Dynamic Transport Services Using Flexible Positioning of Bus Stations

Naoto Mukai
Department of Systems and Social Informatics,
Graduate School of Information Science, Nagoya University
Furo-cho, Chikusa-ku, Nagoya, 464-8603, Japan
naoto@watanabe.ss.is.nagoya-u.ac.jp

Toyohide Watanabe
Department of Systems and Social Informatics,
Graduate School of Information Science, Nagoya University
Furo-cho, Chikusa-ku, Nagoya, 464-8603, Japan
watanabe@watanabe.is.nagoya-u.ac.jp

Abstract

In these years, a dynamic transport service which is called demand-bus system by using the contemporary real-time location technologies is focused by public administrations. The most important feature of the services is on-demand processing; i.e., customers can call transport vehicles anytime and get on/off anywhere. However, in the existing circumstances, the demand-bus system is still on the developing stage. In this paper, we classify the demand-bus system into two types: semi-demand and full-demand, on the basis of bus station types. Moreover, we propose methods to solve the real-time problems, (i.e., assignment and scheduling problems), in the demand-bus system by using anticipated future routes of transport vehicles. Finally, we report simulation results and consider effectiveness and possibility of the demand-bus system in an urban area.

1. Introduction

In these years, there's been an increase of research in intelligent transportation system (ITS) [8]. New real-time location technologies such as global positioning systems and wireless telecommunication systems enables to gain quickly access to positions of mobile objects. It seems that his progresses will bring about a change in transportation services. One of the new transportation services is a demand-bus system [9, 6, 7, 5]. The most important feature of the system is on-demand processing; i.e., customers can call transport vehicles anytime and get on/off anywhere. In the system, real-time decisions is required to quickly process fleet management, traffic assignment, and others.

However, in the existing circumstances, the demand-bus system is still on the developing stage. In particular, the most of existing demand-bus systems are employed in the limited provinces (i.e., small transport areas and a small number of customers). Moreover, these are static transport services because they incorporate pre-reservations in advance of departures from a depot on the basis of timetables (i.e., customers have to contact transport companies, and appoint riding and dropping places on the preceding day). Therefore, our paper contributes the realization of the dynamic demand-bus system in an urban area (i.e., large transport areas and a large number of customers).

In this paper, we classify the demand-bus system into two types: semi-demand and full-demand, on the basis of bus station types. In the semi-demand type, there is a number of stations where customers get on/off transport vehicles in an urban city. The key issue in the semi-demand type is selection of the positions and number of the stations in consideration for characteristics of the city (e.g., population, density and transportation). On the other hand, in the full-demand type, transport vehicles are independent of the stations. In other words, the positions where customers get on/off are not fixed. The key issue in the full-demand type is selection of ride-on and drop-off positions in consideration for the trade-off relation between availability for customers and profitability for transport companies. Moreover, we propose methods to solve two real-time problems: "which transport vehicles are assigned to customers?" and "how delivery orders for customers are scheduled?". The method for assignment problem is based on anticipated future routes of transport vehicles which are stored in a communication server. The anticipated future routes are approximated by a number of straight lines to reduce communica-

0-7803-8963-8/05/$20.00 ©2005 IEEE.

tion costs between vehicles and the server. The method for scheduling problem is based on either taxi concept or share-ride concept.

The remainder of this paper is organized as follows: Section 2 defines the problem in our work. In Section 3, we classify the demand-bus system into two types: semi-demand and full-demand. Section 4 provides assignment and scheduling methods to solve the dynamic demand-bus problem. In Section 5, we report simulation results and consider effectiveness and possibility of the demand-bus in urban city.

2. Problem Setting

The problem in which a fleet of vehicles is routed in order to visit distributed customers is called Vehicle Routing Problem (VRP). Depending on the types of demands (i.e., pre-reservation type or on-demand type), the following two categories exist: Static Vehicle Routing Problem (SVRP) [1, 12, 13, 10, 4, 3] and Dynamic Vehicle Routing Problem (DVRP) [2, 11]. The most of existing demand-bus systems belong to the first category. On the other hand, our work belongs to the latter category. In this section, we formulate the problem of our work.

2.1. Road Network

A road network is based on the concept of graph G which consists of nodes N and edges E as Equation (1). For simplicity, we assume that each transport vehicle is located on any of nodes.

$$\begin{aligned} G &= (N, E) \\ N &:= \{n_1, n_2, \cdots\} \\ E &= \{e_i = (n_i - n_i') : n_i, n_i' \in N\} \end{aligned} \tag{1}$$

2.2. Customers

A customer is given by c who wants to travel from a certain node to other node as Equation (2). The position of a customer at time t is denoted as $c(t)$ which is a pair of x and y coordinates. We assume that each customer walks at a fixed speed $|\vec{c}|$. The demand of a customer is denoted as $D(c)$ which contains start node n_s, ride-on node n_r, drop-off node n_d and goal node n_g as Equation (3).

$$\begin{gathered} C = \{c_1, c_2, \cdots, c_n\} \\ c(t) = (c_x(t), c_y(t), |\vec{c}|) \end{gathered} \tag{2}$$

$$\begin{gathered} D(c) = (n_s, n_r, n_d, n_g) \\ n_s, n_r, n_d, n_g \in N \end{gathered} \tag{3}$$

2.3. Vehicles

A transport vehicle is given by a moving point v which transits on the road network. The position of a vehicle at time t is denoted as $v(t)$ which is a pair of x and y coordinates. The velocity vector of a vehicle at time t is denoted as $\vec{v}(t)$ which is a pair of speeds in x and y directions.

$$\begin{gathered} V = \{v_1, v_2, \cdots, v_k\} \\ v(t) = (v_x(t), v_y(t)) \\ \vec{v}(t) = (\bar{v}_x(t), \bar{v}_y(t)) \end{gathered} \tag{4}$$

Each vehicle has a time-parameterized queue $Q(v,t)$ which shows delivery orders for customers at time t in Figure 1. We also denote the anticipated future traveling distance of a vehicle at time t as $|Q(v,t)|$. The ride-on nodes and drop-off nodes of customers are inserted into the queue, and satisfied in the order of the queue.

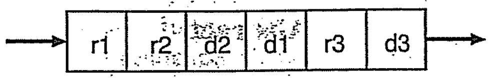


Figure 1. Time-parameterized queue

2.4. Objective

Here, let's consider objectives of our work. Generally, there is a complicate trade-off relation between usability for customers and profitability for transport companies. For example, transport vehicles have to visit customers one by one in order to minimize time intervals of customers from getting-on to getting-off. However, it causes the increase of traveling distance for vehicles (i.e., fuel cost). In this paper, for simplicity we define the time sequence of customer which is composed of walking time $T_{walk}(c)$, waiting time $T_{wait}(c)$ and riding time $T_{ride}(c)$, shown in Figure 2. We regard to minimize the interval of the time sequence as usability for customers, and the traveling distance of vehicles as profitability for transport companies. Consequently, the objective function of our problem is defined as Equation (5).

$$min(\sum_{c \in C}(T_{walk}(c) + T_{wait}(c) + T_{ride}(c)) + \sum_{v \in V}|Q(v,t)|) \tag{5}$$

3. Demand-Bus System

In this section, the classification of the demand-bus system is described. The way for the classification depends on types of bus stations, i.e., the positions where customers get on and off the transport vehicles.

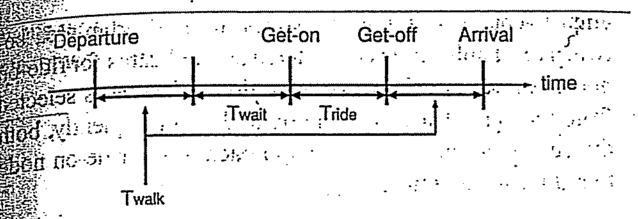


Figure 2. Time sequence of a customer

3.1. Semi-Demand

In the semi-demand type, there is a number of stations where customers get on and off in an urban city. The bus stations are corresponded to a subset of nodes in the road network as Equation (6).

$$S = \{s_1, s_2, \cdots, s_m\} \qquad (6)$$
$$S \in N$$

Consequently, the demand of a customer in semi-demand type is given by Equation (7). Each customer walks from start node n_s to ride-on node s_r to get on a vehicle. Then, the vehicle transports him from ride-on node s_r to drop-off node s_d. Finally, the customer gets off the vehicle and walks from drop-off node s_d to goal node n_g.

$$D(c) = (n_s, s_r, s_d, n_g) \qquad (7)$$
$$n_s, n_g \in N, \quad s_r, s_d \in S$$

3.2. Full-Demand

In the full-demand type, the positions where customers get on and off are not fixed. Therefore, the full-demand type enables more flexible transport services than the semi-demand type. However, there is a key problem: "where customers should get on and off?". The simplest method for the problem is that transport vehicles visit start node n_s and goal node n_g as Equation (8). Customers should just wait for arrival of a transport vehicle at the start node.

$$D(c) = (n_s, n_s, n_g, n_g) \qquad (8)$$
$$n_s, n_g \in N$$

However, customers can move by walk until the transport vehicle arrives. Thus, the selection of appropriate ride-on and drop-off nodes from the road network may increase overall performance of the transport system. The alternative method for the problem is that both a customer and a vehicle go to a meeting node (i.e., ride-on node n_r), then the vehicle transports the customer to a separating point (i.e., drop-off node n_d), and finally the customer walks to the goal node n_g. However, it is difficult to find efficient meeting and separating points without time-wasting heuristic searches. Thus, in this paper we only consider meeting points as Equation 9. Moreover, we propose a simple algorithm to find meeting point using anticipated future routes of transport vehicles in the next section.

$$D(c) = (n_s, n_r, n_g, n_g) \qquad (9)$$
$$n_s, n_r, n_g \in N$$

4. Assigning and Scheduling

In this section, we propose methods for the assignment and the scheduling problems to realize the demand-bus system.

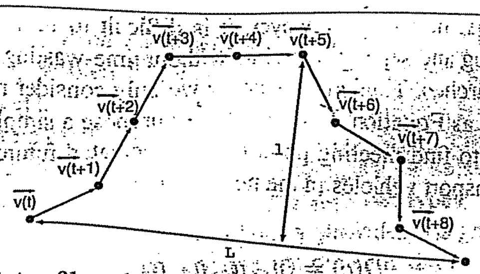
4.1. Assigning

The assignment method provides an answer to the question "which vehicles transport the customers?". The assignment method is based on the table of anticipated future routes which are stored in a communication server. Thus, transport vehicles have to send their anticipated future routes whenever new customers are assigned to each vehicle and anticipated future routes are changed.

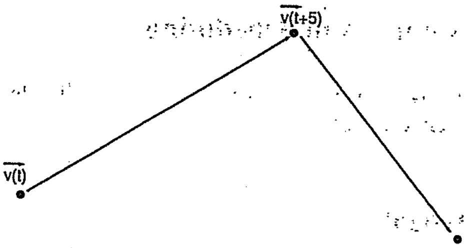
Here, we have to take account of communication cost between a vehicle and a server. In fact, the identification process for exact locations of vehicles needs a large amount of communication messages. In order to control the number of the communication messages, we introduce a straight line approximation algorithm for anticipated future routes. The algorithm is a recursive function according to the rate of length between end-nodes of the route and other nodes. First, we calculate the rates of each node. Next, if the maximum rate is less than a threshold ω, then the route is replaced by one straight road segment. Otherwise, the function is re-invoked to two sub-routes which contain the maximum node as end-node of itself. For example, in Figure 3 the rates of length l/L between end-nodes and other nodes are calculated, and the node with maximum rate is selected as shown in (a). The recursive process is repeated until all of the rates are no more than the threshold ω, and results in approximated future route shown in (b). Needless to say, the degree of approximation depends on the threshold ω, so that we must determine the threshold ω in consideration of communication infrastructure (e.g. number of transport vehicles, communication band, and processing performance).

In the semi-demand type, the assignment method is very simple. A nearest neighbor station by a customer is selected as a ride-on node. And, the customer is assigned to the selected nearest vehicle by the ride-on node.

In the full-demand type, first, each customer is assigned to a vehicle according to the following process. Here, we



(a) Rate of length between end-nodes and other nodes



(b) Approximated future routes

Figure 3. Straight line approximation

define a reachable circle of a customer or a vehicle time t later as a time-parameterized circle with spread speed $|\vec{c}|$ or $|\vec{v}|$ shown in Figure 5. A node on anticipated future route, which is overlapped the reachable circle with $t = t + 1$ of a customer, is searched. If customers assigned to a vehicle are empty, a reachable circle of vehicles is searched instead of the anticipated future route shown in Figure 5. This process is repeated until the overlapped reachable circle or node is found.

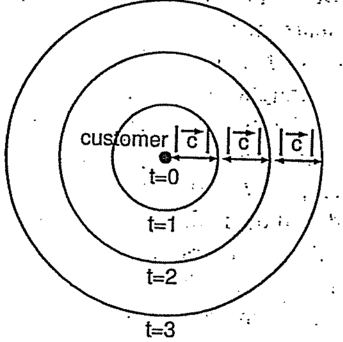


Figure 4. Reachable circle of a customer

Next, a ride-on node is selected on the basis of a positional relation between the assigned customer and the overlapped reachable circle or node. There are three cases for selecting ride-on node. The first case is that the overlapped reachable circle is found (i.e., the customer is assigned to empty vehicle) in Figure 5. Nodes in the overlap area between reachable circles are filtered as candidates for ride-on node. A node which is closest to drop-off node is selected from the candidates as a ride-on node. Consequently, both the customer and the vehicle approach to the ride-on node to reduce waiting time T_{wait}.

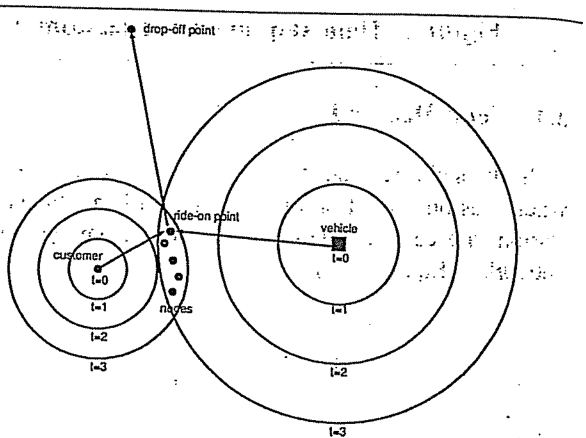


Figure 5. Case 1: empty vehicle

The second case is that the overlapped node is found and the customer can arrive at the overlapped node earlier than the vehicle in Figure 6. In this case, the route of the vehicle is not changed, but the customer has to wait for arrival of the vehicle on the overlapped node as a ride-on node.

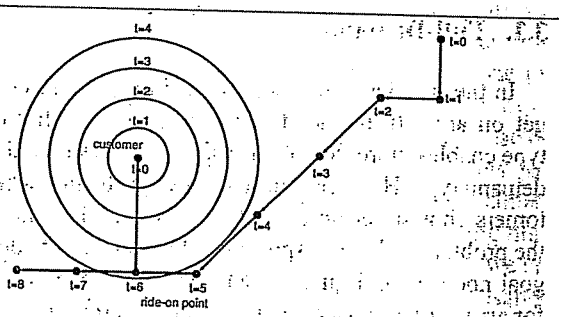


Figure 6. Case 2: customer can arrive earlier than vehicle

The third case is that the overlapped node is found and the customer cannot arrive at the overlapped node earlier than the vehicle in Figure 7. In this case, the route of the vehicle is changed to visit ride-on node for the customer. The selection method is the almost same as the first case except that nodes are filtered in overlap area between reachable circle of the customer and the reachable circle at a previous visiting node (i.e., ride-on or drop-off node for other customer).

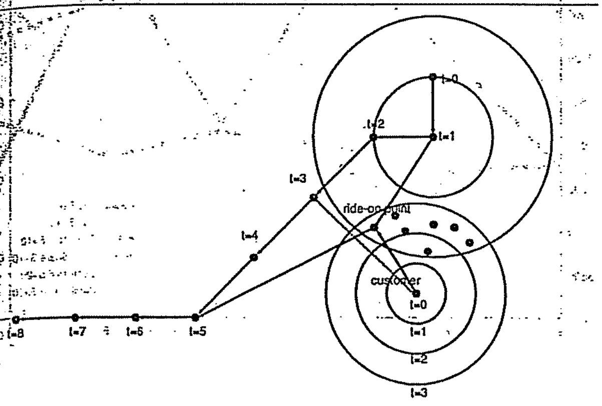


Figure 7. Case 3: customer cannot arrive earlier than vehicle

The influences of the three cases on customers are summarized in Table 1 and 2. Table 1 shows the influences on the assigned customer. In case 1 and 2, both the customer and the vehicle approach to the ride-on node so that waiting time T_{wait} decreases. On the other hand, in either case, walking time T_{walk} increases. Thus, case 2 is the most overburdened assignment for the assigned customer. Table 2 shows the influences on riding customers before assignment for the assigned customer. There is no riding customers in case 1. In case 2, the route of the vehicle does not changed so that the assignment has no influence on riding customers. On the other hand, in case 3, the route of the vehicle changes for the assigned customer so that riding time T_{ride} increases. Thus, riding customers may feel undesirable in case 3.

In this paper, we did not consider the degree of satisfaction. However, we must classify the weight of customer preferences and change behaves of vehicles for each customer. For example, if the customer is an aged person and has difficulty walking, the vehicle should go to his home (start node n_s) although riding customers may feel undesirable.

Case	T_{wait}	T_{ride}	T_{walk}
1	↓	—	↑
2	—	—	↑
3	↓	—	↑

Table 1. Influence on assigned customers

Case	T_{wait}	T_{ride}	T_{walk}
1	—	—	—
2	—	—	—
3	—	↑	—

Table 2. Influence on riding customers

4.2. Scheduling

A scheduling method provides an answer to the question "how delivery order for customers is scheduled?". We introduce two concepts for this question: taxi and share-ride concepts. Both ride-on node n_r and drop-off node n_d of assigned customers are inserted to the time-parameterized queue on the basis of the concept. The delivery order of the queue directly affects waiting time T_{wait} and riding time T_{ride}. Moreover, there is a trade-off relation between waiting time T_{wait} and riding time T_{ride}.

Taxi Concept Vehicles try to transport customers one by one.

Share-Ride Concept Vehicles try to reduce traveling distance whenever possible. As a result, there is a possibility that vehicle takes circuitous routes for a certain customer.

5. Experiments

This section reports the results of experiments which compare six patterns shown in Table 3. We evaluated six criteria: rate of arrival customers, traveling distance, waiting time, riding time, total time and walking distance. In this experiment, full-demand type transport system incorporates assignment method in case 1 so that the implementation of other cases is our future work. We set our experimental environment as follows. The road network of 1000 × 1000 pixels, which contains 986 nodes and 1165 edges, is a part of Nagoya city in Japan. The stations are selected from the road network randomly. The delivery demands are generated by 10% in each time. The start and goal nodes of each demand are selected from the road network randomly. The delivery process is repeated until max time T_{max}. Other parameters are shown in Table 4.

Figure 8 and Figure 9 show the average rate of arrival customers and the average of traveling distance for transport vehicles. From these results, share-ride concept can increase the rate of arrival customers and decrease the traveling distance, also these trends are especially prominent in a small number of vehicles. These trends indicate that our taxi concept in an urban city needs a more number of transport vehicles than share-ride concept. Moreover, the full-demand type shows lower rate of arrival customers than the

Pattern	Type	Concept	Number of Stations
1	full	taxi	-
2	full	share-ride	-
3	semi	taxi	10
4	semi	share-ride	10
5	semi	taxi	20
6	semi	share-ride	20

Table 3. Experimental patterns

k	5-10
$\|\vec{v}\|$	20 pixels
$\|\vec{c}\|$	4 pixels
T_{max}	20000
ω,	0.2

Table 4. Parameter setting

semi-demand type. The reason is that the load-balancing for delivery among vehicles is difficult because of their flexible traveling routes. For the same reason, the semi-demand type (20 stations) indicates more similar results to the full-demand type than the semi-demand type (10 stations).

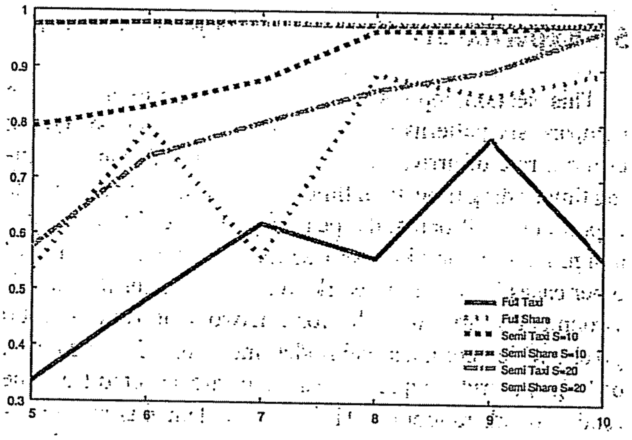


Figure 8. Average rate of arrival customers

Figure 10 and Figure 11 show the average of waiting time and the average of riding time for customers. We can see that there is a trade-off relation between waiting and riding time. In the experimental patterns, patterns 3 and 5 (semi-demand, taxi concept) show good results compared with other patterns at the first view. However, this trend implies the increasing of walking distance from start node n_s to ride-on node n_r and from drop-off node n_d to goal node n_g.

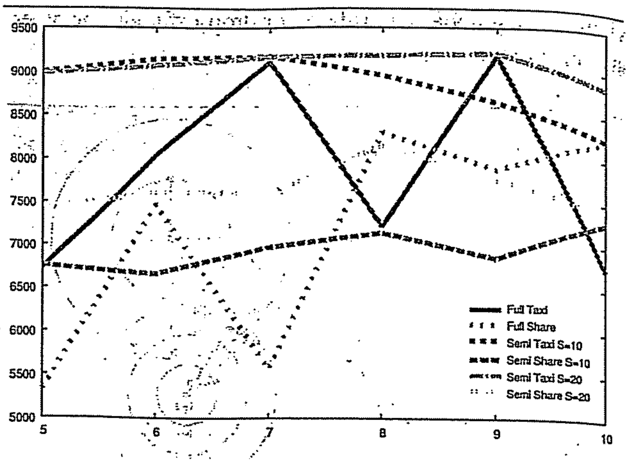


Figure 9. Average of traveling distance

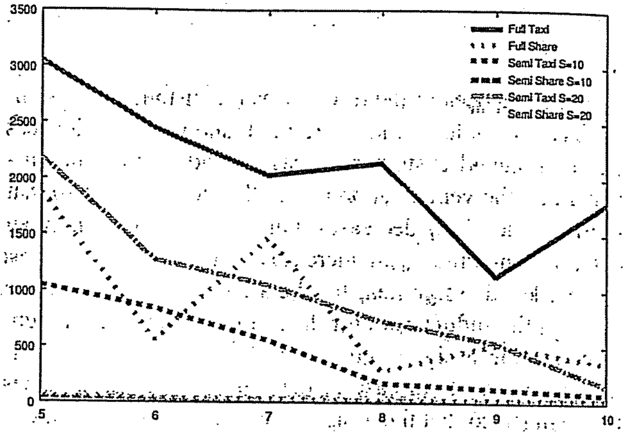


Figure 10. Average of waiting time

Figure 12 and Figure 13 show the average of walking distance and the average of traveling time (i.e., sum of waiting, riding and walking time) for customers. From these results, the full-demand can decrease walking distance, but cause the increase of traveling time. On the other hand, the semi-demand can decrease traveling time, but cause the increase of walking distance.

6. Conclusion

In this paper, we focused on the dynamic transport service which is called demand-bus system. The most important feature of this service is on-demand processing, but existing system is still on the developing stage. Therefore, our paper examined the realization of the dynamic demand-bus service in the large transport area for a large number of

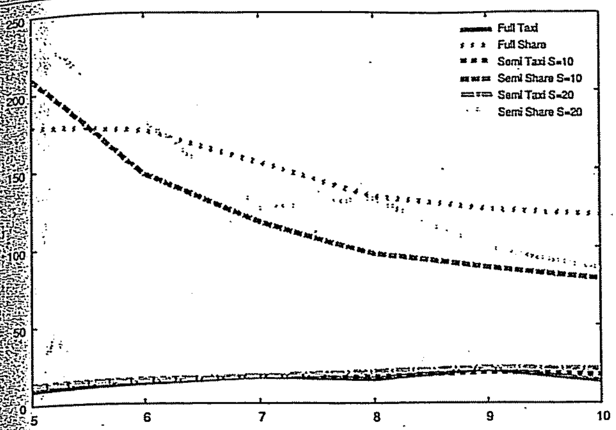


Figure 11. Average of riding time

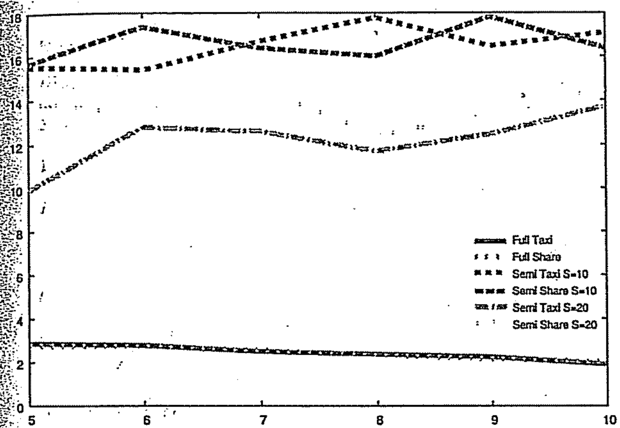


Figure 12. Average of walking distance

customers. We classified the demand-bus system into two types: semi-demand and full-demand, on the basis of bus station types. Moreover, we proposed methods to solve two real-time problems: assigning and scheduling. Finally, we reported our simulation results.

In the future work, we have to consider load-balancing among vehicles. Furthermore, we want to simulate our approach in our real life environment.

Acknowledgment

We would like to thank the 21st Century COE (Center of Excellence) Program for 2002, a project titled Intelligent Media (Speech and Images) Integration for Social Information Infrastructure, proposed by Nagoya University. And, we acknowledge to Prof. Naohiro Ishii of Aichi Institute of Technology for perspective suggestion.

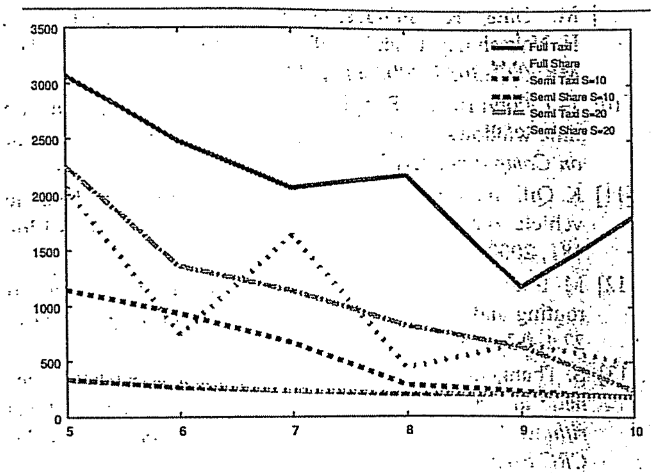


Figure 13. Average of traveling time

References

[1] M. Desrochers, J. Lenstra, M. Savelsbergh, and F.Soumis. Vehicle routing with time windows: Optimizatin and approximation. *Vehicle Routing: Methods and Studies*, pages 65–84, 1988.

[2] M. Gendreau and J. Potvin. Dynamic vehicle routing and dispatching. *Fleet Management and Logistics*, pages 115–126, 1998.

[3] T. Ibaraki, M. Kubo, T. Masuda, T. Uno, and M. Yagiura. Effective local search algorithms for the vehicle routing problem with general time window constraints. In *Proc. of MIC'2001*, pages 293–297, 2001.

[4] S. J. Louis, X. Yin, and Z. Y. Yuan. Multiple vehicle routing with time windows using genetic algorithms. In P. J. Angeline, Z. Michalewicz, M. Schoenauer, X. Yao, and A. Zalzala, editors, *Proceedings of the Congress on Evolutionary Computation*, volume 3, pages 1804–1808, Mayflower Hotel, Washington D.C., USA, 6-9 1999. IEEE Press.

[5] N. Mukai, J. Feng, and T. Watanabe. Distributed bus system using two spatial indexing structures for dynamic vehicle routing problem. In *Proc. of 11th World Congress on ITS*, 2004.

[6] N. Mukai, J. Feng, and T. Watanabe. Heuristic approach based on lambda-interchange for vrtpr-tree on specific vehicle routing problem with time windows. In *Proc. of 17th International Conference on Industrial & Engineering Applications of Artificial Intelligence & Expert Systems*, pages 229–238, 2004.

[7] N. Mukai, J. Feng, and T. Watanabe. Indexing approach for deliverly demands with time constraints. In *Proc. of 8th Pacific Rim International Conference on Artificial Intelligence*, pages 95–103, 2004.

[8] H. Nakashima, K. Kurumatani, and H. Itoh. Supporting a society with ubiquitous computing. *IPSJ-MGN450904t*, 45(9), 2004.

[9] M. Ohta, K. Shinoda, I. Noda, K. Kurumatani, and H. Nakashima. Usability of demand-bus in town area. *IPSJ-MBL02023033*, 2002(115), 2002.

[10] J.-Y. Potvin and S. Bengio. The vehicle routing problem with time windows — part II: Genetic search. *INFORMS Journal on Computing*, 8:165–172, 1996.

[11] K. Qili and K. Orig. A reactive method for real time dynamic vehicle routing problem. In *Proc. of ICTAI 2000*, pages 176–181, 2000.

[12] M. Solomon and J. Desrosiers. Time window constrained routing and scheduling problems. *Transportations Science*, 22:1–13, 1988.

[13] S. Thangiah. Vehicle routing with time windows using genetic algorithms. *Application Handbook of Genetic Algorithms: New Frontiers, Volume II. Lance Chambers (Ed.), CRC Press*, pages 253–277, 1995.