Multi-agent System Collaboration Based on the Relation-Web Model

Maoguang Wang, Hong Mei, Wenpin Jiao, Junjing Jie, and Tingxun Shi

Key Laboratory of High Confidence Software Technologies of Ministry of Education,
School of Electronics Engineering and Computer Science, Peking University,
Beijing 100871, China
wangmg@sei.pku.edu.cn

Abstract. In a multi-agent system, the agents have the capability to work collaboratively to solve the complex task. In recent years social computing provides a new perspective for multi-agent collaboration. This paper first introduces the role model and role relationships including role inheritance, role preference, role binding etc. Then a relation-web model is proposed referring to the social computing research. To a great extent, the relation-web model is used to simulate the social collaboration. The relation weight called trust degree is updated according to their collaboration result. When a complex task is assigned to some agents, the agents will construct the relation-webs for the sub-tasks completion. Finally, to test the relation-web model, an experiment is designed to predict the electricity consumption. The result proves the model to be available and useful while simulating the multi-agent collaboration process for solving the practical problem.

Keywords: relation-web model, multi-agent system, collaboration.

1 Introduction

One main challenge in multi-agent system is to build an effective model for multi-agent system collaboration. Multi-agent system collaboration has become an interdisciplinary research field including economic model, decision theory, logical reasoning and social intelligence etc. Now social computing, simulating complex social processes, provides a new perspective for multi-agent system research. The literature[1] pointed that agent-based social modeling focuses on the cognitive modeling of social behavior and individual agents' interactions. Social computing will strongly influence system and software developments

In multi-agent system, a lot of agents construct an agent society which has the capability to support the complex social behaviors simulation. The multi-agent system collaboration should make full use of the social computing research results. Our major challenge for multi-agent system collaboration is to design models and test the simulation process. Thus our work focuses on the multi-agent system modeling and testing the collaboration model using social computing technologies.

The rest of this paper is organized as follows. Section 2 introduces the related work. Section 3 illustrates the relation-web model in detail. Section 4 gives an

F.L. Wang et al. (Eds.): AICI 2010, Part I, LNAI 6319, pp. 132–144, 2010.

experiment for electricity consumption prediction based on the relation-web model. Finally we conclude in Section 5.

2 Related Work

Considering different agent models and application scenarios, researchers have proposed various collaboration mechanisms [2], such as decision theory, logical reasoning and dynamic planning etc. From the social computing point of view, by integrating the social cooperation mechanism and multi-agent evolution for numerical optimization, Pan et al. [3] provided a social cooperation mechanism which imports the trust degree to denote the historical information for agents. In their work, the acquaintance net model was imported to construct and update the local environment of the agent. It improved the convergence rate by the cooperation characteristic of agents. Tao et al. [4] proposed an extended contract net protocol by introducing the acquaintance coalition policy, the trust degree parameter and the adjustment rules. Virginia Dignum and Frank Dignum [5] highlighted that social relationships determine different types of power links between roles and investigate what is the exact nature of this relationship between roles in an organization and what are the consequences of different structure forms.

Z.E. Chang and S.Li [6] developed a social collaboration model for group decision making process. Multi-agents could collaboratively work together to carry out a common task under a real-time 3D GIS environment. Fei-Yue Wang and Daniel Zeng et al. [1] have done some beneficial research in agent-based social modeling. They thought that the characterization of social structure and relations are typically represented via nodes and ties in network representation, such as complex social networks. Especially simulating complex social processes raises many research issues such as model specification, experimental design, and testing the simulation model. Now there exist the research gaps between individual cognitive modeling and multi-agent social simulation.

Although there is a lot of work on the combination multi-agent theory with social computing, these mentioned works are very useful and inspired us to do the research on the relation-web model to simulate the social network and support multi-agent system collaboration.

3 Relation-Web Model Construction

In this section we first define the basic concepts including role model and role relationships in support of building relation-web model. Then we introduce the relation-web model construction and its application in the multi-agent system collaboration.

3.1 Role Model

Roles are abstract entities which have some goals, responsibilities and capabilities etc. A number of agents are able to play a specified role. Likewise, one agent can play many roles in different scenarios.

Role: R::=<ID, Description, Goals, Responsibilities, Relationship, Collaborators, Authorities, Policies, Priority >

ID is a unique identifier;

Description gives an overview of the role;

Goals is one or several goals for the role to achieve;

Responsibilities describes the specific execution behaviors;

Relationship depicts the relations between the roles, such as role inheritance, role evolution and role conflict etc., which is helpful for role management and reasoning.

Collaborators represents the partners of roles;

Authorities shows the role's behavior permissions and prohibitions;

P**olices** can be regarded as some constrained rules, which includes action policy, goal policy and utility policy etc.;

Priority is used to constrain the role's preference.

Role Relationships:

Role inheritance: $\forall r_1, r_2 \in R$, if $r_1 \subseteq r_2$, the corresponding attributes r_2 is r_1's subset, r_2 inherits r_1 denoted as $r_1 \rightarrow r_2$. Role inheritance has the transitivity, and it provides ways of extending and classifying the existing roles. The users can define different sub-roles by inheritance.

Role assignment: $\forall r_1, r_2, r_3 \in R$, if the role r_1 assigns role r_3 to the agent playing role r_2, it is denoted as $r_1 \xrightarrow{r_3} r_2$. Role assignment provides flexible ways to manage the roles dynamically.

Role conflict: $\forall r_1, r_2 \in R$, if $\exists g_1 \in goals_{r_1}, g_2 \in goals_{r_2}$, and $g_1 \perp g_2$, we call r_1 and r_2 conflict denoted as $r_1 \perp r_2$, where $g_1 \perp g_2$ means r_1 and r_2 have conflict goals.

Role preference: $\forall r_1, r_2 \in R$, if $priority_{r_1} > priority_{r_2}$, it is said that r_1 prevails r_2 denoted as $r_1 \succ r_2$, where $priority_{r_1} > priority_{r_2}$ means r_1 has the higher priority than r_2.

Role evolution: $\forall r_1, r_2 \in R$, if $r_1 \rightarrow r_2$, and r_2 extend r_1's goals, responsibilities, authorities and so on, we define r_1 evolves to r_2 denoted as $r_1 \xrightarrow{+} r_2$. Role evolution shows that the roles may be extended under specific conditions.

The agent is regarded as the autonomous entity which plays the specified roles in the system. Through role binding, the agent instances can be generated from roles.

Role-Agent binding: it is a map from the agents to the roles, the map function is denoted as bind(R): $R \rightarrow 2^{AE}$, 2^{AE} is the power set of the agent AE, that is to say, a number of agents can undertake one role. The role can dynamically bind to agents. Given that the set of agents As $=\{a_1, a_2, \ldots, a_n\}$ and the set of roles Rs $= \{ r_1, r_2, \ldots, r_m \}$, the role binding matrix RB is defined as follows: $RB=Rs^{T}\times As = \{r_1, r_2, \ldots, r_m\}^{T}\times\{a_1, a_2, \ldots, a_n\} = \begin{bmatrix} r_1a_1 & r_1a_2 & \ldots & r_1a_n \\ r_2a_1 & r_2a_2 & \ldots & r_2a_n \\ & \ldots & & \\ r_ma_1 & r_ma_2 & \ldots & r_ma_n \end{bmatrix}$,

$$r_i\, a_j = \begin{cases} 1 & r_i \text{ binds to } a_j \\ 0 & else \end{cases} \quad (1 \le i \le m, 1 \le j \le n \quad m, n \in \mathbb{N})$$

In this definition, r_ia_j represents whether r_i binds to a_j or not. If ra=1, agent a inherits the r's attributes. For example, the r's goals is mapped to a's goals. That is, $\forall\ g_r \in goals_r,\ \exists g_a \in goal_a \Rightarrow \text{bind}(g_r) \rightarrow g_a$.

For example, in an international conference there are various roles such as conference chair, session chair, author, presenter and so on. When a paper is accepted and the author registers for the conference, the author's role evolves to a paper presenter. The role of presenter can bind to specific students or professors. The relationship between roles is depicted as figure 1.

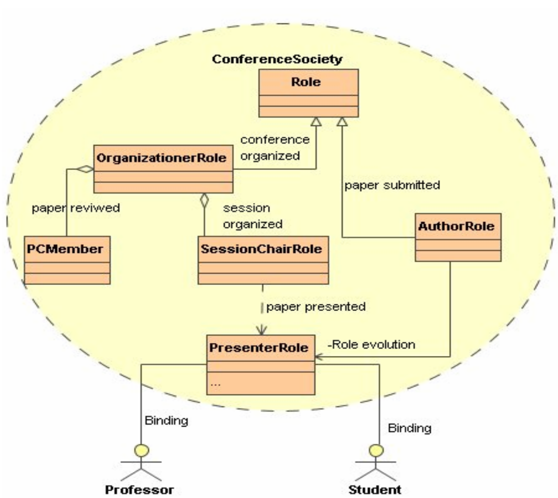


Fig. 1. Role relationships in the conference

In this figure we can see the concrete professor and student agents bind with the roles dynamically in the conference scenario. Furthermore they will play different roles while their attributes change.

3.2 Relation-Web Model

Drawing on the experience of social computing and social intelligence research result, we build dynamic relation-web model to support multi-agent collaboration. The relation-web model describes the related agents to work collaboratively in figure 2.

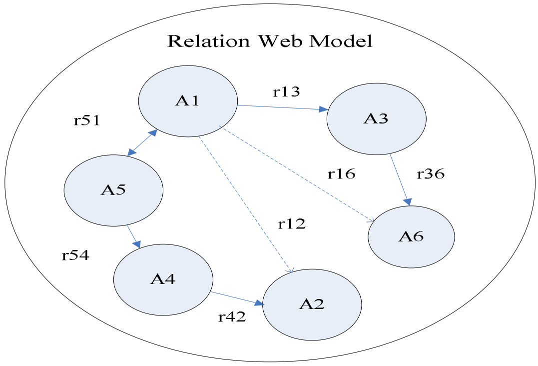


Fig. 2. Relation-web model

Relation-web model: RG= （As, Vs） , where As={A_1, A_2, …, A_n}, A_i is the agent, i=1,2,…,n; Vs={V_1, V_2, …,V_n}, V_i is the set of the agent A_i's relation weights between A_i and its acquaintances, V_i={r_{ij}, i, j=1, 2,…,L, j≠i, L is the number of A_i' s acquaintances}. The value of r_{ij} shows the relation intensity and interaction properties such as competition or coordination. Different from the proposed similar model[7],the relation-web model has reflexivity, non-symmetry and transitivity.

In fact an abstract relation-web model can be constructed by abstract roles. Furthermore, the concrete models can derive from the abstract model by role binding. However, to illustrate the model's essential characteristics, we just use the agents to construct the relation-web model directly. In Figure 2 A_1 establishes the relationship with A_3 and A_5 directly, A_4 establishes a relationship with A_2 directly. The dashed lines r_{12}, r_{16} represent that A_1 can interact with A_2 , A_6 by the relation transitivity, r_{16}= r_{13}⊙$r_3$6, ⊙ is a composite operator. According to the agents' interactions[8], r(A_i, A_j) also called trust degree is defined as r(A_i, A_j)= $\begin{cases} [-1,0) & \mathit{Competition} \\ 0 & \mathit{NA} \\ (0,1] & \mathit{C}\text{oordination} \end{cases}$ .r∈ $[-1,0)$ means that the agents will compete to achieve the specified goal. If r∈ $(0,1]$, it means that the agents will collaborate to achieve the common goal. Otherwise if r=0, it shows that the agents have no relations at the moment. For instance, given a football match the relation weight set {-1, -0.5, 0, 0.5, 1} is used to represent the {intense competition, competition, independent, coordination, full collaboration}.

Supposed that agent is simplified as A_i=<AID_i, Cap_i, $Addr_i$, $Rw_{\underline{i}}$>, the parameters represent the agent identifier, capability, address and the set of relation weights respectively. $Cap_i=\{C_1,C_2,\ldots,C_m\}$ will judge whether the agent has the capability to fulfill the given task. $Rw_{\underline{i}}$={Acq_i, R_i}, the agent A_i*'s* acquaintances Acq_i and the set of relation weight R_i={r_{ij}, j =1, 2, …, |$Acq(A_i)$|, j≠i}. If $r_{ij} \geq 0$, the higher trust degree means that the agents have more probability to work collaboratively. When the agents need to collaborate to fulfill the assigned task, it will search its acquaintance set. Given that l =<A_0, A_1, …, A_n>is the relation chain from A_0 to A_n, $L(l)=|l|$ is the length of $L(l)$. The minimum distance from A_i to A_j denoted as $D(A_i, A_j) = \min(L(l))$ [8]. If $D(A_i, A_j) \neq \infty$, it is said that A_i can reach A_j.

Autonomic society[6] *AS=<AEs, Rel, DF>*, *AEs* is the finite set of agents, *Rel* is the set of trust degree, *DF* is a special agent responsible for agent registration, index and lifecycle management. When agents work collaboratively to fulfill the complex task, it will increase the communication cost during task decomposition and execution. If $|Acq(A)| \in [1, |AEs|-1]$, the average execution cost(task) $\in [O(1), O(|AEs|/2)]$. Therefore, we should make a trade-off between the space complexity and the cost of maintaining the acquaintances.

In practical applications, considering the trust degree changes dynamically, we should adjust the relation-web model dynamically to meet the trust degrees changes. For complex multi-agent systems, agents usually have no global knowledge and trust degrees will not be obtained accurately beforehand. When agents work together, it will succeed or fail. According to the final result, we will increase the trust degree if they succeed finally. Otherwise the trust degree will decrease. This will insure that the agents with higher success rate will increase the relation weight gradually.

Let the trust degree of A_j at t moment is $T(A_i, A_j, t) \in [-1,1]$, and at the initial moment $t=0$, the trust degree of $A_i \rightarrow A_j$, $T(A_i, A_j, 0) = 0$. While agents work collaboratively, T will increase a value δ(reward) if they succeed. Otherwise T will decrease a value γ(penalty) . How to determine the values of δ and γ is a key problem.

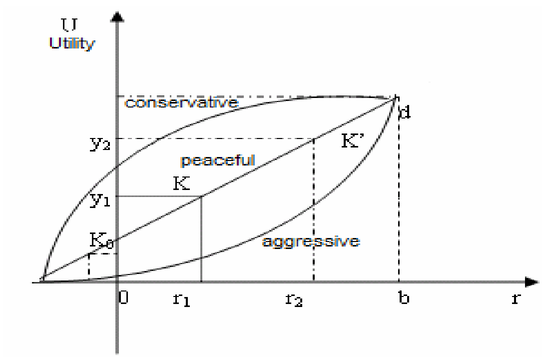


Fig. 3. The three types of agent

We divide the agents into three kinds: conservative agent who is allergic to the loss, peaceful agent whose utility is proportional to the adding profit, and aggressive agent that is more allergic to the gain profit. According to the kinds of agent, $\mathrm{val}(\delta, \gamma)$ is defined as follows.

$$\mathrm{val}(\delta,\gamma) = \begin{cases} \gamma > \delta, \text{Conservative agent} \\ \gamma = \delta, \text{Peaceful agent} \\ \gamma < \delta, \text{Aggressive agent} \end{cases}$$

When A_i and A_j collaborate successfully, $T(A_i, A_j, t+1) = \min(1, T(A_i, A_j, t) + \delta)$. While agent A_i and A_j fail to finish the task except other emergencies, $T(A_i, A_j, t+1) = \max(-1, T(A_i, A_j, t) - \gamma)$.

Furthermore, the trust degree can be defined as a relation weight function denoted as $F_t(x_1, x_2, \ldots, x_k)$, where $x_1, x_2, \ldots, x_k$ is the influence factors, which is capable of describing more complex relation-web models.

3.3 Relation-Web Model Based Collaboration

For the multi-agent system, the capability of solving complex problems is usually established by the agents' collaboration. In our work, once the task T_j is allocated, first we build agent society according to the relation-web model. The irrelevant agents will be excluded or considered with a low probability. While building the agent society, we mainly match the agents' capabilities and compare their trust degrees.

Let AS_T represents the agent society for task T, T is assigned to some agent $A_i \in AS_T$. S_i represents that the relation-web of A_i ($AS_T \supseteq S_i$). The agent$\in AS_T$ has some capability to support the task T completion. Once the relation-web is built, the task is decomposed into some sub-tasks according to the allocation policy. And the agents will be responsible for completing the specified sub-tasks. Finally all the returned sub-results will be summarized. Figure 4 illustrates multi-agent system collaboration for task T based on relation-web model.

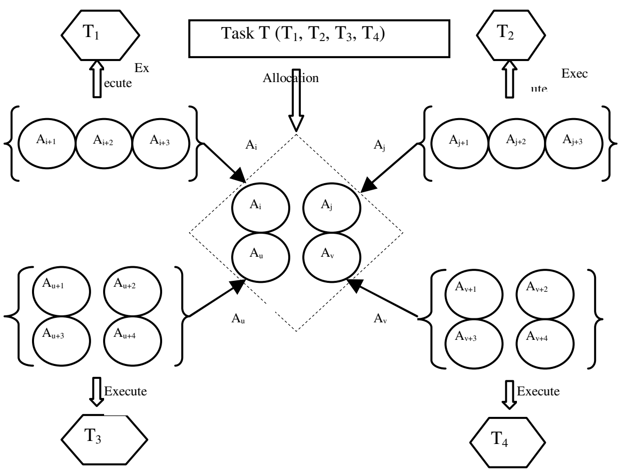


Fig. 4. Multi-agent system collaboration

The execution steps are as follows:

(1) The task T is decomposed into four sub tasks T_1, T_2, T_3 and T_4.

(2) DF agent assigns tasks to the agent society including four agents A_i, A_j, A_u and A_v.

(3) According to the respective relation chain, A_i, A_j, A_u and A_v search and match the capability of the acquaintances to finish the sub-tasks.

Table 1. Basic roles in the relation-web model

ID	**PSA(Power Company)**
Description	According to the power consumption, the PSA role suggests the power price to the government
Goals	Suggest the reasonable power price
Relationship	CA←, →PMA
Collaborators	CA, PMA
Responsibilities	PSA=(WaitStartStep·QueryCustomer·GetCustomerConsumption·CalculateStepTotalConsumption·SubmitTotalComsumption·SendPriceAdvice)$^+$
Authorities	Read: PowerPrice PersonalConsumption Write: StepTotalConsumption Send: PriceAdvice
Polices	PriceMeetMarket==TRUE
Priority	Low
ID	**CA(Electicity Consumer)**
Description	According to the personal income, power price and social influence, the CA role computes the power consumption.
Goals	Evaluate the power consumption
Relationship	→PSA, →PMA
Collaborator	PSA, PMA
Responsibilities	CA=(WaitStartStep·ReceivePrice·ContactNeighbour*·ConsumePower·ReplyNeighbour·SendPersonalComsumption)$^+$
Authorities	Read：PowerPrice NeighboursList PersonalParameters LastStepConsumption StepID Write：StepPersonalConsumption SocialWeights
Policies	PersonalPowerConsumption>0
Priority	Medium
ID	**PMA(Government Sector as the price maker)**
Description	According to the constumer power consumption and the suggested power price, the PMA role makes the power price.
Goals	Specify the power price
Relationship	CA←,PSA←
Collaborator	CA,PSA
Responsibilities	PMA=(ContactCustomer*·GetCustomerConsumption·ListenPriceAdvice·SpecifyPowerPrice)$^+$
Authorities	Read: PowerConsumption CompanyPrice Write: SpecifiedPrice
Policies	PowerPriceAcceted==TRUE
Priority	High

(4) Match capability, sort and select the acquaintances with higher trust degree. A_i, A_j, A_u, A_v will form their respective relation-webs, which will be responsible for T_1, T_2, T_3 and T_4.

(5) The agents collaborate to finish the sub-tasks.

(6) After they all fulfill the sub-tasks, the result are summarized and evaluated.

If the utility of A_i, A_j, A_u and A_v for executing sub-tasks is $\mathrm{cost}(T_{i,j,u,v})$ respectively, the utility of task T is $\mathrm{Cost}(T)=\frac{\sum_i \mathrm{cost}(T_i)}{n}$, n=4.

4 Experiment Design and Analysis

To test the relation-web model in a real application, we design the experiment for electricity forecasting. Firstly, we analyze and define the roles including PSA, CA, PMA in power supply chain. The basic roles depicted in table 1.

PMA charges for making the electricity policy between CA and PSA. PSA provides power to CA, and CA evaluates the power demands according to the family income, electricity price, household appliances, etc. PMA takes a variety of effective measures to regulate the electricity price. The agents build a relation-web model and work collaboratively while playing different roles. Figure 5 depicts the collaboration scenario.

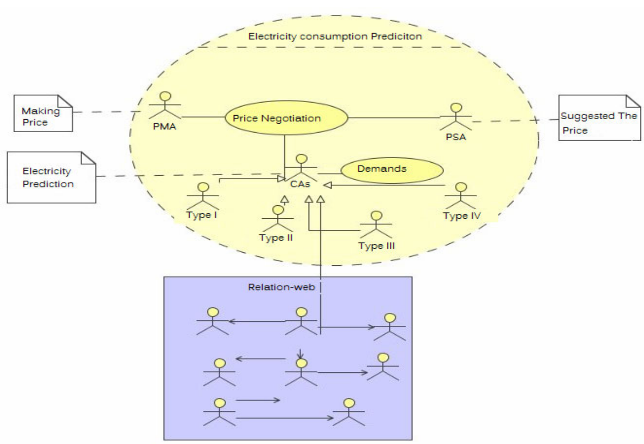


Fig. 5. The relation-web model for electricity consumption prediction

The electricity consumption is affected by a lot of factors. Usually, the electricity consumption can be expressed as C=f(P, Z), where C is the quantity of electricity consumption, P is the electricity price, Z represents the other influence factors including income level, household appliances, habits of using electricity and so on. The electricity demand of CA is expressed by the formula: $C(i,t)=a+bP(i,t)+cZ(i)+\varepsilon(i,t)$, $C(i,t)$ is the family *i*'s electricity consumption in time interval t, $P(i,t)$ is electricity price, $Z(i)$ is the user attribute eigenvector, a, b, c are constant coefficients, $\varepsilon(i,t)$ is the error term.

The literature [9] builds measurable short-term economic model from user requirement, which considers the influences including income and electricity price. They used the model to predict the electricity consumption without considering other social influence factors.

In fact, every family's electricity consumption is not only affected by the consumer's income and electricity price, but also affected by the number of family members and household appliances, saving habits and education etc. By analyzing the influencing social factors, the calculation model of CA's electricity consumption is represented as $C(i,t) = a + bP(i,t) + cZ(i) + dS(i,t) + \varepsilon(i,t)$, where $S(i,t)$ is social factor vector, d is coefficient, the others are same as the model mentioned above. The influencing factors include family income, policy and other social factors such as education, advertisment etc.

We build the multi-agent relation-web model experiment for electricity predition: PSA, CA and PMA represent the power company, the consumer, the government sector respectively. SA is used to integrate the final result. SA sets the related parameters for every participant and controls the simulation process. PMA makes the electricity price. CA estimates electricity requirement accoring to the relation-web model. CAs sends individual electricity demand to SA, SA calculates the total electricity consumption. Figure 6 depicts the structure of multi-agent relation-web platform.

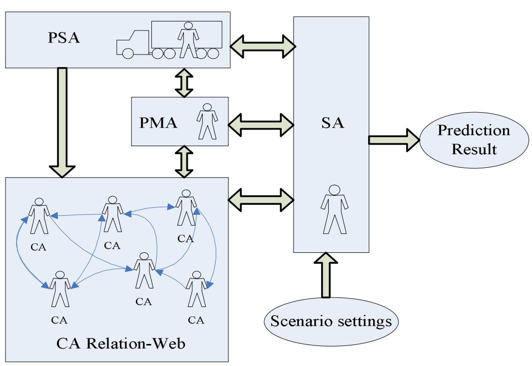


Fig. 6. Relation-web experiment platform

MAGE [10] is a multi-agent platform that is used to develop the relation-web model. CA represents every family and they influence each other. The dynamic influence is quantified by changing relation weight. In this experiment there are two kinds of influences on electricity consumption, one is direct influence from the government policy and education, the other is indirect influence from the neighbours.

In the experiment, there are four types of consumers, the diagram distinguishes these four types with different colours. Figure 7 depicts the relation-web construction.

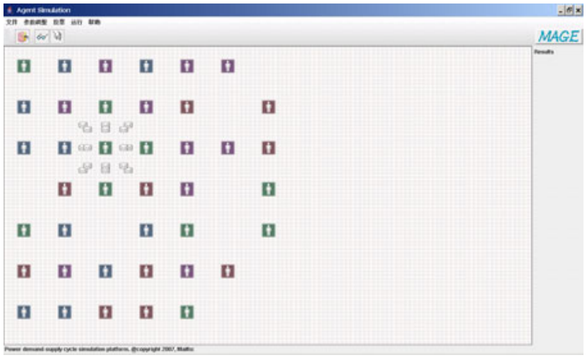


Fig. 7. Relation-web for multi-agent collaboration

As figure 7 depicted, each CA fixed on a grid defined as CA(x, y), (x, y) is the coordinate of CA. To simplify the relation-web model, we define eight grids circling around CA(x,y) as the CA's neighbors. Thus, the maximum number of the neighbors is eight. Every CA is only influenced by its neighbors. For example, $CA(1,1)$ has three neighbors including $CA(1,2)$, $CA(2,1)$ and $CA(2,2)$. The influence is calculated by the trust degree. Every neighbor calculates its influence over $CA(1,1)$ and transfers the total value to $CA(1,1)$ which adds all the values as the influence from the neighbors. The society influence is denoted as $S(i,t) = f(\sum_{j=0}^{n} sw_j)$, where sw_j represents the trust degree for agent *j* to *i*, n is the total number of agent *i*'s neighbors. According to the domain knowledge, we define the trust degree function as $f(x) = \dfrac{1}{1+e^{(-(x-5))}}, x \in [0,9]$.

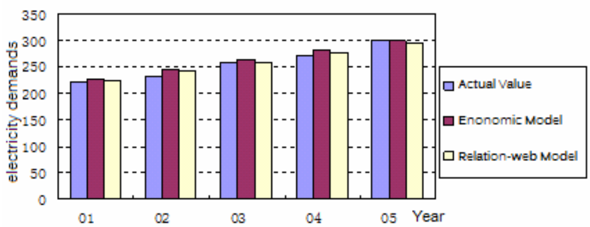
In the simulation experiment, each agent represents different customers which are bound to different roles. We use the model to predict the electricity consumption. The execution steps are as follows:

1) Initialize and construct the relation-web model. The initialization includes setting the parameters for relation-web scale, the types of agent playing different roles, the maximum response time and iteration steps and so on. Then generating the concrete agents randomly.
2) PMA makes the suggested electricity price and informs CAs and PSA.
3) Each CA gets the price, collaborates with the neighbours and calculates the society influences.
4) Each CA takes into account the social influence, estimates its own electricity demands and reports the demands to PSA.
5) PSA collects and calculates all of the consumers' electricity demands, inform the total demand to PMA, and give some suggestions if necessary.

6) PMA decides whether it should adjust the electricity price according to the feedback demands. Goto step 3), start next iteration.
7) Finish the simulation process and output the result.

During the collaboration, the consumers interact with its neighbours to get social influence value.

The experiment result gives the electricity demand prediction illustrated in figure 8.



Actual Vaile.	221.70	232.80	259.20	272.20	300.40
01-05with economic model.	228.45	246.05	264.88	282.85	299.37
01-05 with relation-web model(a=10,b=20,c=30, d=20).	225.43	241.63	259.68	277.10	295.20

Fig. 8. Electricity prediction based on relation-web model

From the diagram, the electricity demand prediction based on the relation-web model has a better result which is closer to the real value.

The experiment result shows we can predict the electricity demands more accurately when considering the social influence factors. In fact when taking into account the social influences based on the relation-web model, the customer's habits will be calculated for the precise prediction.

5 Conclusions and Future Work

Multi-agent system collaboration has become an interdisciplinary research and application field. Referring to the social collaboration mechanism, we provide a relation-web model for simulating simple social network. And the experiment result gave a better prediction for real applications. To improve the efficiency, now we mainly adopt a top-down method during the task allocation in the experiment. However, an efficient relation-web model should be constructed not only by a top-down method, but also constructed and trained by the bottom-up method from the beginning. Our future work will focus on the dynamic relation-web model update by combing these two methods and will take advantage of the fuzzy rules and semantic reasoning to describe complex relation-web.

Acknowledgments. This work is supported by the National Basic Research Program of China (973) under Grant No. 2005CB321805, 2009CB320703, the Science Fund for Creative Research Groups of China under Grant No. 60821003, the High-Tech Research and Development Program of China under Grant No. 2009AA01Z139 and China Postdoctoral Science Foundation No. 20090460151.

References

1. Wang, F.-Y., Zeng, D., Carley, K.M., Mao, W.: Social Computing: From Social Informatics to Social Intelligence. IEEE Intelligent Systems, 79–83 (2007)
2. Wang, M., Shi, Z., et al.: Agent Grid Collaborative Schema Based on Policy. In: Communications in Computer and Information Science. 3rd International Conference on Intelligent Computing, vol. 2, pp. 469–478 (2007)
3. Pan, X.-Y., Jiao, L.-C.: Social cooperation based multi-agent evolutionary algorithm. Journal of Xidian University 36(2), 247–280 (2009)
4. Tao, H., Wang, Y., Guo, M., Wang, H.: A multi-Agent Negotiation Model Based on Acquaintance Coalition and Extended Contract Net Protocol. Journal of Computer Research and Development 43(7), 1155–1160 (2006)
5. Dignum, V., Dignum, F.: Coordinating tasks in agent organizations Or: Can we ask you to read this paper? In: Coordination, Organization, Institutions and Norms COIN@ECAI 2006 (2006)
6. Chang, Z.E., Li, S.: Social collaboration model support for collaborative 3D GIS environment based on multi-agent method. In: The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Beijing, vol. XXXVII, Part B2, pp. 1059–1063 (2008)
7. Chen, G., Lu, R.: The Relation Web Model-An Organizational Approach to Agent Cooperation Based on Social Mechanism. Journal of Computer Research and Development 40(1), 107–113 (2003)
8. Wang, M.: Research on Modeling and Collaboration in Autonomic Computing. Phd Dissertation, Chinese Academy of Sciences (2007)
9. Song, Y., Gu, A., Wu, Z.: Urban residential electricity consumption forecast model. Electric Power 39(9), 66–70 (2006)
10. Shi, Z., Zhang, H., Dong, M.: MAGE: Multi-Agent Environment. In: ICCNMC 2003, pp. 181–188. IEEE CS Press, Los Alamitos (2003)