IMPLEMENTATION AND EVALUATION OF A TIME-DEPENDENT GROUP MATCHMAKING MECHANISM

5ZC－05

Lawrence Chang* Gen Hattori** Chihiro Ono** Satoshi Nishiyama**
Hidenori Nakazato*** Yoshiyori Urano***
*GITS, ***GITI, Waseda University **KDDI R&D Laboratories Inc.

1. Introduction

Agents join a marketplace to find desire agents in exchanging services. Like-minded agents form into communities within a deadline through the assistance of a facilitator. The facilitator performs matchmaking and ensures gratify utility is gained. Nevertheless, existing facilitators are inattentive to potential newcomers. A better utility can be obtained if newcomers are considered at matchmaking. We have proposed a facilitator that exploits a time-dependent group matchmaking mechanism for community formation in which potential newcomers are taken into account [1]. In this paper, we present an implementation and evaluation of this mechanism. We discuss on the computational complexity encountered and suggest a way to resolve it. In addition, we demonstrate the effectiveness of this mechanism by comparing to a conventional one.

2. Group Matchmaking Mechanism

In our model, the facilitator performs group matchmaking with consideration of newcomers. In a marketplace, agents register attribute and lifetime to the facilitator for community formation. Each agent has an individual utility table where utility values of other agents are stored. Based on registrations, the facilitator performs matchmaking by employing a Markov decision process model [2] for decision-making.

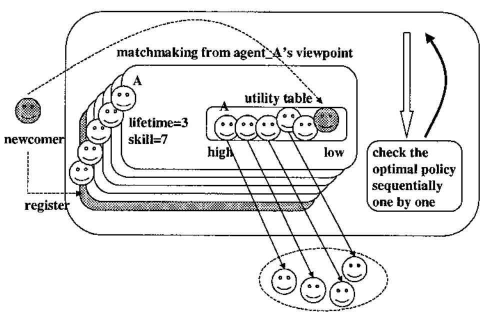

Figure 1 Matchmaking at facilitator's level

In Choi et al's work [3], a similar mechanism was proposed and following formulae were derived:

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

$$V^*_t(n) = \max(V_c(n),\ V_t(n)) \quad (2)$$

$$\pi^*_t(n) = \begin{cases} accept & if \quad V_c(n) > V_t(n) \\ wait & otherwise \end{cases} \quad (3)$$

At time step t, n denotes the current collection of agents and n' is the change in collection. Newcomer is represented by e and $n'+e$ is the collection available at next time step. The facilitator conforms to the optimal policy $\pi^*_t(n)$ for making decision on whether to accept or wait for better result in community formation by comparing the current utility $V_c(n)$ on hand and the expected utility $V_t(n)$ for waiting. The expected utility $V_t(n)$ consists of a cost function $c(t)$, a transition probability function $\Pr(n'+e)$ from current to the next time step, and an optimal value function $V^*_{t-1}(n'+e)$ at next time step. The elements of transition probability are the probability of newcomers and the probability of change in collection, which consists of the probability of losing due to lifetime and community formation.

Our 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 drive better quality.

For the group matchmaking, we introduce the following formula for current utility:

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

where m is the size of a community and $u_a(i)$ denotes the utility value of each agent in the current collection for the agent undergoes matchmaking (figure 1).

As for driving a better quality for the individual utility, the facilitator performs matchmaking sequentially for each agent based on agent's own utility table. In addition, each agent has individual probability of losing due to community formation, which can be used to calculate transition probability ($\Pr(n'+e)$) accurately.

3. Complexity Issue

3.1 Representation of Utility Table

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

However, this concept becomes inapplicable to our model. In our model, the transition probabilities for agents which have same utility value by a specific agent are different since the utility value of these agents by other agents may differ. Thus two agents in the same category should be treated differently, which causes the concept of category useless. Therefore in our model, agents in the utility table for a specific agent are treated separately during the calculation of expected utility for next time step, which increase the computational complexity from $O(\text{(number of categorie)}^N)$ to $O(\text{(number of average agents in the utility table)}^N)$. The reduction techniques for computation become more important than Choi et al's case. In this paper, we use category by regarding all agents having the same probability for simplicity.

3.2 Complexity and Reduction Technique

For calculating expected utility, the facilitator requires enumerating every collection of possible utility tables at next time step for each agent by considering newcomers and agents lost due to lifetime or community formation (figure 2). However, there exists unbounded of possible utility tables. 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).

So far, some reduction techniques for reducing the complexity have been suggested. In our paper, we apply the threshold setting to eliminate insignificant utility tables with small probability so that the number of utility table becomes bounded. The formula for transition probability is defined as:

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

where l is the probability of losing due to both lifetime and community in category j.

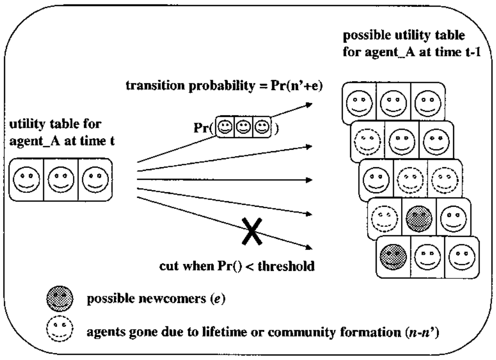

Figure 2 Threshold setting for complexity reduction

4. Evaluation

To demonstrate the effectiveness of our facilitator, we conduct two experiments:

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

2) We study the impact on threshold setting towards utility gain. In addition, we compare the utility gain of our facilitator against a greedy facilitator, which expresses no consideration of newcomers. The greedy facilitator forms a community immediately with enough agents.

4.1 Setting

For evaluations, we use following initial values: the marketplace initiates with 10 and 3 agents which has different utility table respectively. Each agent consists of an attribute and a lifetime; 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 probabilities of losing due to lifetime and 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.

4.2 Empirical Result

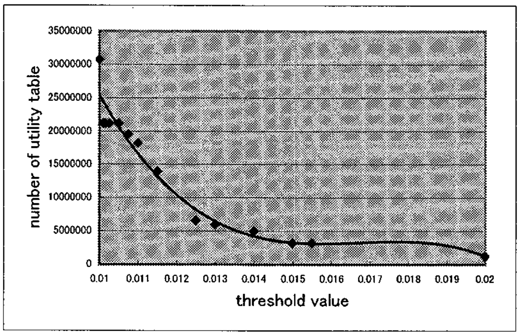

Figure 3 Complexity of the facilitator

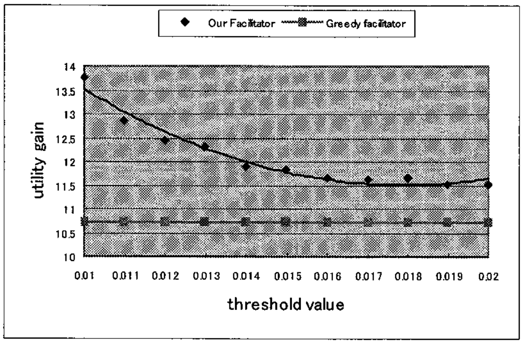

Figure 4 Performance of two facilitators

Figure 3 depicts the number of possible utility tables required for calculating expected utility under various threshold settings. The number of utility tables declines dramatically from 0.01 to 0.015 which shows that most utility tables have small probability value.

Figure 4 shows the difference between our 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 our facilitator overall outperforms the greedy one.

Although the utility gain shows a better result, it has not reached the optimal yet. In order to generate much better results, further reduction techniques are required. We believe that using individual probabilities can also improve in utility gain.

5. Conclusion

In this paper, we present an implementation and evaluation of the time-dependent group matchmaking mechanism. We conclude that the proposed mechanism is a feasible one through empirical results.

6. References

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

[2] Puterman, M. L. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley and Sons, 1994.

[3] Choi, S.P.M. and Liu, J. *"A Dynamic Mechanism for Time-Constrained Trading"*. In Proc. of Agents 2001, May.