propose a negotiation strategy based on Q-reinforcement learning to make both Agents make concession to some extent, predict opponent's information and optimize negotiation strategy by RBF neural network.

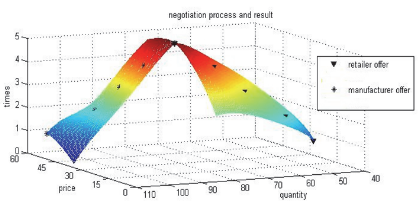

Figure 4: The result of simulation by the adaptive negotiation strategy.

Experiment shows that when compared to only using Q-reinforcement learning, the new method can reduce negotiation times and improve efficiency of resolving conflicts. In future, we will study multi-agent self-adaptive negotiation method for resolving conflicts on supply chain, exploring other learning mechanism to improve the intelligence and adaptability of supply chain.

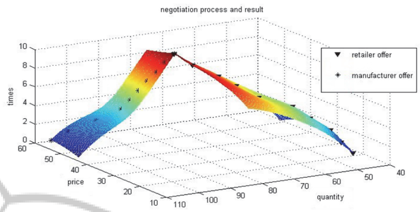

Figure 5: The result of simulation by Q-reinforcement learning algorithm.

ACKNOWLEDGEMENTS

This research was financially supported by National Natural Science Fund of China (71371018, 71071005), and Beijing Philosophy and Social Science Fund of China (13JDJGB037)

REFERENCES

Hao, J. Y., Cheng H. F., 2012. An Adaptive Bilateral Negotiating Strategy over Multiple Items. *In Proceedings of IEEE International Conferences on Web Intelligence and Intelligent Agent Technology.*

Wang, G., Wong, T. N., Yu, C.X., 2013. A Computational Model for Multi-agent E-commerce Negotiations with Adaptive Negotiation Behaviors. *In Journal of Computational Science.*

Kumar V., Mishra N., 2011. A Multi-agent Self Correcting Architecture for Distributed Manufacturing Supply Chain. *In IEEE Systems Journal.*

Sara S., Ali S., Reza S., 2012. Applying Agent-Based System and Negotiation Mechanism in Improvement of Inventory Management and Customer Order Fulfillment in Multi Echelon Supply Chain. *In Arabian Journal for Science and Engineering.*

Yu C., Gao J., Gu M. H., etc., 2009. Automatic Negotiation Decision Model Based on Machine Learning. *In Journal of Software.*

Watkins C., Dayan P., 1992. Q-reinforcement learning. In Machine Learning.

Sui X., Cai G.Y., Shi L, 2010. Multi-agent Negotiation Strategy and Algorithm Based on Q-Learning. *In Computer Engineering.*

Chun S., Lei L., Fan L., etc, 2012. An Adaptive Market-driven Agent Based on Multi-agent

Reinforcement Learning for Automated Negotiation. *In International Journal of Digital Content Technology and its Applications*.

Ariel M., Analía A., 2013. A Reinforcement Learning Approach to Improve the Argument Selection Effectiveness in Argumentation-based Negotiation. *In Expert Systems with Applications*.

Shen, J., 2007. Hierarchical Reinforcement Learning Theory and Method. *In Harbin Engineering University Press*.

Shi, Z. Z., 2009. Neural Network. *In Higher Education Press*, Beijing.