Comparison between Centralized Global Optimization and Distributed Local Optimization for Traffic Jam Avoidance

Ken Ohara
Graduate School of Engineering
Osaka Prefecture University
Sakai, Osaka 599-8531, Japan
+81-72-254-9351
ohara@ci.cs.osakafu-u.ac.jp

Yusuke Nojima
Graduate School of Engineering
Osaka Prefecture University
Sakai, Osaka 599-8531, Japan
+81-72-254-9351
nojima@cs.osakafu-u.ac.jp

Hisao Ishibuchi
Graduate School of Engineering
Osaka Prefecture University
Sakai, Osaka 599-8531, Japan
+81-72-254-9350
hisaoi@cs.osakafu-u.ac.jp

ABSTRACT

We consider a traffic flow model where the information about the actual travel time for each alternative route is not available when each driver performs route selection. For such a traffic flow model, we examine two routing methods to minimize the average travel time over all vehicles running in the model. One method tries to minimize the average travel time globally. It is assumed in this method that a central manager determines the routes of all vehicles. Since the number of combinations of vehicles' routes exponentially increases as the number of vehicles increases, we need an efficient combinatorial optimization technique. In this paper, we employ a genetic algorithm to search for a near-optimal route combination for all vehicles. In the other method, each driver tries to minimize his/her own travel time locally with no central manager. It is assumed in this method that each driver selects the route with the shortest estimated travel time among alternative routes. Each driver uses a neural network for the travel time estimation. Through computational experiments, we clearly demonstrate the characteristic features of each method.

Categories and Subject Descriptors

I.2.1 [**Artificial Intelligence**]: Applications and Expert Systems – *Games.*

General Terms

Experimentation.

Keywords

Traffic congestion, traffic flow model, route selection, genetic algorithms, neural networks.

1. INTRODUCTION

Traffic congestion exacts a terrible social and economic toll on society. It occurs only when a demand for a roadway is greater than its capacity. There are two common approaches to the alleviation of traffic congestion [1]. One is to expand roadway's capacity, and the other is to adjust demands for roadways. The latter approach includes methods for reducing total traffic demand for roadways (e.g., a driver changes the time zone when (s)he uses a car and chooses the alternative transportation) and methods for distributing traffic properly. Our study is concerned with proper traffic distribution. Traffic assignment to distribute traffic properly includes many mathematical assignment techniques such as equilibrium assignment [2]. In these mathematical techniques, it is usually assumed that the information about the actual travel time for each alternative route is available when each driver performs route selection [3]. Such information, however, is not usually available in the real-world road traffic system because the road environment around drivers constantly changes. Figure 1 shows an example of this situation. Suppose that each vehicle runs from the start point S to the goal point G. Each driver has to select one route from the three alternative routes A, B and C. As shown in Fig. 1, traffic congestion exists on the route B. The drivers who get the information about travel time for each alternative route probably select the route A or C to avoid the traffic congestion on the route B. When many drivers use the same traffic information, the traffic congestion on the route B will be cleared up soon. In this case, such route selection by many drivers may cause new traffic congestion on the route A and/or C. Based on these discussions, we assume in this paper that the information about the actual travel time for each alternative route can not be given to a driver because the actual travel time depends on other drivers' route selection.

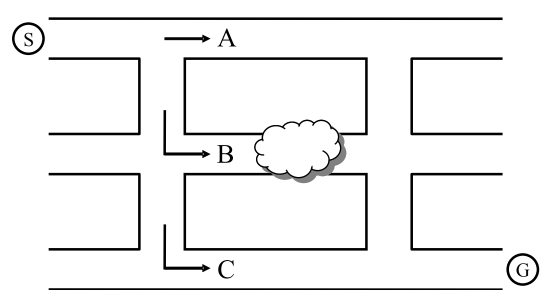

Figure 1. Example of a traffic flow model. Traffic congestion exists on the route B.

In this paper, first we develop a traffic flow model which has similar characteristics to real-world road traffic. In this model, the information about the actual travel time for each alternative route is not available when each driver performs route selection. Next we explain two routing methods to minimize the average travel time over all vehicles running in our traffic flow model. One method tries to minimize the average travel time globally. It is assumed that a central manager determines the routes of all vehicles. Since the number of possible combinations of vehicles' routes exponentially increases as the number of vehicles increases, we need an efficient combinatorial optimization technique. We employ a genetic algorithm [4] to search for a near-optimal route combination for all vehicles. In the other method, each driver tries to minimize his/her own travel time locally with no central manager. It is assumed that each driver selects the route with the shortest estimated travel time among alternative routes. Each driver uses a neural network for the travel time estimation.

This paper is organized as follows. First we explain our traffic flow model in Section 2. Next we explain the two routing methods (i.e., centralized global optimization and distributed local optimization) to minimize the average travel time in Section 3. Then the two routing methods are compared with each other through computational experiments on our traffic flow model in Section 4. Experimental results show the characteristic features of each method. Finally Section 5 concludes this paper.

2. TRAFFIC FLOW MODEL

In this section, we explain our traffic flow model. This model will be employed in Section 4 to compare the two routing methods through computational experiments.

In general, traffic flow models can be divided into macroscopic and microscopic models. In macroscopic models, traffic flow is treated as a phenomenon based on fluid dynamics [5]. On the other hand, traffic flow is treated as the interaction between each vehicle in microscopic models. Yikai et al. [6], [7] proposed a traffic flow model based on fuzzy estimation of each vehicle' behavior. Tamaki et al. [8], [9] proposed a traffic flow model using cellular automata where the stochastic velocity model [10] was utilized. These studies are examples of microscopic models.

In this paper, we develop a traffic flow model using cellular automata as in [6]-[15]. A cellular automaton is a discrete model which has been studied in computability theory, mathematics, and theoretical biology. It is based on a regular grid of cells, each of which assumes one of a finite number of states. Time is also discrete. The state of a cell at time $t+1$ is a function of the states of a finite number of neighboring cells at time t.

2.1 Global Transition Rules

We explain global transition rules used in our traffic flow model. Figure 2 shows the route map of our model. The simulation area is divided into squared cells. Roads (i.e., alternative routes) are represented by white cells in Fig. 2. All vehicles travel from the start point S to the goal point G.

Each driver performs route selection at the point P_1, so that every vehicle runs on either the route L_1 or L_2 to the goal point G. We assume that the route L_1 is the main route to goal point G. There is a traffic signal on the route L_1. We also assume that the route L_2 is a detour for the route L_1 to the goal point G. The two routes merge at the point P_2. We assume that the vehicle traveling on the route L_1 has the right-of-way because the route L_1 is the main route. The traffic signal has only two lights: red (i.e., stop) and green (i.e., go). The two lights alternate. The duration times of the red and green lights are four and two time steps, respectively.

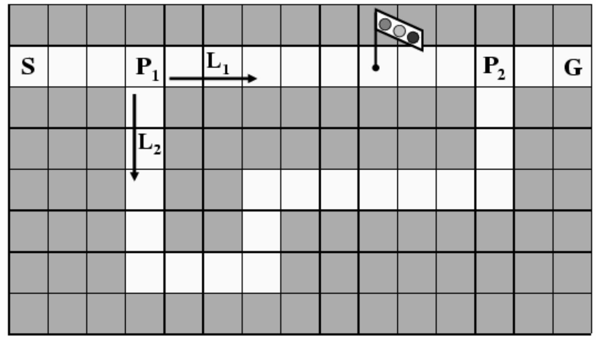

Figure 2. Route map of the traffic flow model.

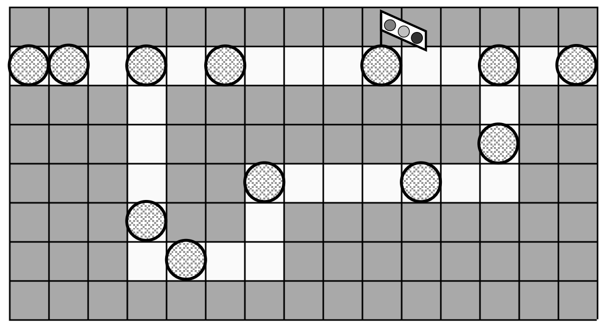

Figure 3. Example of a smooth traffic flow.

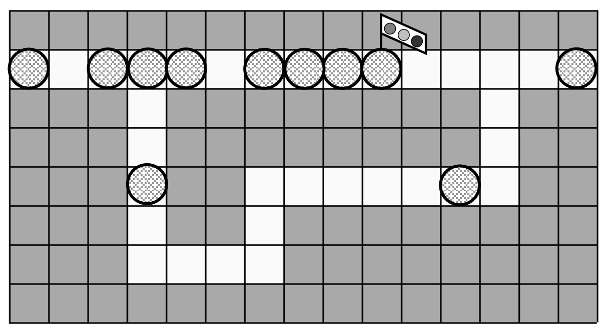

Figure 4. Traffic congestion on the route L_1.

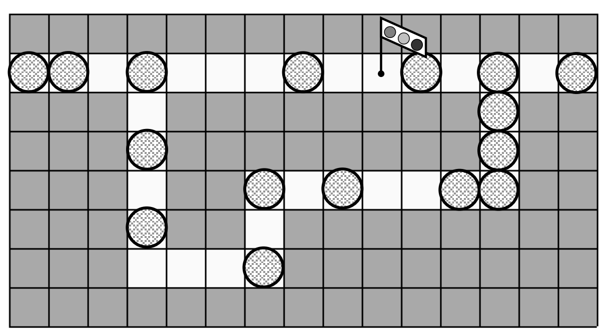

Figure 5. Traffic congestion on the route L_2.

Figures 3-5 show three typical situations in our traffic flow model. We can see a smooth traffic flow in Fig. 3 which will be realized by the centralized global optimization method in Section 4. On the other hand, there is traffic congestion in Fig. 4 and Fig. 5. This is because almost all vehicles select the route L_1 in Fig. 4 and the route L_2 in Fig. 5. In these figures, we need an efficient route selection method to realize a smooth traffic flow such as Fig. 3.

In our traffic flow model, the information about the actual travel time for each route is not available when each driver performs route selection at the point P_1. This is because the interaction at the merging point P_2 depends on other vehicles behinds the driver. There exist many similar situations in the real-world traffic.

2.2 Local Transition Rules

In this subsection, we explain local transition rules in our traffic flow model. Our model is a deterministic one, which follows the Wolfram's rule 184 (CA-184) [13], [14] except for the traffic signal and the merging point P_2. The state of each cell is empty or occupied by a vehicle. The positions of all vehicles running in the model are updated synchronously. At every state transition time, each vehicle stays at the current cell or jump to its destination cell. The local transition rule is simply stated as "a vehicle moves only when its destination cell is empty." This means that the drivers are short-sighted. That is, they do not know whether the vehicle in front can move or is also stuck by another car. Therefore, the state of each cell s_i is entirely determined by the occupancy of the cell itself and its two nearest neighbors s_{i-1} and s_{i+1} along the route. Figure 6 summarizes the local transition rule where all the eight possible configurations $(s_{i-1}\, s_i\, s_{i+1})_t \longrightarrow (s_i)_{t+1}$ are given. Empty and occupied cells are shown by white and black squares, respectively. In Fig. 6, the state $(s_i)_{t+1}$ of the center cell at the next time step $t+1$ is specified based on the states s_{i-1}, s_i and s_{i+1} at the current time step t.

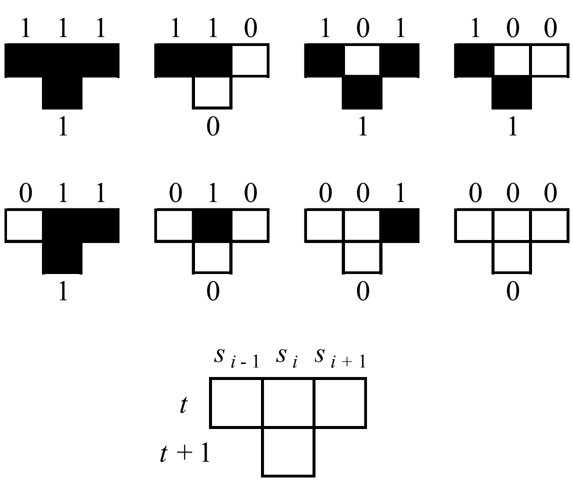

Figure 6. Illustration of local transition rules for the state s_i of the *i*-th cell under the motion rule of CA-184.

3. ROUTING METHODS

In this section, we explain two routing methods (i.e., centralized global optimization and distributed local optimization) to minimize the average travel time over all vehicles running in our traffic flow model. In the centralized global optimization method, the average travel time over all vehicles is minimized by a central manager who can determine the routes of all vehicles. On the other hand, each driver tries to individually minimize his/her own travel time based on the estimation of the travel time for each route in the distributed local optimization method.

3.1 Centralized Global Optimization

In this method, the routes of all vehicles are determined by a central manager who can observe and control all vehicles. It is, however, very difficult to find the optimal route selection for all vehicles for large-scale problems. This is because the number of possible combinations of vehicles' routes exponentially increases as the number of vehicles increases. Thus we need an efficient combinatorial optimization technique. We use a genetic algorithm to search for a near-optimal route combination.

When we apply a genetic algorithm to an optimization problem, we have to represent each solution by a string. In the case of route selection, the combination of routes for N vehicles is coded as a string of length N as

$$r = r_1\, r_2 \cdots r_i \cdots r_N\,, \quad i = 1, 2, \ldots, N\,, \tag{1}$$

where r_i denotes the route selected by the i-th vehicle, that is, $r_i = 0$ means that the i-th vehicle selects the route L_1 and $r_i = 1$ means that the vehicle selects the route L_2. The fitness value $f(r)$ of the string r is the average travel time over the N vehicles.

The outline of our genetic algorithm is written as follows:

Step 1: Randomly generate M binary strings of length N to construct an initial population.

Step 2: Execute a single computer simulation using each binary string for our traffic flow model with the N vehicles to calculate the average travel time. The fitness value of each string is the calculated average travel time.

Step 3: Generate the next population by iterating the following genetic operations M times.

1. A pair of parent strings are selected from the current population using the binary tournament selection scheme.
2. A new string is generated from the selected pair of parent strings by crossover and mutation.

Step 4: Update the current population using its single elite sting and the newly generated population.

Step 5: If a prespecified stopping condition is not satisfied, return to Step 2. Otherwise, the algorithm terminates.

In this paper, we use the one-point crossover and the bit-flip mutation in Step 3 and use the elitist strategy in Step 4.

3.2 Distributed Local Optimization

In this method, each driver selects the route with the shortest estimated travel time between the two alternative routes. We develop route selection agents [16], [17] using three-layer feed-forward neural networks to estimate the travel time for each route from the available road information.

The available road information for the travel time estimation is as follows:

1. The traffic signal (green: 1, red: 0),
2. The degree of traffic congestion of each route.

We define the traffic congestion degree f_i of the route L_i as

$$f_i = \sum_{j=1}^{m} c_j \cdot x_{ij}\ , \quad i = 1,\ 2\ , \tag{2}$$

$$c_j = \left(\frac{1}{2}\right)^{j-1}, \quad j = 1,\ 2,\ \ldots, m\ , \tag{3}$$

$$x_{ij} = 0 \text{ or } 1\ , \tag{4}$$

where m is the number of the cells taken into account ($m = 5$ in this paper), c_j is the weighting factor for the j-th cell from the point P_1, and x_{ij} is the state of the j-th cell on the route L_i from the point P_1 (i.e., empty: 0, occupied: 1). Figure 7 shows an example. Whereas there exist two vehicles on each route, their congestion degrees are different for the vehicle A (i.e., $f_1 = 1.2500, f_2 = 0.5625$). This is because the vehicles on the route L_1 are nearer to the point P_1 than those on the route L_2.

Each agent chooses a route in the following manner:

Step 1: Obtain the available road information.
Step 2: Estimate the travel time for each route using the neural network.
Step 3: Select the route with the shortest estimated travel time.

All agents use the common (i.e., shared) neural network. The input vector to the neural network is a three-dimensional vector of the following form: (the route number, the traffic signal, the traffic congestion degree). The output is the estimated travel time for the selected route. The back-propagation algorithm [18] is used to train the neural network.

First the neural network is randomly initialized. Next each agent chooses one route using the neural network. In the learning phase, each agent selects the route L_1 or L_2 according to the following probabilities:

$$q_i = 1 - \frac{d_i}{\sum_{j=1}^{2} d_j}\ , \quad i = 1,\ 2\ , \tag{5}$$

where d_i is the estimated travel time for the route L_i ($i = 1$,2), which is calculated as the output from the neural network.

When a vehicle arrives at the goal point G, a single input-output pair for the learning of the neural network is obtained. The input vector in the generated input-output pair consists of the selected route and the used information at the point P_1 when the vehicle chose that route. The corresponding target output is the actual travel time of the vehicle. Whenever the input-output pair is obtained, the learning of the neural network is performed to minimize the following cost function:

$$E_p = \frac{\left(t_p - o_p\right)^2}{2}\ , \tag{6}$$

where p is an index of the input-output pair, t_p is the target output (i.e., the actual travel time), and o_p is the output from the neural network (i.e., the estimated travel time).

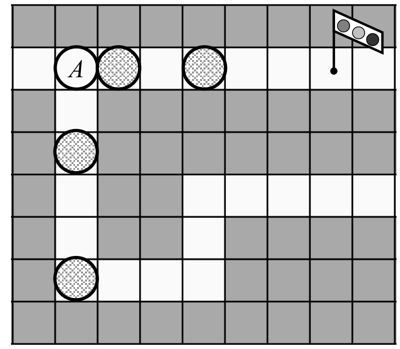

Figure 7. Example of a traffic flow.

4. COMPUTATIONAL EXPERIMENTS
In this section, we compare the two methods with each other to demonstrate the characteristic features of each method.

4.1 Learning for Estimating Travel Time
We performed the learning of the neural network using the following parameter values:

Number of hidden unit: 20,
Learning rate: 0.8,
Momentum term constant: 0.6,
Total number of input-output pairs: 100000.

In Fig. 8, we show how the squared error in (6) was decreased by the learning of the neural network. From Fig. 8, we can see that the error decreased to almost zero by the learning of the neural network. Thus we can say that the route selection agent can estimate the travel time for each route properly.

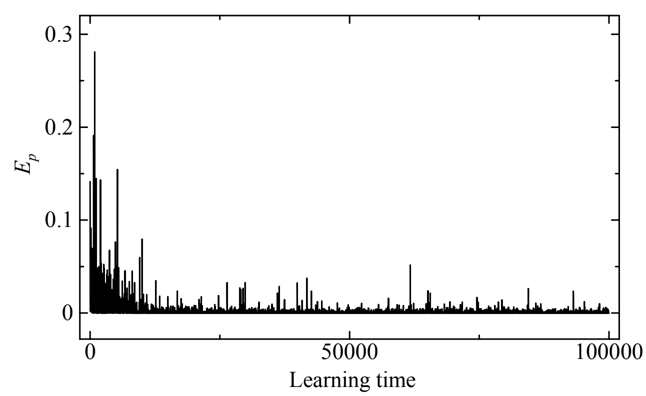

Figure 8. Squared error in the estimation of the travel time.

4.2 Comparison between Two Methods
We compared the two methods with each other (i.e., centralized global optimization and distributed local optimization). For comparison, we also examined a random selection method and a simple heuristic method. In the random method, each driver selects a route randomly. We calculated the average travel time from 50 independent runs of our computer simulation. On the

other hand, the simple heuristic method always chooses the route with the less congestion degree. The average travel time can be calculated in a deterministic manner.

We used the following parameter values in the centralized global optimization method follow:

Population size: 50,
Tournament size: 2,
Crossover probability: 1.0,
Mutation probability: $1/N$ (N: Number of vehicles),
Stopping condition: 1000 generations.

The average travel time was calculated from ten independent runs except for the case with 10000 vehicles. In this case, the average travel time was calculated from five independent runs.

4.2.1 Case 1: Number of vehicles is small

Here we report experimental results with 20 vehicles. Since the number of vehicle is very small, we can examine all combinations of vehicles' routes (i.e., 2^{20} combinations). This means that we can obtain the optimal value of the average travel time over all vehicles by centralized global optimization. The optimal value is obtained as 18.55. We show the average travel time by each method in Table 1. When we used a genetic algorithm for centralized global optimization, the same average travel time was obtained as 18.55. On the other hand, the average travel by the distributed local optimization method is large than its optimal value in Table 1 whereas it outperformed random selection and simple heuristic. From these results, we can see that the local optimization by each vehicle to minimize its own travel time does not lead to the globally optimal value of the average travel time over all vehicles by the central manager.

Table 1. Average travel time obtained by each method in the case where the number of vehicles is 20.

Route selection method	Average travel time
Global optimization by GA	18.55
Local optimization by NN	21.95
Random selection	23.24
Simple heuristic	22.30

4.2.2 Case 2: Number of vehicles is large

In the same manner as the previous subsection, we perform computer simulations using genetic algorithm-based global optimization, neural network-based local optimization, random selection and simple heuristic. We examined various specification of the number of vehicles: 20, 50, 100, 200, 500, 1000, 2000, 5000, and 10000. Experimental results are summarized in Fig. 9. On the other hand, Figure 10 shows the decrease in the average travel time during the evolution by the genetic algorithm for each specification. The average travel time in Fig. 10 was calculated using the elite string at each generation. From Fig. 9 and Fig. 10, we can see that the performance of the genetic algorithm-based global optimization method was degraded by increasing the number of vehicles. When the number of vehicle is 10000, it is inferior to the simple heuristic method in Fig. 9. This is because the centralized global optimization becomes very difficult due to the exponential increase in the search space (i.e., 2^{10000} combinations).

On the other hand, the performance of the distributed local optimization method is much less sensitive to the increase in the number of vehicles. This is because the optimization is performed by each vehicle. The best results were obtained by the distributed local optimization method when the number of vehicles was 5000 and 10000 in Fig. 9.

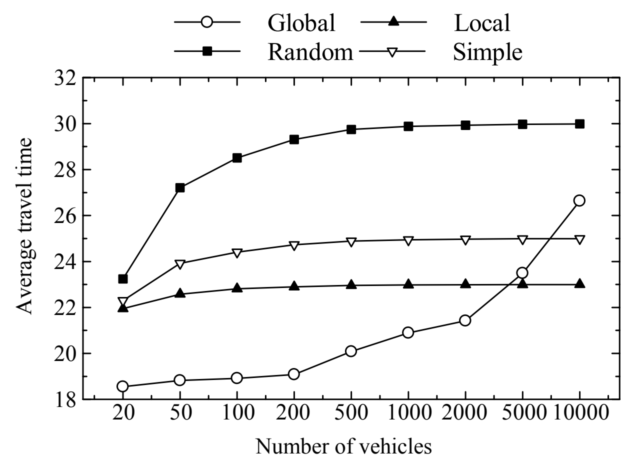

Figure 9. Average travel times obtained by each method.

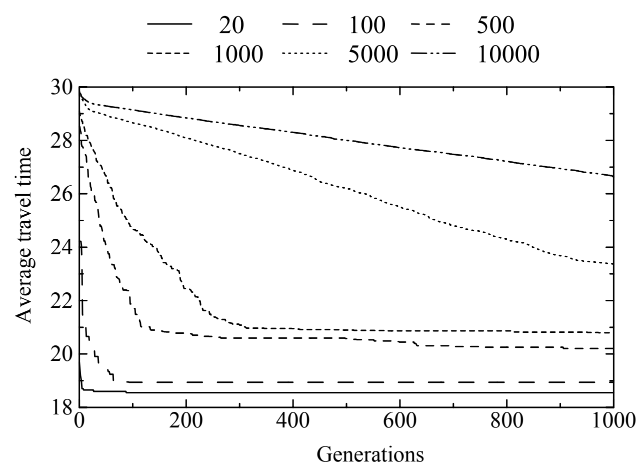

Figure 10. Average travel time of the elite string of the global optimization.

5. CONCLUSIONS

In this paper, we examined the performances of the two routing methods (i.e., centralized global optimization and distributed local optimization) for minimizing the average travel time over all vehicles using a simple traffic flow model with two alternative routes. Through computational experiments, we compared these two methods with each other. When the number of vehicles was very small (i.e., 20 vehicles: a micro-scale traffic flow model), we obtained the same optimal value of the average travel times by the genetic algorithm-based global optimization method as the enumeration of all possible combinations of vehicles' routes. The performance of the genetic algorithm-based global optimization method, however, was significantly degraded by the increase in the number of vehicles. When the number of vehicle was large (i.e., 10000 vehicles: a macro-scale traffic flow model), the genetic algorithm-based global optimization method was inferior to a simple heuristic method. In contrast, the neural network-based local optimization method was less sensitive to the increase in the number of vehicles. When the number of vehicles was 5000 and 10000, the best results were obtained by the local optimization method.

Real-world road traffic can be thought as a system with a lot of uncertainty and unpredictability. Let us consider the case where a traffic accident happens. In this case, it is very difficult for the centralized global optimization method to quickly search for the optimal routes of all vehicles especially when the traffic accident influences the route selection of a large number of vehicles. Thus the use of only the centralized global optimization method can not always generate a smooth traffic flow. Due to the lack of the global optimization ability, the use of only the distributed local optimization method can not always generate a smooth traffic flow, either.

In our traffic flow model, a relatively good result can be obtained when all vehicles choose the route L_2 independent of the number of vehicles. In this case, the average travel time is 21.00, which is better than all the four methods in Fig. 9 when the number of vehicles is more than 2000. This extreme traffic flow, however, is not likely to happen in real-world situations. This is because the cooperative behavior of all vehicles is required to realize this extreme traffic flow with the relatively good average travel time. When almost all vehicles select the route L_2, the other vehicles can benefit from choosing the route L_1. In this sense, our traffic flow model can be viewed as a kind of iterated dilemma game (e.g., see [19] for the iterated prisoner's dilemma game).

One future research direction is to simplify our traffic flow model as an iterated dilemma game. Another future research direction is to make our traffic flow model more realistic (e.g., by increasing the number of routes, traffic lanes and traffic signals, and employing a stochastic velocity model). We are also planning to implement a hybrid approach which is a combination of centralized global optimization and distributed local optimization.

6. ACKNOWLEDGMENTS

This work was partially supported by the SCAT (Support Center for Advanced Telecommunications Technology Research) Foundation and the Grant-in-Aid for Exploratory Research: KAKENHI (18650055).

7. REFERENCES

[1] Road Bureau, Ministry of Land, Infrastructure and Transport, http://www.mlit.go.jp/road/road_e/contents.html.

[2] Akiyama, T., and Nomura, T. The Proposal of Fuzzy Traffic Assignment Models, *Journal of the Eastern Asia Society for Transportation Studies*, Vol.3, No.6, (1999) 263-277.

[3] Wardrop J. G. Some theoretical aspects of road traffic research, *Proc. Inst. Civ. Eng. Part 2*, (1952) 325-378.

[4] Goldberg D. E. Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley, (1989).

[5] Lighthill, M. J., and Whitham, G.. B., On kinematic waves. I. Flood movement in long rivers, *Proc. Royal. Soc*., London, 229 (Ser. A), (1955) 281-316.

[6] Yikai, Y., Honda N., and Itakura N. Fuzzy Driving Model for Road Traffic Simulator. *Journal of Japan Society for Fuzzy Theory and Systems*, (2000) 425-435.

[7] Yikai, K., Honada, N., and Satoh, A. A Fuzzy Processor Consisting of Memory and Controlling. *LSI Fifth IFSA World Congress*, (1993) 789-792.

[8] Tamaki T., Yasue S., and Kita E. City Traffic Simulation Using Cellular Automata with Stochastic Velocity Model. *The 2004 International Conference on Parallel and Distributed Processing Techniques and Applications - PDPTA2004*, (2004) 331-337.

[9] Tamaki T., Yasue S. and Kita E. Traffic Flow Simulation Using Cellular Automata, *Proceedings of Civi and Structural Engineering Computing*, Netherland, (2003).

[10] Nagel K., and Schreckenberg, M. Cellular automaton model for freeway traffic, *Journal of Physics 1 france*, Vol.2, (1992) 2221-2229.

[11] Maerivoet S. and De Moor B., Cellular Automata Models of Road Traffic, *in Physics Reports*, vol. 419, nr. 1, (November 2005) 1-64.

[12] Larrag M.E., Del Rio J.A., and. Alvarez-lcaza L. Cellular automata for one-lane traffic flow modeling, *Transportation Research*, Part C 13, (2005) 63–74.

[13] Wolfram, S. Theory and Application of Cellular Automata. World Scientific, (1986).

[14] Yukawa, S., Kikuchi, M. and Tadaki, S. Dynamical phase transition in one-dimensional traffic flow model with blockage. *J. Phys. Soc. Jpn 63, 10*, (1994) 3609-3618.

[15] Schreckenberg, M., Schadschneider, K., Nagel, A., and Ito, N. Discrete stochastic models for traffic flow, *Phys. Rev. E51*, (1995) 2939-2949.

[16] Rogers, S., Fiechter, C., and Langley, P. An adaptive interactive agent for route advice. *in Proc. of the third annual conf. on Autonomous Agents*, ACM Press, (1999) 198-205.

[17] Rogers, S., Fiechter, C., and Langley, P. A route advice agent that models driver preferences, *in Proc. of the 1999 AAAI Symposium*, Stanford, CA, (1999) 106-113.

[18] D. E. Rumelhart, J. L. McClelland, and the PDP Research Group, *Parallel Distributed Processing*, Vol.1. MIT Press, Cambridge, MA, (1986).

[19] Axelrod, R. The evolution of strategies in the Iterated Prisoner's Dilemma, in L. Davis (ed.), *Genetic Algorithms and Simulated Annealing*, Morgan Kaufmann, Los Altos, (1987) 32-41.