# Competing in a Queue for Resource Allocations among Non-Cooperative Agents

**Pinata Winoto**[1] and **Tiffany Ya Tang**[2]

**Abstract.** In this paper, we investigate a multi-agent non-cooperative game for resource allocations based on an M/D/1 queuing model. Specifically, agents with common goals to maximize utility are deployed to compete with each other to bid or bribe for quicker service provided by the server. The bid/bribe of each agent in the queue is not revealed, but the outcomes, in terms of the pair of (bid/bribe; total waiting time), are publicly available from the server. Agents choose from one of three available strategies: random strategy, Nash equilibrium strategy and linear regression strategy, for their decision-makings. Bayesian update is integrated into the linear regression technique for searching an optimal bid/bribe. Besides, weighted average, second order autoregressive process (AR(2)), and random walk are utilized to predict service speed. After each agent obtained service, it re-evaluates its strategy and adjusts it accordingly. Results show that in the long run, the dominant strategy depends on the service speed. When the service speed is low, random strategy dominates the society. But if the service speed is high, linear regression strategy dominates. The model can be extended to study agent-based social simulations and decentralized scheduling for resource allocations in an open multi-agent system.

## 1. INTRODUCTION

It is common practice to use FIFO setting in a queue. However, when customers have different valuations of waiting time or urgencies, FIFO setting is not appropriate. Many modifications are available, such as, multi-priority/multi-class setting [2], and priority pricing setting [1, 4, 6, 8, 9]. This paper starts with the same issue, buying position in an M/D/1 queue by customers through legal/illegal activities. It is a non-cooperative repeated game, where each player is randomly selected to have the opportunity to enter the queue. Each player does not know how long the queue is, and he can choose whether to enter the queue by paying a non-revisable bid to the server or not enter the queue at all. The payoff depends on the waiting time spent in the queue and how much he has paid. The waiting time depends on the bid he has paid and those of other players have paid. The results show that in the long run, dominant strategy in the society depends on the server's service speed. For instance, random strategy (in terms of bid) may dominate the queue if the service speed is too low. But when the service speed is very high, a regression strategy is dominant. Two prospective applications of this model are agent-based social simulation and decentralized scheduling for resource allocation [10] for open multi-agent systems. In the context of social simulation, we study the equilibrium of bidding/bribing strategy used by customers in a queue. Customers (agents) could adopt illegitimate way (bribe) in the dirty competition for the service. Indeed, agents will act based upon normative rational behavior (in terms of von Neumann-Morgenstern Expected Utility) in their decision-makings [5, 11]. However, if we regard the bribe as a legitimate action, the system becomes a closed-bid auction for buying a position in a queue. This approach becomes decentralized scheduling [10], for resource allocation such as queuing for remote method/object invocation, printing, and so on in a multi-agent system.

## 2. THE SIMULATION

### 2.1 The M/D/1 queue

There is one queue with one server at the end of it. Customers come to the queue according to a Poisson process with a mean rate of $m$ customers per unit time. The server distributes a service that has constant value $P$, with service speed, $u$ service per unit time. Each customer has different value of waiting time $v$ generated from uniform cumulative distribution $A(v) = Av$, $v \in [0, v_1]$, and $v_1$ is common knowledge. When a customer comes to the queue, he can decide not to enter the queue, or pay a bid $x$ (non-revisable and non-refundable) to the server. Thus, if a customer must wait in the queue for $W$ unit time after giving bid $x$, then the net profit he gets is $P - x - vW$. When a new customer enters the queue, the server will reorder the queue according to the bids received, with highest bid be served first. The queue is assumed to exist for a long time.

Let $z = mPA$ and $r = m/u$. Then according to Lui[7], if $r > z / (1 + z)$ and all customer with $v \le zv_1 / r(1+z)$ join the queue and bid under the following equation:

$$x = \frac{1}{mA}\left(1 + z - \frac{rAv}{\{[1/(1+z)] + rAv\}^2} - \frac{1}{[1/(1+z)] + rAv}\right) \quad (1)$$

and customers with $v > zv_1 / r(1+z)$ do not join the queue, then these strategies form a Nash equilibrium, i.e., no customer will deviate from its strategy because it is the best to maximize his profit. Moreover, if $r \le z / (1 + z)$, and all customers bid:

$$x = \frac{1}{mA(1-r)} - \frac{vr}{m(1-r+rAv)^2} - \frac{1}{mA(1-r+rAv)} \quad (2)$$

[1] Department of Computer Science, University of Saskatchewan, Saskatoon, SK S7N 5A9, Canada. email: piw410@mail.usask.ca

[2] Department of Computer Science, University of Saskatchewan, Saskatoon, SK S7N 5A9, Canada. email: yat751@mail.usask.ca

then these strategies also form a Nash equilibrium.

Given these bidding strategies, Lui[7][1] argues that the server could maximize his income (sum of all bids received) by adjusting the service speed $u$ such that it is not too small or too big, i.e.,

$$\frac{1}{u^*} = \frac{PA}{1+mPA} \quad (3)$$

Intuitively, if the service speed is very fast, then most customers are willing to wait, and yet give few bids. As the service speed goes slower, most customers will not choose to wait, thus compete to buy better positions in the queue by raising their bids. But if the service speed goes too slow, some customers become desperate and will not join the queue at all, which will reduce the total bids.

The main drawback of this setting lies in that the decisions are made in the equilibrium state, i.e., the queue exists for a long time, and everyone knows the coming rate and the strategy used by others, the service speed is exactly $u^*$, and none of the customers knows the length of the queue. Two questions can be addressed in this case: What happens if some of the agents do not play Nash equilibrium strategy? And what happens if the service speed does not follow Nash strategy?

## 2.2 Experimental design

In addition to playing Nash equilibrium strategy, all agents could choose two other strategies: random and linear regression. In random strategy, an agent randomly decides whether to enter queue or not, and draws a random bid from uniform distribution [0, $X$]. The chance is set to $v/2v_1$ for not entering the queue, e.g., if $v = 4$ and $v_1 = 10$, then by 20% chance he will not enter the queue.

All agents can learn not only from their own experiences but also from those of others and use linear regression methods to predict their waiting time. They are capable of memorizing up to 100 pair values of bid and waiting time, and using up to 50 pairs for the regression. And then compute their optimal bid:

$$\underset{x}{Max}\, G = P - x - vW(x) \quad (4)$$

Where $W(x)$ is obtained from the linear regression:

$$W^{-1} = a + b\,x^2 + \varepsilon \quad (5)$$

A Least Squares method is used to solve it. And an agent will stay out of the queue if and only if the optimal $G < 0$. The only problem is whether the past data is trustworthy. For example, we cannot neglect the inconsistency of the server in setting his service speed. Thus, only some of the data may be useful in the regression while others become out of date or erroneous. Hence, at any time each agent holds a belief for the reliability of any pair of data and updates them through Bayesian update. Intuitively, if an agent runs regression from a set of data and the forecast made is close to the actual result, then he will still rely on them in the future. In our simulation we set the belief level to [0, 1] with initial belief equals to 0.5.

[1] In Lui[7], he studied the optimal service speed to maximize the bribes received in a queue. As we treat bribe as legitimate payment (bid), the whole system becomes auction.

Let the difference between the inverse expected waiting time from regression with the inverse actual waiting time be $\Delta W_t^{-1}$, the prior belief is $b_{t-1}$, and $s$ is the standard error of the regression then we define the posterior belief as

$$\begin{aligned} b_t &= 1 - (1 - b_{t-1})\,(\Delta W_t^{-1}/s) && \text{if } \Delta W_t^{-1}/s \le 1 \\ b_t &= b_{t-1}\,(s / \Delta W_t^{-1}) && \text{if } \Delta W_t^{-1}/s > 1 \end{aligned} \quad (6)$$

Agents will select up to 50 data where the data with higher belief is more likely to be selected.

Agents who adopted this strategy may also rely on the service speed. If the service speed is very volatile then the accuracy of the present data is low; thus some adjustments are needed. There are many methods to predict future service speed, but three methods are adopted in our experiment: weighted average, random walk and second order autoregressive (AR(2)). AR (2) is defined as

$$u_t = \mu + \gamma_1 u_{t-1} + \gamma_2 u_{t-2} + \varepsilon_t \quad (7)$$

Where $\mu$ is the constant coefficient of regression, $\gamma_1$ and $\gamma_2$ are coefficients for lagged (past) service speeds, and $\varepsilon_t$ is residual of regression.

After an agent collects the time series data for service speed from AR (2), he will predict future service speed, and adjust current optimal bid accordingly. A trend to slower service speed indicates that the server tries to increase his total revenue by inducing higher $x$ as $W(x)$ increases. Hence a higher optimal bid is needed in the future in order to offset the increasing waiting time. The procedure for adjustment is given as follow:

$$x^*_{\text{new}} = [1 + (u_t / u_{t+W^*} - 1)/2]\; x^*_{\text{old}} \quad (8)$$

But to avoid winner's curse, he may reject to respond or even reduce the optimal bid by probability $p_d$, i.e., he will reduce his optimal bid randomly from 0% to 10%. The initial $p_d$ for any forecasting method is randomly given from 0 to 1, and the next $p_d$ is updated according to equation (13) described later. The same procedure applies for decreasing service speed. If the service speed stays unchanged, then agent does not need to adjust his optimal bid.

The adjustment explained above is valid for AR(2) model, because both random walk and weighted average can not be used *directly* to forecast future trend. In later cases, the following additional computations are needed.

If an agent uses weighted average (WA), then the difference between recent average speed and past average speed will determine the trend of future service speed. In our setting, the number of samples for averaging is 10, and the lagged is also 10. The weights are in the order from 1 to the sample size. Thus, the difference is:

$$\Delta u = \frac{\sum_{i=0}^{9} (10-i)\,u_i}{55} - \frac{\sum_{i=10}^{19} (20-i)\,u_i}{55} \quad (9)$$

Where i represents the lagged from present, e.g., i = 0 represent current value, i = 1 represents 1-lagged value, etc. A positive $\Delta u$ represents an increasing service speed. The rest of adjustment process is the same as those in AR(2), except that $1/u_{t+W*} = (W*/\Delta u) / u_t$.

If an agent uses random walk, then the difference is always zero, that is, no adjustment is needed, because the agent believes that the expected value of future service speed equals to current one.

Each agent can choose one out of three strategies {Nash, Random (ZI), Linear Regression (LR)}. If he chooses Nash or Linear Regression, then he needs to choose a method to predict service speed, i.e., {WA(10, 10), AR (2)-30, Random walk (RW)}. In all, 7 different strategies are provided. For the first few periods, all agents can use random strategy (ZI) or (Nash, RW). After certain periods, they are free to choose other strategies and rank these strategies based on their experiences and preferences. All agents iteratively compute the most preferred method to predict the service speed. And those agents who received services will rank the strategies for bidding. If the strategy produces a "satisfactory" outcome, i.e., at least as good as the expected one, then its rank will increase, otherwise decrease. The rating is from 0 to infinite, with random initial rate $\in [75, 125]$. The new rating is given by,

$$\textit{New rating} = \textit{Satisfaction} + \textit{Old rating} \quad (10)$$

where

$$\textit{Satisfaction} = 1 + \lfloor 5\eta \rfloor$$
$$\eta = (\text{Actual Gain} - \text{Expected Gain}) / \text{Expected Gain} \quad (11)$$

to rank the strategies. Or

$$\textit{Satisfaction} = 1 + \lfloor 50(0.2 - \varphi) \rfloor$$
$$\varphi = |\ \text{Actual value} - \text{Forecast value}\ | / \text{Range of value} \quad (12)$$

to rank the forecasting methods.

Moreover, $p_d$ is updated by:

$$p_d^{new} = (1.2 - \varphi)\, p_d^{old} \quad (13)$$

Whenever an agent is selected to enter a queue, he will choose the strategy with the highest rating. If there is more than one strategy with highest rating, he will choose one of them randomly.

## 2.3 Risk of being penalized

In previous setting we assume that customers involve in a legitimate auction, therefore we do not impose any risks to both the bidder and server. But as described before, this simulation can be extended easily for simulation of bribery behavior. Thus, we add the deterrence effect in our model, e.g., the probability that any briber and bribee may be arrested will increase with regards to an increasing bribe. And if they are arrested, then both will be penalized to pay certain amount of fine $F$, which is assumed to be constant.[2] If we assume that both the customer and the server are risk neutral, i.e., their decisions are based on von Neumann - Morgenstern expected utility, then the server could determine the optimal service speed $u^*$ such that maximizing his total expected revenue from all customers, i.e.,

$$\arg\max_u \sum_i \left[ \left(1 - p_a(x_i)\right) x_i - p_a(x_i)\, F \right] \quad (14)$$
$$\text{subject to: queue length} \le \text{N}$$

Where $p_a(x)$ is the probability of apprehension, $x_i = x(u)$ is the bribe received from customer-$i$ and $F$ is the fine paid if arrested, and $p_a(0) = 0$ and $p_a(P) = 1$.

Customer-$i$ determines the optimal bribe using a similar formula, i.e.,

$$\arg\max_X \left[ (1 - p_a(x))G(x) - p_a(x)F \right] \quad (15)$$

Where $G(x)$ is the expected net profit. If the value inside the bracket is less than zero, then the customer will not join the queue. It may occur even when the expected gain $G(x)$ is large.

Under this setting, there may exist many corner solutions to customer problem such that $x(v) = 0$ for more than one $v$. But if more than one customer does not bribe, one of them can take advantage by giving an infinitely small bribe. The only difference is to use equation (15) rather than equation (4).

# 3. RESULTS AND DISCUSSIONS

A simulator is designed to construct thousand agents, each of whom is capable of memorizing 100 sets of data for service speed, 100 pairs for (bid, waiting time) and its corresponding belief value, the ratings for all 7 strategies, and several other parameters. Newton's method and Simulated Annealing method are used in solving optimization problem. We choose $p_a(x) = x/P$, $x \in [0, P]$ for the probability of apprehension. This function is not fully supported by empirical evidence; instead, we choose it for the purpose of simulation only.

Figure 1 shows a kink increasing relationship between service time (inverse of service speed or time per unit of service) and average social loss. Social loss is the total loss due to waiting in the queue. Intuitively, longer service time causes the queue to grow faster; therefore social loss rises faster. But whenever the average number of incoming customers (incoming rate) is less than the service speed, the length of the queue is always near zero. Therefore the curve is flat near the origin. But when the service speed is smaller than the incoming rate, then the length of the queue grows faster, which increases the waiting time and social loss. This phenomenon is shown as the steep curve.

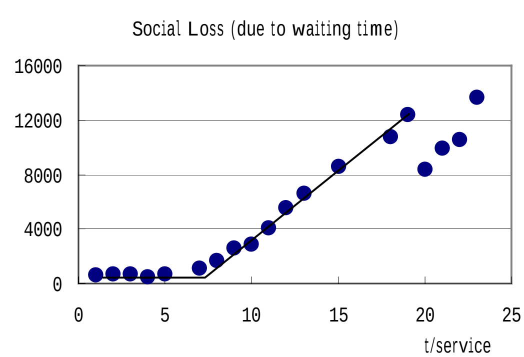


**Figure 1.** Social Loss

[2] One may think to use an increasing fine with regards to an increasing bribe given/received as a more realistic setting. However, we choose a constant fine for two reasons: firstly, make the model simple and easy to be studied; and secondly, follow the result of crime studies that increasing fine is less important compared to increasing the probability of apprehension in deterring crime [5].

Figure 2 and 3 show the participation rate and server's total revenue. As more people are reluctant to join the queue if the expected waiting time becomes longer, the participation rate decreases in terms of service time. Figure 3 shows that the server's total revenue is a decreasing function in terms of service time.

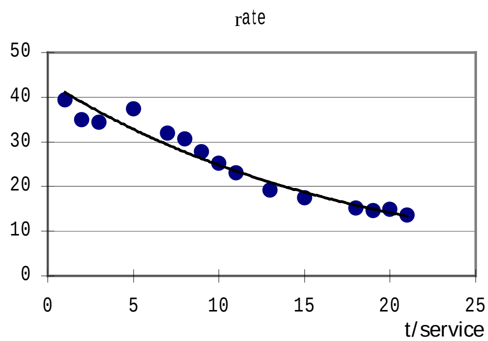


**Figure 2.** Participation Rate

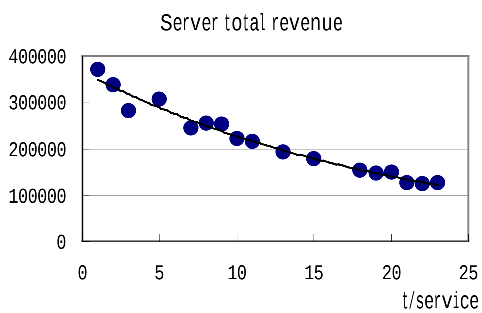


**Figure 3.** Server's Revenue

Figure 4 and 5 show the dynamic of the dominant strategies of customers who join the queue (in 20 moving average trend line). The vertical axis represents the fraction of customers using one of three available strategies. The horizontal axis represents the periods. Figure 4 is the typical pattern when the service speed is high. And figure 5 is the typical pattern when the service speed is low. Figure 4 shows the dominance of Linear Regression in the long run, while figure 5 shows the dominance of random strategy in the long run. In both cases, Nash strategy is dominated. Figure 6 and 7 show the actual gain made by the same agents in figure 4 and 5. From those 4 figures, we can conclude that using Linear Regression strategy could better predict the expected gain in high-service-speed-queue (figure 4 and 6). Higher accuracy of linear regression prevents agents from joining the low-service-speed-queue (figure 5 and 7), which facilitates the dominance of random strategy.

Figure 8 shows the situation when the service speed is volatile. The result is roughly the mix of figure 4 and 5. This unstable service speed also leads to an increase of the social loss, approximately 5600 compared to average social loss in stable service speed (approximately 4700). The main reason is that an increase of uncertainty raises error in prediction, which in turn raises social loss.

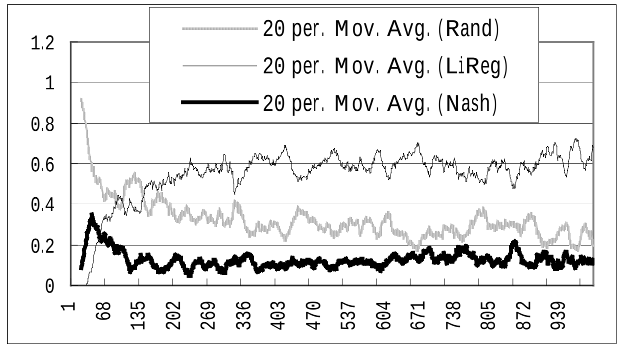


**Figure 4.** Strategies in High-Service-Speed-Queue

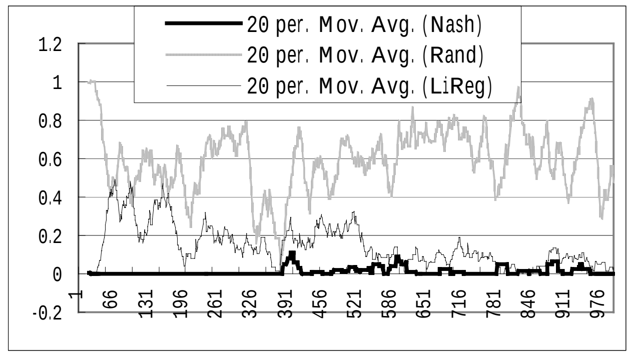


**Figure 5.** Strategies in Low-Service-Speed-Queue

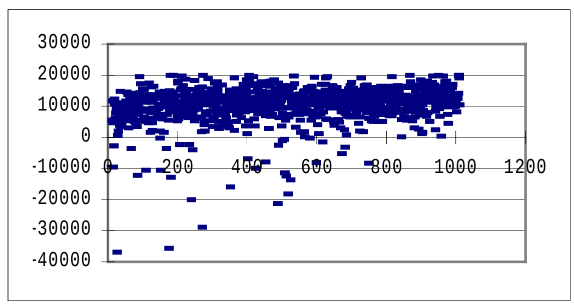


**Figure 6.** Average Gain by Customers in High-Service-Speed-Queue

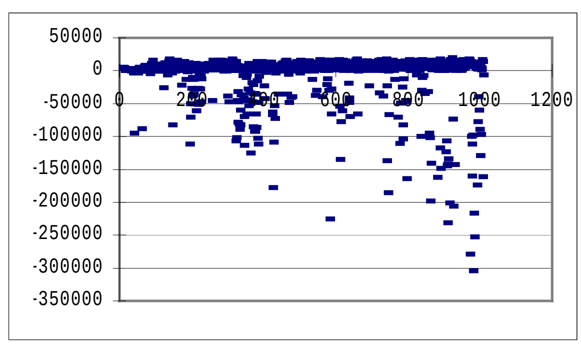


**Figure 7.** Average Gain by Customers in Low-Service-Speed-Queue

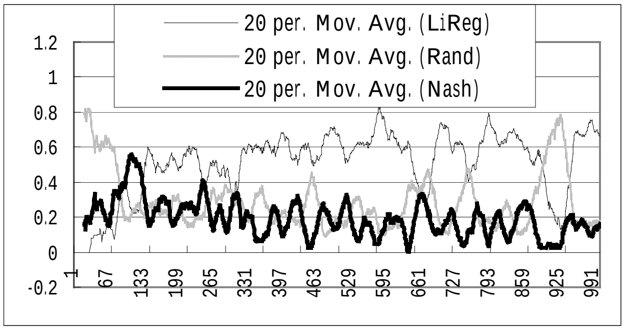


**Figure 8.** Strategies in Random/Volatile-Service-Speed-Queue

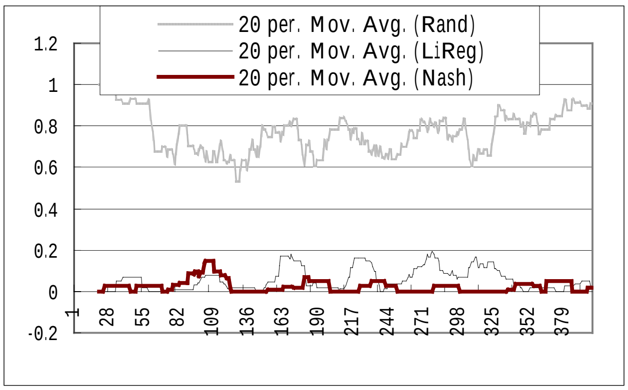


**Figure 9.** Strategies in the Presence of Punishment

Figure 9 shows how the strategy changes when the risk of punishment is introduced into the system. Just like when the service speed is very low, many of 'good' predictors avoid giving bribe and thus will not join the queue, leaving random strategy dominate the queue.

## 4. CONCLUDING REMARKS

As stated before, the flexibility and extendibility of this model is one of the main contributions of this work. On one hand, it tries to enrich the agent-based social simulation by proposing experiment in queuing model where agents could take legitimate/illegitimate action. On the other hand, it studies the resource allocation in queuing model through priority pricing mechanism. Generally, three main conclusions could be derived from the study:

- In most cases when not all parties follow Nash equilibrium strategy, Nash equilibrium strategy is not a dominant strategy as shown in the experiment. It is dominated by linear-regression strategy (with various learning processes) and random strategy.
- Lower service speed will have negative impact on the revenue of both server and customers (see fig. 1 and fig. 3).
- Holding service speed constant will reduce the social cost; and making the bribe legal (as non-refundable auction) will increase participation rate and the social welfare.

In spite of it, more works needs to be conducted, among them:

- Extending the current model to many other types of single-server-queue, or to multiple-server-queue system.
- Implementing more complex strategies in agents' decision process.

## AKNOWLEDGEMENTS

We would like to thank Paul Brewer, Francis Lui, Gordon McCalla, Julita Vassileva and two anonymous reviewers for their comments and supports. This work is funded through research assistantships from the Canadian Natural Sciences and Engineering Research Council and postgraduate studentships from Hong Kong University of Science and Technology and Hong Kong Polytechnic University.

## REFERENCES

[1] Anderson, M.Q. Optimal Admission Pricing and Service Rate Control of an M[x]/M/s Queue with Reneging. Naval Research Logistics Quaterly, 30, 1983, 261-270.

[2] Babad, J.M. Joining Policies in a Multipriority Multiclass Batch Computer System. Comm. of the ACM, 19(3), March 1976, 127-136.

[3] Conte, R.; Hegselmann, R.; Terna P. (eds.). Simulating Social Phenomena. Springer, Berlin, 1997 (entire issue).

[4] Kleinrock, L. Optimum Bribing for Queuing Position. Operations Research, 15, 1967, 304-418.

[5] Leung, S. F. Dynamic Deterrence Theory. Economica, 62(245), 1995, 65-87.

[6] Low, D. Optimal Dynamic Pricing Policies for an M/M/s Queue. Operations Research, 22, 1974, 545-561.

[7] Lui, F. T. An Equilibrium Queuing Model of Bribery. Journal of Political Economy, 93(4), 1985, 760-781.

[8] Mendelson, H. and Whang, S. Optimal Incentive-Compatible Priority Pricing for the M/M/1 Queue. Operations Research, 38, 1990, 870-883.

[9] Rao, S. and Peterson, E.R. Optimal Pricing of Priority Services. Operations Research, 46, 1998, 46-56.

[10] Wellman, M.P., Walsh, W.E, Wurman,P.R., Mac-Kie,J.K. Auction protocols for decentralized scheduling. In Proceedings of the 18th International conference in Distributed Computing Systems, 1998.

[11] Winoto, P. and Tang, T.Y. Evaluating the stochastic properties of agent society from economics' perspective. Proceedings of the 2001 IEEE Systems, Man and Cybernetics Conference, vol. 5, 2905-2910.