P081

A TIME-DEPENDENT GROUP MATCHMAKING MECHANISM FOR CREATION OF AGENT-BASED COMMUNITIES

Lawrence Chang[*] Gen Hattori[**] Chihiro Ono[**]
Satoshi Nishiyama[**] Hidenori Nakazato[***] Yoshiyori Urano[***]

[*]GITS, [***]GITI, Waseda University [**]KDDI R&D Laboratories Inc.

t00a172@gits.giti.waseda.ac.jp, {gen, ono, tomo}@kddilabs.jp, {nakazato, urano}@giti.waseda.ac.jp

Abstract

In many agent-mediated electronic marketplaces, a facilitator is exploited to conduct matchmaking in finding agents with the same interests. During matchmaking, the consideration of newcomers and group matchmaking is crucial since favorable newcomers can bring preferable offers to the facilitator for generating better matches and hence maximizing the agents' utility gains, furthermore, group matchmaking is urgently needed in everyday life for collaboration works. In Choi et al's proposal, a solution for considering newcomers based on the Markov decision process is presented. However, it is not applicable for group matchmaking. In addition, the notion of individual utility tables and lifetime for each agent to meet the needs in real world has been neglected. Therefore in this paper, we propose the mechanism in the support of all of these requirements by extending Choi et al's proposal. We also evaluate the facilitator to verify the effectiveness of this mechanism.

1. Introduction

Electronic marketplaces that utilize agent technology for users to ease trading and automate buying and selling tangible goods have been popular for years. Recently, agents have met to find right partners within a deadline in order to exchange services. The marketplaces mediated by agents exploit a facilitator to provide an infrastructure of trade. The facilitator conducts matchmaking to pair existing agents with the same interests. However, many existing facilitators are inattentive to potential newcomers at matching. Favorable newcomers can bring preferable offers to the facilitator to generate better matches for agents and hence maximizes the agents' expected utilities. Current facilitators are also incapable of performing group matchmaking. The function of group matchmaking is urgently needed because daily examples such as playing tennis with a group of four requires good methodology for finding the right partners.

Therefore in this paper, we propose a matchmaking mechanism that takes newcomers into consideration. We achieve this by studying a similar proposal that exploits the Markov decision process in considering newcomers. The facilitator employs this mechanism for group matchmaking in a time-dependent manner to create communities for agents. We also conduct evaluations on the facilitator to verify the effectiveness of the proposed mechanism.

2. Purpose

The example scenario shown in Figure 1 explains the necessity of our research – A user seeks three participants to play doubles tennis at 4pm, sends agent_A off to the facilitator at 1pm. Agent_A registers to the facilitator with user's capabilities such as skills and experience, and other attributes such as the deadline and the location of preferences for finding other agents.

Based on agent_A's registration, the facilitator searches its dynamic database and finds that only agent_B matches agent_A's preferences for playing tennis. At 2pm, interested in playing tennis, agent_C registers. Agent_C waits along with agent_A and agent_B for another agent's arrival as one more player is required for the commencement of a tennis match to take place. At 3pm, agent_D registers. The facilitator now ponders as to whether agent_D is the best candidate for the missing fourth tennis player and either immediately accepts agent_D in the doubles match or waits for other potential new agents before the time deadline since agent_D's skills are far too low compared to the other agents.

Each agent has a different utility function since agent_A and agent_C have subjective rating on agent_D. In addition, as each user sets different commencing time for the tennis game, each agent has a different lifetime according to the user's deadline.

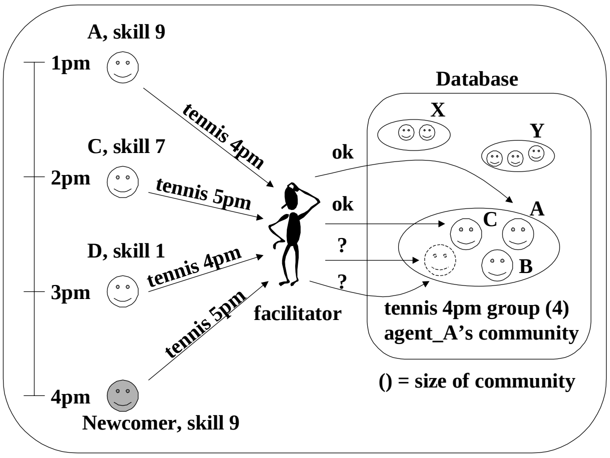

Figure 1 Tennis scenario of matchmaking

In order to realize the above scenario, we believe the following requirements are necessary:

1) Consideration of potential newcomers
2) Group matchmaking mechanism
3) Individual utility function for agents
4) Lifetime restriction on each agent

3. Related Work

3.1 Markov Decision Process

A Markov decision process (MDP) [1] is a model for sequential decision making when outcomes are uncertain. At any decision epoch, choosing an action in a state generates a reward and determines the state at next decision epoch through a transition probability function. The MDP is referred to as a finite-horizon model if the set of decision epochs is finite. Decision makers seek policies or strategies that are optimal for choosing an action.

3.2 MDP for Time-Constrained Trading

Choi et al [2] has proposed a dynamic mechanism for agents in time-constrained trading. Their work focuses on maximizing the agent's expected utility throughout negotiations by exploiting the MDP model for the agent's decision making. In the electronic marketplace, at trading, the agent is paired up with a counterpart to negotiate deals. The agent decides to accept the current offer on hand or to wait for a better one in the next time step by comparing the utilities. If the agent accepts the current one, an immediate utility is gained otherwise a cost is received for waiting. The agent has to finalize a deal within a given deadline. In this model, newcomers are considered during negotiation since each agent utilizes the MDP model for decision-making by receiving updated statistical information continuously.

The agent's trading problem is formulated into a MDP model and elements are as follows:

1) State space: All possible collections of offers
2) Action space: Accept and wait
3) Transition probability function: The changes of offer collections
4) Reward function: A cost if wait and an immediate utility if accept

Every offer that an agent receives can be classified into one of the K categories according to its utility function. All offers in the category i have the same utility value of v(i). The probability of losing an offer in category i is denoted by l_i. A collection of all offers is represented by n and n_i stands for the number of offers in category i. Notation e_i stands for a new offer in category i.

When an agent decides to accept the current best offer, the current utility is defined mathematically as:

$$V_c(n) = \max{}_j (v(j) \mid n_j > 0) \qquad (1)$$

When an agent decides to wait, it receives an expected utility. Given the remaining time t and the change in offer collection n' at next time step, the expected utility can be mathematically formulated as:

$$V_t(n) = c(t) + \sum_{e_i} \sum_{n' \in N} \Pr(n'+e_i) \cdot V^*_{t-1}(n'+e_i) \qquad (2)$$

where $c(t)$ is the cost for waiting. $\Pr(n'+e_i)$ is the probability of the next state and $V^*_{t-1}(n'+e_i)$ is the optimal value at the next time step.

In particular, the probability of the next state is:

$$\Pr(n'+e_i) = \Pr(e_i) \cdot \prod_{j=1}^{K} \binom{n_j}{n'_j} \cdot l_j^{(n_j - n'_j)} \cdot (1 - l_j)^{n'_j} \qquad (3)$$

where n_j denotes the offer collection in category j while n'_j denotes the change in offer collection. Now the formula for optimal value function is

$$V^*_t(n) = \max\left(V_t(n), \max{}_j (v(j) \mid n_j > 0)\right) \qquad (4)$$

Thus the optimal policy for trading becomes:

$$\pi^*_t(n) = \begin{cases} accept \quad if \quad V_t(n) < \max{}_j (v(j) \mid n_j > 0) \\ wait \qquad\qquad otherwise \end{cases} \qquad (5)$$

Although Choi et al's proposal considers deadline trading and potential newcomers, nevertheless, their work alone is inadequate and insufficient in understanding our research.

For requirement 2), decision-making is made by individual agents that illustrate the MDP model as being adopted in a distributed manner. It is feasible when agents negotiate in pairs and the same utility function is employed. However, it becomes complicated when one considers matchmaking in groups and agents with different utility functions. Therefore the act of the facilitator making decisions for all agents is required.

For requirement 3), the sharing of the same utility function is impractical due to the varying of eagerness of each agent.

For requirement 4), the duration of an offer depends on the utility function alone. The notion of the lifetime for agents is neglected. In our work, we intend to overcome these problems.

4. Proposal

4.1 Basic Ideas

We propose a mechanism that considers potential newcomers at matchmaking by exploiting MDP at the facilitator level [3] [4] while agents exploit the MDP model in Choi et al's proposal in a distributed manner. When considering newcomers, the facilitator exploits the Markov decision process model for decision-making. To achieve group matchmaking, the facilitator considers the utility of multiple agents at the same time. In addition, when utilizing the fact that all agents have different utility and lifetime, the facilitator assigns different utility functions and lifetime to each agent for decision-making.

4.2 Assumptions

Since this study concentrates on building a novel matchmaking mechanism for the facilitator, some issues related to matchmaking systems have been neglected. To ensure that the proposed mechanism works well for most multi-agents systems, we have the following assumptions:

1) Agents are rational ones

In Ono et al's work [5], agents' reputations are used to distinguish trustable ones against those that are not. In the proposed model, the trust or reputation of each agent is neglected since agents and their users are assumed to act rationally in every way.

2) Requested agents accept community formation

We assume that other agents who are requested to join the community merely accept the community formation.

3) Agents share the same attribute types.

In the proposed model, we assume that agents share the same ontology for attributes. In addition, as learning each attribute value correctly from the user requires great studies and endurance, such as MARI [6], which concentrates on how to compute user's utility function promptly through a well-designed GUI and various type of predefined function, it is assumed that attribute values are successfully learnt from users.

4.3 Architecture

The flow of the proposed facilitator can be described as follows (we refer it closely to the tennis scenario in Section 2 and Figure 2):

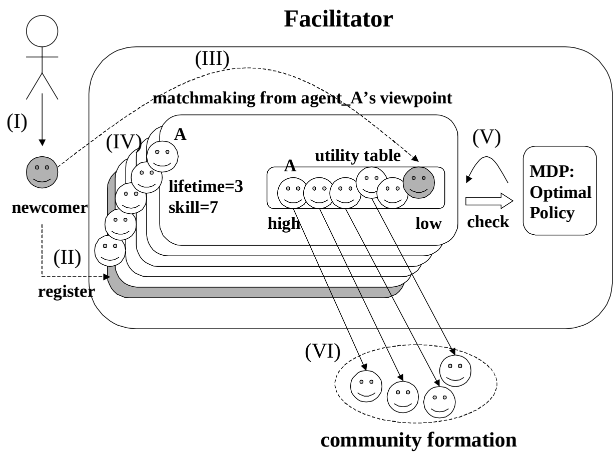

Figure 2 Matchmaking at facilitator's level

To join in a designated community, a user provides eligible values of fixed (deadline, member of the community) and flexible (skills and experience) attributes to his/her agent, (I). An agent maintains the user's preferences. After the agent learns the user's preferences, it registers on behalf of the user to a facilitator located at a server, (II).

Upon registration, the facilitator allows this specific agent to evaluate all existing agents to determine a utility value individually and likewise, other agents also evaluate this agent. For the specific agent, the utility value of other agents are stored in its *utility table* while other existing agents update their utility table by adding the utility value of this one. In other words, others know the specific agent as the *newcomer* ((II) and (III) in Figure 2). When the evaluation phase is completed, the specific agent is then added to the end of an agent list, where all agents await for matchmaking.

The facilitator performs matchmaking sequentially, starting from first agent to last in the agent list, shown in (IV), Figure 2. For each agent (each matchmaking), the facilitator computes a current utility and an expected utility. A current utility is the sum of the top four (tennis scenario, where the community size is four) utility values while an expected utility is the estimation of a maximum utility that can be received at the next time step. The estimation is made using the formula defined in Section 4.4.

If the current utility is larger than the expected utility, the facilitator learns that waiting for agents with higher utility before the deadline might not seem promising, (V). Therefore the specific agent is better off to form a community with current available agents. As a result, a community is formed and the agents are unregistered from the facilitator, (VI). If the next agent who is suppose to undergo matchmaking is unregistered by community formation, matchmaking is skipped for this agent.

When all agents have undergone matchmaking, for those remaining, the lifetime is decrement by one unit and their utility tables are updated with current information. At this point, the facilitator has completed sequential matchmaking for all agents for one time step and proceeds to the next time step to repeat the whole process again. Note that if for the lesser agents exist compared to the community size, the facilitator automatically requests agents to wait and matchmaking is skipped.

4.4 Mechanism and Problem Formulation

In this section, the matchmaking mechanism is formulated into a MDP model and elements of the MDP model for calculating the optimal strategy are defined. The state space consists of all possible collections of agents, n. The action space consists of two actions: *accept* and *wait*. The transition probability functions are defined by the changes of agent collections. This allows the current utility and the expected utility as rewards for accept and wait respectively.

As the proposed model shares similarities to Choi et al's model, some of the formulae defined in Choi et al's proposal are adopted.

Equation (2) remains unchanged in calculating the expected utility since one must also consider newcomers and the probability of losing or retaining each agent. Equation (2) satisfies requirement 4) because parameter t represents the lifetime of an agent. The proposed definition of the optimal utility is the higher value between the current and the expected utility, which is also identical to that of Choi et al. Thus the optimal value function in Equation (4) remains unchanged except that the current utility differs to that of Choi et al. As for optimal policy, since the facilitator has only two actions, the formula from Equation (5) remains as it is and is applied directly.

Apart from the similarities, the proposed facilitator shows two distinct characteristics from Choi et al's work:

1) A group matchmaking mechanism that meets the requirement of real life applications, and
2) An individual utility for each agent to ensure better quality

For group matchmaking, the following formula for current utility is proposed:

$$V_c(n) = \sum_{x=1}^{m} \max{}^{x}(u_a(z) \mid z \in n \wedge n > 0) \qquad (6)$$

where m is the size of a community and $u_a(z)$ denotes the utility value of each agent in the current collection when a specific agent, a, undergoes matchmaking. Equation (6) satisfies requirement 2) because the utility value up to m agents are considered for the source agent for community formation. It demonstrates the proposed group matchmaking mechanism. Equation (6) also satisfies requirement 3) because each agent has a different utility function $u_a(i)$ by the source agent.

As for ensuring a better quality for the individual utility, the facilitator performs matchmaking sequentially for each agent based on the agent's own utility table (Figure 2). In addition, each agent has individual probability of losing due to lifetime h and community formation f, which can be used to calculate transition probability $\Pr(n'+e)$ accurately. We define the transition probability function as follows:

$$\Pr(n'+e) = \Pr(e) \cdot \prod_{x=0}^{a \in n-n'} (h+f)_x \cdot \prod_{y=0}^{a \in n'} (1-(h+f))_y \qquad (7)$$

Equation (7) satisfies requirement 3) because f has utilized the notion of individual utility function of each agent. Equation (7) also satisfies requirement 4) because the probability of losing due to lifetime h depends on the notion of the lifetime of each agent.

5. Complexity Issues

5.1 Computational Complexity

According to Equation (2) and (3), when

calculating the expected utility, the facilitator is required to enumerate all collections of possible utility tables at the next time step for each agent by considering newcomers and agents lost due to lifetime or community formation. However, an unbounded number of utility tables exist. As a result, the facilitator generates infinite utility tables in a single time step and the problem can only be solved in nondeterministically polynomial- time (NP).

5.2 Representation of Utility Table

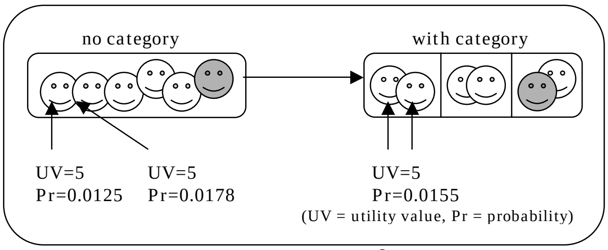

Figure 3 Concept of category

In Choi et al's paper, agents having the same utility values have the same probability of losing and can be treated equally. Therefore the concept of classifying these agents into the same category shown in Figure 3 is used for computational simplicity.

In the proposed model, the losing probabilities for agents that have the same utility value by a specific agent is different since the utility value of these agents by other agents may differ. Thus for taking advantage of the different utility, the proposed model obeys following properties:

1) The facilitator uses different utility table for each agent for calculating the expected utility
2) Agents with different losing probability in the utility table of a specific agent are treated differently during the calculation of the expected utility

However, when applying different losing probabilities to the same utility value, the calculation of the expected utility becomes too complicated, which is O((number of average agents in the utility table)^N). As the reduction techniques for computation are more important than in Choi et al's case, it is assumed that the same utility value has the same losing probability. In other words, the proposed model makes use of property 1) and leaves property 2) to the next study. As a result, agents with the same utility value are treated equally for simplicity, which is like classifying these agents into the same category in Figure 3. The computation then becomes O((number of categories)^N).

5.3 Reduction Technique

So far, some reduction techniques for reducing the complexity have been suggested. In this paper, the threshold setting to eliminate insignificant utility tables with small probability so that the number of utility table becomes bounded is applied (Figure 4). The formula for transition probability function is:

$$\Pr(n'+e_i) = \Pr(e_i)\cdot\prod_{j=1}^{K}\binom{n_j}{n'_j}\cdot(h+f)^{(n-n')}\cdot(1-(h+f))^{n'} \quad (8)$$

where h is the probability of losing due to lifetime and f is the probability of losing due to community formation in category j.

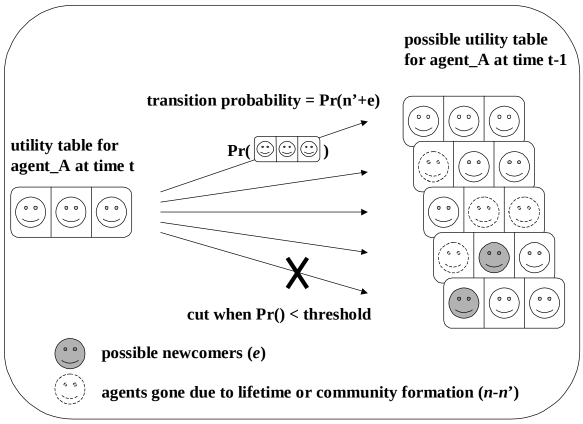

Figure 4 Threshold setting

In Figure 4, agent_A has the current utility table with three agents including agent_A itself at time t. The facilitator then takes this utility table and applies statistical information such as the possible number of newcomers due to arrive at the next time to numerate all possible collection of utility tables at the next time step. For each possible utility table, the facilitator determines its probability value and compares it with the threshold value. If the probability value is smaller than the threshold value, the facilitator cuts it off.

6. Evaluation

6.1 Experiment Settings

To demonstrate the effectiveness of the proposed facilitator, three experiments are conducted:

1) To testify the effectiveness of the reduction technique, the number of utility tables required for calculating expected utility of each agent by using different threshold values are analyzed.

2) To study the effectiveness of the proposed facilitator, the utility gain of the proposed facilitator and a greedy facilitator, which takes no consideration of newcomers is compared. The greedy facilitator forms a community immediately with enough agents.

3) To examine the impact on the length of an agent's lifetime towards the complexity and quality of the proposed model, the matchmaking for a specific agent is analyzed.

For the evaluations, all experiments share the following parameter settings. The maximum value of lifetime is limited to 5. The utility table of each agent contains 5 possible categories. The utility value is ranged from 0 to 4. Newcomers arrive according to the Poisson distribution with a mean of 0.4. The probability of losing due to lifetime and the probability of community formation are both 0.1. The cost for waiting is 0.01 and the size of community is set to 4. The facilitator terminates matchmaking when global time step reaches 100.

For each experiment, the number of agents differs at initialization. The marketplace commences with 10 agents in experiment 1 and 3 agents in experiment 2, and 5 agents in experiment 3. Furthermore, for experiment 3 explicitly, we set the threshold value to 0.01 for reduction.

6.2 Empirical Results

This section analyzes experiment results via various graphs. First, the affects of the threshold value on the complexity and quality of the mechanism are examined.

Figure 5 depicts the number of possible utility tables required when calculating the expected utility under various threshold settings. The number of utility tables decline dramatically from threshold 0.01 to 0.015, showing that most utility tables have a small probability value.

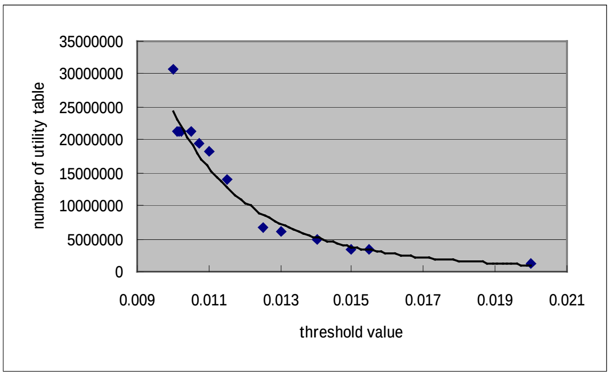

Figure 5 Complexity of the facilitator

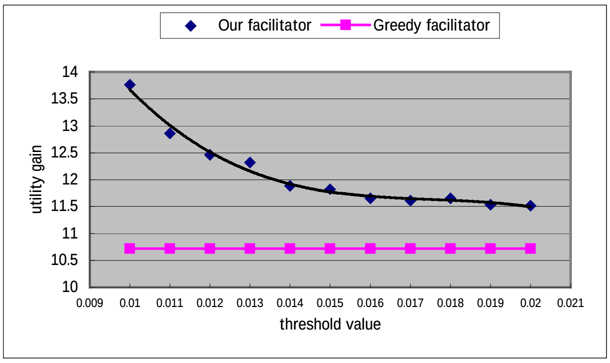

Figure 6 Quality of two facilitators

Figure 6 shows the difference between the proposed facilitator and the greedy facilitator in the average of utility gain of communities formed for 100 time steps under different threshold values. It clarifies that the facilitator outperforms the greedy one overall. Considering that the maximum utility gain for a community with four members from the parameter setting is 16, the proposed facilitator shows fairly good result where the threshold value is lower than 0.15.

To follow, one examines whether the lifetime of an agent has influence on its complexity and quality.

Figure 7 depicts the number of possible utility tables required in calculating the expected utility under different lifetimes. The graph clearly shows that the number of utility tables increase exponentially as lifetime increases.

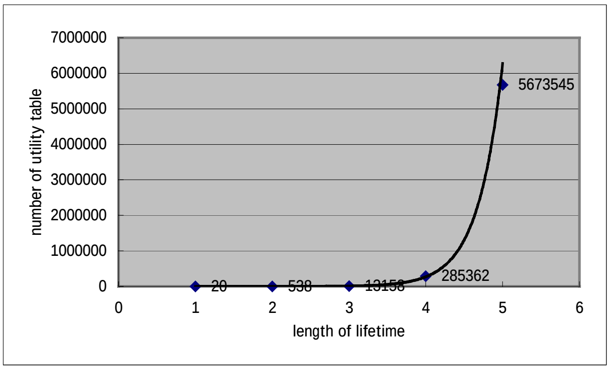

Figure 7 Complexity against agent's lifetime

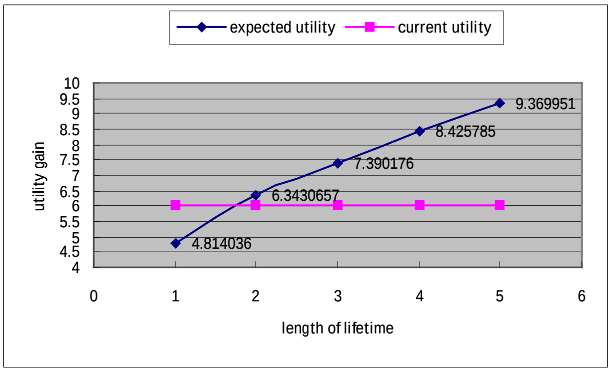

Figure 8 Quality against agent's lifetime

Figure 8 shows that the utility gain increases consecutively with lifetime. The utility gain will achieve final converge at 16.

7. Conclusion

In this paper we proposed a time-dependent group matchmaking mechanism with consideration of newcomers as the facilitator. In the model, all agents register themselves to the facilitator with a utility function and attributes for community formation. The facilitator filters and classifies agents according to the given information. The facilitator performs MDP matchmaking to form optimal communities.

The implementation and evaluation of this mechanism was also presented. Furthermore, there has also been discussion on the computational complexity encountered and the application of the threshold setting as a solution. In the simulation, the effectiveness of this mechanism by comparing the overall utility gain to a greedy facilitator is demonstrated. In addition, the complexity change against various threshold values has also been examined.

8. Reference

[1] Puterman, M. L. Markov decision processes: discrete stochastic dynamic programming. John Wiley and Sons, 1994.

[2] Choi, S and Liu, J. "A Dynamic Mechanism for Time-Constrained Trading." In Proceedings of Fifth International Conference on Autonomous Agents (Agents 2001), Montreal, Ontario, Canada. May 2001.

[3] Chang et al, "A Time-dependent group matchmaking mechanism for creation of agent-based communities." In Proc of 63rd IPSJ National Convention, 2E-04, 2001.

[4] Chang et al, "Implementation and Evaluation of A Time-dependent group matchmaking mechanism." To appear in Proc of 64th IPSJ National Convention, 5ZC-05, 2002.

[5] Ono et al. "Facilitator Agent Based on Word-of-mouth Trust." IPSJ Journal, p478-486, Vol 43 No. 2 Feb 2002.

[6] Gaurav et al. "Sustaining Individual Incentives while Maximizing Aggregate Social Welfare: A Mediated Brokering Technique for Trading Agents in Next-Generation Electronic Markets." In Proceedings of the International Conference on Internet Computing, IC'2001.