Efficient Routing Strategy in Traffic Flow

Takayuki Kimura†, Tohru Ikeguchi‡, and C.K. Tse†

†Department of Electronic and Information Engineering, The Hong Kong Polytechnic University
Hung Hom, Kowloon, Hong Kong
‡Graduate School of Science and Engineering, Saitama University
255 Shimo-Ohkubo, Saitama, 338-8570, Japan
Email: kimura@eie.polyu.edu.hk

Abstract—To alleviate congestion in traffic networks is one of the most important problems to establish effective movements of vehicles. In this paper, we propose a new routing strategy using a memory effect. From the view points in the field of the complex networks, this paper deals with efficient control of flow on the complex networks. Conventional routing strategies use probabilistic rules to route the vehicles on the traffic networks. On the other hand, our proposed method uses a deterministic rule, or the memory effect. The memory effect means how many vehicles are moved through the nodes in the past history. In addition, another reason why we use the memory effect is that every node does not need to communicate the adjacent nodes to route the vehicles to their destinations because the memory effect is the information owned by each node. From the results of the numerical simulations, by using the memory effect, our proposed method reduces the traffic congestion effectively compared with the conventional routing strategies. Then, we confirmed that our proposed strategy effectively routes the vehicles to their destinations.

1. Introduction

An important problem to establish effective flows on networks is how to alleviate congestion. Recently, more and more researchers have begun to develop models for analyzing dynamic flows on complex networks to resolve this problem. For example, Ohira et al. have investigated the optimal network structure for packet flow[1], and proposed routing strategy with a stochastic rule which has tolerance for the computer networks with congestion. Arenas et al.[2] analytically showed a transition point for hierarchical branching networks. Several structures of the computer networks with congestion have been tested in [3]. Park et al.[4] proposed a cascade failure strategy for complex networks, and they found a congestion phase-transition phenomenon by a key parameter characterizing the node capacity. Kimura et al. proposed a routing strategy with chaotic neurodynamics[5, 6] to alleviate packet congestion in computer networks, and the proposed routing strategy shows high performance for several topologies of the complex networks[5, 6]. The routing strategy with chaotic neurodynamics[5, 6] is improved by adding the information of the adjacent routers[7], and the improved chaotic routing strategy shows higher performance than the conventional routing strategies[7].

A good routing strategy for traffic networks is to be moved vehicles to their destinations as quickly as possible by avoiding traffic congestion. The Dijkstra algorithm[8] is one of the basic routing strategies. If the number of vehicles in the traffic network is very small, this routing strategy works well. However, if the number of the vehicles increases, the Dijkstra algorithm exhibits poor performance because it uses only the information of the shortest paths; the node through which many shortest paths pass are easily congested. Thus, it is a very important problem to propose a sophisticated routing strategy for avoiding the traffic congestion. Recently, Wu et al. proposed dynamic traffic model to route the vehicles in the traffic networks[9]. They applied a stochastic rule to determine the next node of the vehicles, and found a phase transition from free flow to congestion for the traffic networks[9].

In this paper, as one of the new routing strategies for the traffic networks, we proposed a routing strategy with a memory effect. From results of computational simulations, we confirmed that proposed routing strategy shows higher performance than the conventional routing strategies for different underlying network topology.

2. Traffic network model

We used weighted and undirected graphs $G = (V, E)$ to construct traffic network models[9] where V is the set of nodes, and E is the set of links. Each node represents an intersection in the traffic network, and each link represents a road connection between the intersections. $N = |V|$ signifies the number of nodes in the network. In this traffic model, the ith node ($i = 1, \dots, N$) has two characteristics described as follows:

1. Largest stored capacity H_i. The largest stored capacity corresponds to the maximum number of vehicles stored in the ith node. The largest stored capacity H_i is defined by $H_i = \rho \times k_i$, where ρ is a control parameter and k_i is a degree of the ith node.

2. Processing capacity C_i. Processing capacity corresponds to the the number of vehicles moving to the

next nodes from the ith node at each iteration. The processing capacity C_i is defined by $C_i = \lambda \times H_i$, where λ is a control parameter.

In this paper, we set $\rho = 2.0$ and $\lambda = 0.2$. In this traffic network, when a vehicle is generated at a node, it is moved from a node to another through the links. The vehicle is stored at the tail of the stored capacity of the moved node. All the vehicles are moved according to First-In-First-Out basis. In addition, if the stored capacity of the moved node is full of the vehicles, the vehicle is not moved to the node, then, the vehicle will wait for the next opportunity.

3. A routing strategy with a memory effect

To realize a proposed vehicle routing strategy using a memory effect, first, we introduced a neural network in the same way used in [5, 6, 10]. In our proposed routing strategy, each node has its own neural network which is fully connected by neurons which correspond to the adjacent nodes. A connection between the ith node and its jth adjacent node in the traffic network is expressed by the ijth neuron in the neural network.

In the proposed routing strategy, each neuron has two kinds of effect: a memory effect and a distance effect. The memory effect is defined as follows:

$$\zeta_{ij}(t+1) \quad = \quad \alpha \sum_{\gamma=0}^{t} k_r^{\gamma} x_{ij}(t-\gamma), \tag{1}$$

where α is a scaling parameter of the memory effect; k_r is a decay parameter; $x_{ij}(t)$ is an output of the ijth neuron at the tth time and will be defined by Eq.(3).

Using the memory effect, if a neuron which has just fired hardly fires for a while. Namely, each node can memorize a past routing history using the memory effect, then, an adjacent node to which many vehicles have been moved is not selected as a moved node for a while.

The distance effect is defined as follows:

$$\xi_{ij}(t+1) \quad = \quad \beta\left(\frac{d_{ij} + d_{jg(v_i(t))}}{\sum_{i}^{N_i}(d_{ij} + d_{jg(v_i(t))})}\right), \tag{2}$$

where β is a controlling parameter of the distance effect; N_i is the number of adjacent nodes of the ith node; $v_i(t)$ is a moving vehicle from the ith node at the tth time; $g(v_i(t))$ is a destination of $v_i(t)$; d_{ij} is the shortest distance between the ith node and the jth adjacent node; $d_{jg(v_i(t))}$ is the shortest distance between the jth adjacent node and $g(v_i(t))$.

The output of the ijth neuron is defined as follows:

$$x_{ij}(t) = \begin{cases} 1 & \left(\begin{array}{l}\text{if the } j\text{th adjacent node is de-}\\ \text{termined as the moved node}\\ \text{from the } i\text{th node}\end{array}\right), \\ 0 & \text{(otherwise).} \end{cases} \tag{3}$$

If the sum of Eqs.(1) and (2) of the ijth neuron takes the largest value among all the neurons in the neural network, the ijth neuron fires, that is, a vehicle at the ith node is moved to its jth adjacent node. Then, the output of the ijth neuron, $x_{ij}(t)$, is updated by Eq.(3).

4. Computer simulation

To measure the performance of the proposed routing strategy, we compared the proposed routing model with two conventional routing strategies. The first one is the Dijkstra algorithm. Vehicles in the Dijkstra algorithm moved to their destinations only along the shortest paths. In this model, we used the hop information as the weights of the links, then, calculated the shortest paths between the nodes in the traffic networks.

The second one is the Dynamic Traffic Model (DTM) proposed in [9]. To find the optimal adjacent nodes for the vehicles, this model uses the following probability:

$$P_j = \frac{T_{ij}^{\phi}(1+k_j)}{\sum_{i=1}^{N_i} T_i^{\phi}(1+k_i)}, \tag{4}$$

where ϕ is a control parameter; k_i is the degree of the ith node. T_{ij} is the distance between the ith node and the jth node which is calculated by the Bureau of Public Roads formula[11]. The Bureau of Public Roads formula[11] is defined as follows:

$$T_{ij} = t_{0i}\left(1 + \delta\left(\frac{Q_i}{H_i}\right)^{\psi}\right), \tag{5}$$

where δ and ψ are control parameters; t_{0i} denotes a cost at zero flows; Q_i is the number of existing vehicles of the ith node. we set the parameters ϕ in Eq.(4) to 0.1, δ and ψ in Eq.(5) to 0.15 and 4 in the same way used in [9].

In these simulations, we used two different types of the proposed routing strategies. The first one is the memory routing strategy with the hop information (MH), and the second one is the one with betweeness information (MB). A different point between these proposed routing models is characteristic of the weights of the links. The first one (MH) used the number of hops between the nodes to calculate the shortest distance, and the second one (MB) used the betweeness to calculate the shortest distance in the traffic network.

We conducted computer simulations by the following procedures. First, we generated vehicles with which the origins and the destinations were randomly assigned. In addition, each node calculates the shortest path from the node to the other nodes. Namely, each node always has a static routing table which contains an information list of the shortest distances between any two nodes. A selection of the adjacent node and movement of the vehicle were simultaneously conducted at every node. We conducted 30 simulations and averaged the results.

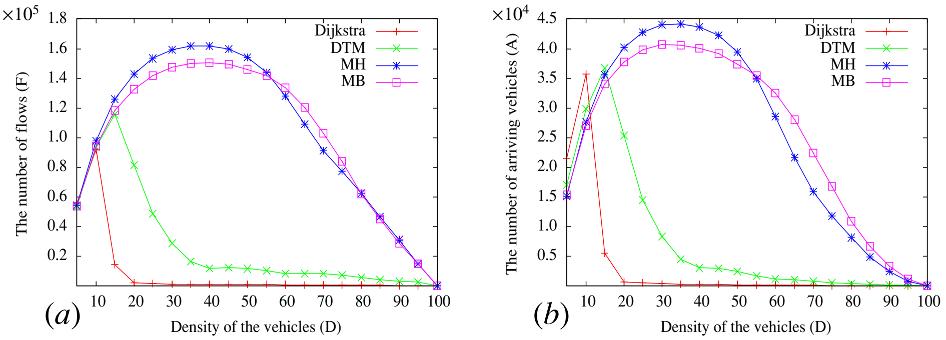

Figure 1: Relationship between the density of the vehicles (D) and (a) the number of flows (F), and (b) the number of vehicles arriving at their destinations (A) by the Dijkstra algorithm (Dijkstra), the Dynamic Traffic Model (DTM)[9], the memory routing strategy with the hop information (MH), and the memory routing strategy with the betweeness information (MB) for the scale-free networks.

We repeated the node selection and vehicle movement for 1,000 iterations. We fixed the total number of vehicles in the traffic network. Thus, when a vehicle arrived at its destination, we generated a new vehicle. Then, a origin and a destination of the new vehicle are randomly decided using uniformly distributed random numbers. We set the parameters in Eqs.(1)–(3) as follows: $\alpha = 0.5$, $k_r = 0.95$, and $\beta = 15.0$.

As topologies of the traffic networks, 100 nodes of the scale-free networks[12] and the Waxman's network[13] are used in these simulations. To evaluate the performance of the proposed strategies and the conventional routing strategies, we measured density of the vehicles, D, the number of flows, F, and the number of vehicles arriving at their destinations, A.

First, we evaluated four routing strategies, the Dijkstra algorithm (Dijkstra), Dynamic Traffic Flow (DTM)[9], the memory routing strategy with the hop information (MH), and the memory routing strategy with the betweeness information (MB) for the scale-free networks[12]. The scale-free networks are generated in the same way as Barabási and Albert[12]. This network is constructed by the following procedure: first, we made a complete graph which has four nodes. Then, we put a new node with three links at every time step. Next, we connected three links of the newly added node to the nodes already existing in the traffic network with the probability $\Pi(k_i) = \dfrac{k_i}{\sum_{j=1}^{N} k_j}$, where k_i is the degree of the ith node ($i = 1, \ldots, N$); N is the number of nodes at a current iteration.

Results for the scale-free networks are shown in Fig.1. In Fig.1(a), although the number of flows (F) in the Dijkstra algorithm (Dijkstra) and the Dynamic Traffic Model (DTM) rapidly decreases when the density of the vehicles (D) becomes larger, the memory routing strategy with hop information (MH) and the one with betweeness information (MB) keep high numbers of flows. In addition, In Fig.1(b), the number of arriving vehicles (A) in the proposed routing strategies (MH and MB) is larger than the Dijkstra and DTM.

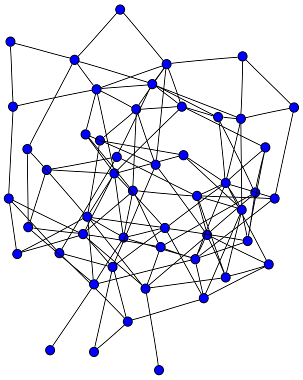

Figure 2: An example of the Waxman's network constructed by 50 nodes[13]. Parameters in Eq.(6) are set to $a_1 = 0.5$ and $a_2 = 0.15$, respectively.

Next, we evaluated four routing strategies for the Waxman's networks[13]. The Waxman's network is constructed by the following procedure: first, we put N nodes in a lattice pattern. Then, we connected the nodes by the following probability:

$$P_{ij} = a_1 \exp\left(\frac{-d_{ij}}{a_2 d_{max}}\right), \tag{6}$$

where P_{ij} is connection probability between the ith node and jth node; d_{ij} is the shortest distance between the ith node and jth node; d_{max} is the maximum distance in the network; a_1 and a_2 are control parameters which take between 0 and 1. In this network model, if a_1 becomes larger, there will be more links on the network. In addition, if a_2 becomes larger, there will be longer links on the traffic networks. An example network is shown in Fig.2. In this paper, we set the parameters $a_1 = 0.5$ and $a_2 = 0.5$.

Results for the Waxman's networks are shown in Fig.3. In Fig.3(a), as well as the results of the scale-free networks (Fig.1), the proposed routing strategies (MH and MB) keep larger number of flows (F) than the Dijkstra and DTM. Moreover, in Fig.3(b), the proposed routing strategies also

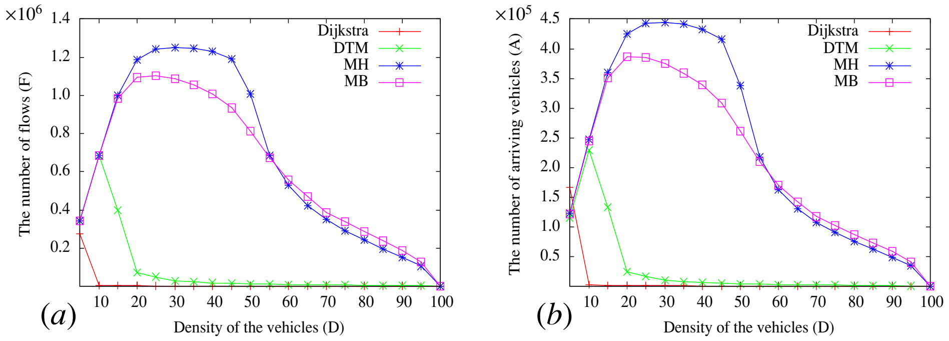

Figure 3: Relationship between the density of the vehicles (D) and (a) the number of flows (F), and (b) the number of vehicles arriving at their destinations (A) by the Dijkstra algorithm (Dijkstra), the Dynamic Traffic Model (DTM)[9], the memory routing strategy with the hop information (MH), and the memory routing strategy with the betweeness information (MB) for the Waxman's networks.

keep larger number of arriving vehicles (A) than the Dijkstra and DTM. In the case of the Waxman's networks, the performance of the memory routing strategy with the hop information (MH) is rather higher than the one with the betweeness information (MB) when the density of vehicles is between 20 and 50.

Vehicles in the Dijkstra algorithm and DTM cannot move to their destinations when the density of the vehicles increases because there is few flows in the traffic network. These results indicate that the traffic congestion easily occurs in the traffic network by using the Dijkstra algorithm and Dynamic Traffic model. Thus, the vehicles cannot arrive at their destinations because of the traffic congestion. On the other hand, vehicles in the proposed routing strategies can move to their destinations even if the density of the vehicles increases because the flows in the traffic network are kept higher. Further, from the results of Figs.1 and 3, even if the topology of the traffic network changes, the proposed routing strategy keeps high performance.

5. Conclusion

In this paper, as a new routing strategy in the traffic network, we proposed the memory routing strategy. Using memory effect, the proposed routing strategies show higher performance than the conventional routing strategies for the scale-free networks and the Waxman's networks. By using the memory effect, the performance of the proposed routing strategies becomes outstanding when we compare its performance with that of the decent-down hill routing strategy and the stochastic routing strategy. However, we do not clarify why the memory effect effectively works for alleviating the traffic congestion in this paper. Thus, in the future works, we consider analysis of the memory effect from the view point of the alleviation of the traffic congestion.

The research of T.I. is partially supported by Grant-in-Aids for Scientific Research (B) (No.20300085) from JSPS.

References

[1] T.Ohira and T.Sawatari, "Phase transition in computer network traffic model," *Physical Review E*, vol.58, pp.193–195, 1998.

[2] A.Arenas, A.Diaz-Guilera and R.Guimera, "Communication in Networks with Hierarchial Branching," *Physical Review Letters*, vol.86, pp.3196–3199, 2001.

[3] L.Zhao, Y.C.Lai, K.Park and N.Ye, "Onset of traffic congestion in complex networks," *Physical Review Letters*, vol.71, 026125-1–026125-8, 2005.

[4] K.Park, Y.C.Lai, L.Zhao and N.Ye, "Jamming in complex gradient networks," *Physical Review Letters*, vol.71, 065015-1–065105-4, 2005.

[5] T.Kimura and T.Ikeguchi, "A new algorithm for packet routing problems using chaotic neurodynamics and its surrogate analysis," *Neural Computing and Applications*, vol.16, pp.519–526, 2007.

[6] T.Kimura and T.Ikeguchi, "An optimum strategy for dynamic and stochastic packet routing problems by chaotic neurodynamics," *Integrated Computer-Aided Engineering*, vol.14, pp.307–322, 2007.

[7] T.Kimura and T.Ikeguchi, "An efficient routing strategy with load-balancing for complex networks," *Proceedings of NOLTA2007*, pp.31–34, 2007.

[8] E.W.Dijkstra, "A note on two problems in connection with graphs," *Numerical Mathematics*, vol.1, pp.269–271, 1959.

[9] J.J. Wu, H.J.Sun, and Z.Y.Gao, "Dynamic urban traffic flow behavior on scale-free networks," *Physica A*, vol.307, pp.653–660, 2008.

[10] T.Kimura, H.Nakajima, and T.Ikeguchi, "A packet routing method for complex networks by a stochastic neural network," *Physica A*, vol.376, pp.658–672, 2007.

[11] Y.Sheffi, "Urban Transportation Networks," *Prentice-Hall*, 1985.

[12] A.-L.Barabási and R.Albert, "Emergence of scaling in random networks," *Science*, vol.286, pp.509–512, 1999.

[13] B.M.Waxman, "Routing of Multipoint Connections," *IEEE Journal on Selected Areas in Communications*, vol.6, pp.1617–1622, 1988.