Toward simulating entire cities with behavioral models of traffic

Resilient transportation systems enable quick evacuation, rescue, distribution of relief supplies, and other activities for reducing the impact of natural disasters and for accelerating the recovery from them. The resilience of a transportation system largely relies on the decisions made during a natural disaster. We developed an agent-based traffic simulator for predicting the results of potential actions taken with respect to the transportation system to quickly make appropriate decisions. For realistic simulation, we govern the behavior of individual drivers of vehicles with foundational principles learned from probe-car data. For example, we used the probe-car data to estimate the personality of individual drivers of vehicles in selecting their routes, taking into account various metrics of routes such as travel time, travel distance, and the number of turns. This behavioral model, which was constructed from actual data, constitutes a special feature of our simulator. We built this simulator using the X10 language, which enables massively parallel execution for simulating traffic in a large metropolitan area. We report the use cases of the simulator in three major cities in the context of disaster recovery and resilient transportation.

Introduction

Transportation authorities can rely on traffic simulations [1, 2] to evaluate the effectiveness of a particular action for a particular traffic situation. Traffic simulation can be used to infer the consequences of an action by applying predefined rules that govern the behavior of traffic flows. The models of traffic simulation range from “microscopic” to “macroscopic,” depending on the level of detail [3, 4]. A microscopic model tracks the location of individual vehicles, while a macroscopic model tracks some features of flows such as speed and density. The quality of the decisions that the transportation authority can make often depends on how realistic the model of traffic simulation is.

Microscopic models allow more detailed study and more faithful modeling of transportation systems than their macroscopic counterparts [4]. As a result, there has been a significant amount of effort in building traffic simulators with microscopic models [5–8]. For example, Nishi et al. [9] report that a minute change in the configuration of merging lanes can significantly reduce congestion, but such an impact can be captured only by microscopic models. A macroscopic model does not allow us to directly study the impact of small changes that cannot be represented by that model. In addition, we cannot directly study the impact of traffic control in more detail than what the macroscopic model tracks. For example, if we want to evaluate the impact of traffic control on specific vehicles such as ambulances, a microscopic model would be more appropriate than a macroscopic model.

However, a microscopic model can have considerably more parameters than a corresponding macroscopic model. The values of these parameters, for example, determine how the driver of a vehicle, an agent, chooses the speed, the lane, or the route, as well as the origin and the destination. We need to carefully set these values for individual agents, because they essentially determine the results of the simulation. Calibrating these values is time-consuming and often relies on intuitions and knowledge of an expert on transportation systems [5, 10, 11]. Because of the difficulty of calibration, some of the details are often omitted from microscopic models. For example, the driver of a vehicle is often assumed to take the route that minimizes...
simulation models, but they do not state specifically how to record trajectories of vehicles that are measured with reference [11], which advocates the use of more data to reduce the number of variables that need to be calibrated. However, reference [11] does not propose any specific approach. Goulias et al. [18] recommend collecting probe-car data, which involves the recording of trajectories of vehicles that are measured with the Global Positioning System (GPS) for building realistic simulation models, but they do not state specifically how to use the collected GPS data [18]. We will use the probe-car data to set the values of the parameters that determine how the drivers of vehicles select their routes. Most existing approaches would calibrate these values by iterating a simulation by varying the values of the parameters at each iteration until values are found such that the output of the simulation matches observed values, such as the number of vehicles that travel through a road segment [5, 10, 19].

The detailed model of route choice increases the computational complexity of traffic simulation. For scalability, we use XAXIS (X10-based Agents eXecutive Infrastructure for Simulation), a platform that allows massively parallel execution of an agent-based simulation, which we developed in [20]. However, in [20], the driver of a vehicle always selects the route that has the shortest travel time, and it is unclear whether the traffic simulation with the detailed model of route choice is tractable at the scale of an entire city. We show that Megaffic2 enables a nearly linear scale-up with respect to the number of processor cores. We also evaluate the accuracy of Megaffic2 by comparing the results of the simulation with observed volumes of traffic in Hiroshima, a major city in Japan. These numerical experiments constitute the third contribution of this paper. Comparison of Megaffic2 with other approaches of parallel execution of agent-based simulation, such as in [21–24], with respect to efficiency and scalability is left for future work. In addition, comparison of Megaffic2 with other simulators with respect to accuracy is fertile territory for future work.

The remainder of the paper is organized as follows. We start by presenting our model of traffic simulation and specify which parameters are estimated from probe-car data. We then discuss the use cases of Megaffic2. Finally, we show the results of numerical experiments and remind the reader that resilient communities must often rely on a resilient transportation system that enables quick evaluation, rescue, flight, distribution of relief supplies, and versatile methods for mitigating traffic congestion during disasters or other kinds of emergencies.

**Simulation model**

As mentioned, agent-based simulation of traffic flows tracks the location of each agent, representing the driver of a vehicle, who travels, interacting with other agents, according to various models of their behaviors. Each agent is assigned an origin, a destination, and a departure time according to a model of origin-destination (OD) generation. The simulator creates the agent at the origin at the departure time. The agent chooses a route from the origin to the destination, according to a model of route choice, and travels along that route. The simulator allows the agent to travel along that route, during which the travel speed is changed according to a model of speed selection, and the lane is changed according to a model of lane selection. In this
section, we describe details of the models and present how the parameters of the models are determined.

We build our simulation model from map data, census data, and probe-car data. The map data have the information about road segments and intersections. The census data give a table, which we refer to as an OD table, whose entry represents the number of trips from a subarea of the map to another during each hour of a day. The probe-car data record the trip histories of multiple drivers of vehicles, where a trip history is a sequence of longitude, latitude, and time observed with the GPS. Because the GPS data are prone to error, we use a reliable technique of map-matching with a hidden Markov model [16] to recover the routes that are taken by those drivers of vehicles from the probe-car data. In the following, we refer to those data after map-matching as probe-car data.

We use the technique of \( L_1 \)-regularized Poisson regression with adaptive baseline [17] in our model of OD generation for effective utilization of two data sources that are complementary to each other: (i) the census data, which has the exact number of trips but at the coarse granularity of subareas, and (ii) the probe-car data, which has a small number of sampled trips, but with the exact information about the locations of their origins and destinations. The two sources of data are integrated according to information about landmarks, including hotels and railway stations, that are available in the map data. Prior work [25] uses probe-car data to generate ODs but uses only probe-car data, with availability often limited to particularly small regions. The approach of Morimura and Kato [17] complements this limited availability of probe-car data with census data, which is often available for a greater area. When we cannot create an OD table directly from census data, we might still be able to generate one by simulation based on activity patterns of households, as shown in [26].

The technique of Morimura and Kato [17] gives the probability that a pair of intersections becomes an origin and a destination for each hour of a day. The model of OD generation creates agents together with their origins, destinations, and departure hours according to this probability. Exact departure time is determined uniformly at random within the departure hour, but this can be made more realistic when more detailed data are available.

Our model of route choice determines the route of an agent from its origin to its destination, taking into account three quantities: travel time, travel distance, and the number of turns. Specifically, the route that minimizes the weighted sum of the three quantities is selected, where the weight depends on individual agents. We refer to the weight used for an agent as the agent’s personality.

We can estimate the personality of a driver whose trip is recorded in the probe-car data. In this study, we estimate the personality for each trip recorded in the probe-car data, so that a driver with multiple trips is considered as multiple drivers. We then seek to select the personality that best explains each trip. We refer to the personality thus estimated as observed personality. For personality estimation, we calculate the travel time along a segment of a road from its distance and its speed limit, but this can be made more realistic by estimating the travel time from probe-car data by using a nonparametric estimation technique [27]. The details about the procedure of personality estimation will be reported elsewhere, but related techniques can be found in the literature [28, 29].

To generalize the limited set of observed personality to the personality of all of the agents in the simulator, we fit the set of observed personality to a mixture of Dirichlet distributions. Namely, the personality of an individual agent is generated probabilistically according to the fitted mixture of Dirichlet distributions.

Once the personality of an agent is determined, Dijkstra’s algorithm can be used to find the route, from the agent’s origin to its destination, that minimizes the convex combination of the three quantities under consideration, where his personality is used as the weight in the convex combination. To take into account the number of turns, Dijkstra’s algorithm is run on a network whose vertex represents a segment of a road and whose edge represents a connection from a segment of a road (which we refer to as the first road segment) to a neighboring segment of a road (which we refer to as the second road segment). The edge cost then represents the convex combination of the travel time along the second road segment, the travel distance along the second road segment, and the indicator (zero or one) of whether there is a turn from the first road segment to the second. The travel time used for route selection is updated every 10 simulated minutes according to the simulated travel time.

While an agent travels along its route, it adjusts its speed, depending, for example, on the distance to the preceding vehicle. We use Gipps’ car-following model as our model of speed selection [30]. An agent also changes its lanes, depending, for example, on the turn that it is going to make next and on the space available in the neighboring lane. We use the model of lane selection proposed by Toledo et al. [31]. The parameters in the model of speed selection and those of lane selection are set as recommended in [30, 31].

Use cases
In this section, we apply Megaffic2 to simulating the traffic flows in the following three cities: Sendai in Japan, Nairobi in Kenya, and Rio de Janeiro in Brazil. In these case studies, we run the simulation with simpler models than the full model of Megaffic2, which we use in our study for validation with traffic flows in Hiroshima city. The purpose of this section is not to demonstrate the full capability of Megaffic2 but to show how Megaffic2 can be used to help transportation authorities. Throughout the experiments in this
section, we use a personal computer with 20 GB of memory and two Intel Xeon** E5540 central processing units (CPUs) of 2.53 GHz. The CPUs have eight cores in total.

**Sendai in Japan**

Sendai city suffered from severe damages from the Tohoku earthquake of March 11, 2011. A Japanese newspaper company, Kahoku Shimpo, reported an article on April 30, 2011, that some of the roads above a subway line had been severely damaged and caused heavy traffic congestion in the city. The traffic jam around a hospital was so severe on April 27 that many people could not arrive at the hospital. Traffic congestion in the city center prevents the operation of city functions, such as the police, hospitals, and fire stations. In particular, it appears that ambulances had difficulty in reaching the hospital. We try to reproduce this situation and apply what-if analysis by closing some of the roads in the city center so that only the vehicles for rescue and police can use these roads.

Although Megaffic2 can generate necessary models from a set of GPS data, it was difficult to obtain a sufficient volume of GPS data from Sendai city. To use Megaffic2, we first choose the city center as a simulation area, obtain road network data from the OpenStreetMap (www.openstreetmap.org/), and identify the damaged roads in the road network data. We divide the city map into 3 × 3 rectangular regions of equal size. Then we assign the number of vehicles to run between each pair of regions. In this case study, we calibrate the OD table through preliminary simulations so that the traffic volume inside the city becomes moderate, but this step of calibration can be omitted when census data and probe-car data are available for generating ODs. We simulate the traffic flow of 5,000 seconds in the city with different road closures. There are approximately 34,000 intersections and 70,000 road segments in our selected area, and we generate approximately 43,000 trips. Each run of the simulation required approximately 4 minutes.

**Figure 1** shows the simulation results. The upper left map shows a heat map representing the numbers of vehicles on the roads at 1,000 simulated seconds. Note that colored line segments are not depicted if no car is running on a particular segment of a road. The other four maps show heat maps representing the average speed of vehicles during the last 1,000 simulated seconds, where red, green, and blue...
denote 0 km/hr, 15 km/hr, and 30 km/hr, respectively. No colored line segments are depicted when the average speed is more than 30 km/hr. Three key performance indicators (KPIs) are shown in the heat maps of Figure 1: “number of cars” denotes the number of vehicles at the last simulated second in the whole simulated area; “jam length” denotes the total length, in kilometers, of the roads whose average speed is less than 5 km/hr during the last 1,000 simulated seconds; and “CO₂ emission” denotes the total amount of the CO₂ emission in the whole simulated area during the entire simulated time interval.

We find that traffic demand to the hospital in the city caused traffic congestion on the major road between the hospital and a railway station, where the average speed is less than 10 km/hr and the heatmap color in Figure 1 is orange or red. In this sense, the simulation has reproduced the traffic situation reported in the news article. To find a way to resolve this congestion, we conduct simulations by closing some of the major road segments around the hospital. The road segments that we close in our simulation are those for which MegaCiti2 reported traffic congestion (orange or red heat map, where the average speed is less than 10 km/hr) when the only closed roads are the ones that were damaged above the subway line. We evaluate what will happen if we close some of these originally congested road segments.

“Road closure #3” in Figure 1 suggests that appropriate road closure is effective to reduce the original traffic congestion on the major roads between the hospital and the railway station. On the other hand, we find that the other closure scenarios induce traffic congestion (orange or red heat map) on a neighboring major road. This result suggests that we should carefully design a road closure scenario to deal with traffic congestion around the important facilities in the city center.

**Nairobi in Kenya**

As the economy of Kenya develops, the traffic demands in Nairobi city increase. However, the traffic infrastructure has not been well developed yet. Consequently, Nairobi city suffers from severe traffic congestion. Traffic congestion can be reduced by constructing new roads or by converting roundabouts into intersections or elevated roads. However, since such constructions require a long time, we evaluate whether any road closure can reduce the congestion. Note that road closures are easy to deploy.

We choose the center business district as a simulation area and obtain road network data from the OpenStreetMap. We then split the map into 3 × 3 regions and design an OD table between the regions on the basis of the traffic flow statistics in Nairobi city reported in Figure 6 from [32]. Because we assume that every road has only one lane in each direction in this case study, the volume of traffic flow thus determined is then reduced by a factor of one third, which we find to result in moderate level of congestion. Note that exact number of lanes will be used in our study of validation with traffic flows in Hiroshima city. OpenStreetMap does not provide the information about how to control traffic signals, which hence are ignored in our simulation. One might be able to refine the road network data by the use of the techniques in [33].

Because we have found a news article about a flood in Nairobi, we simulate the case where one of the major roads is disconnected by the flood, which is shown in the upper left map in Figure 2. We prepare four scenarios for what-if simulation: one includes no road closure (the upper right map in Figure 2), which means that the city administrator does nothing to deal with the flood, and the other three scenarios involve road closures (three lower maps). We simulate the traffic flow of 1 hour for each scenario. The upper left map shows a heat map representing the number of vehicles on each road at 600 simulated seconds. Recall that colored line segments are not depicted if no car is running on a particular segment of a road. The other four maps show heat maps representing the average speed of vehicles during the last 600 simulated seconds, where red, green, and blue denote 0 km/hr, 10 km/hr, and 20 km/hr, respectively. No colored line segments are depicted when the average speed is over 20 km/hr. The map has approximately 1,500 intersections and 2,800 road segments, and we generate approximately 10,000 trips in each run of simulation. The simulation of each scenario required approximately 10 seconds. Figure 2 shows three KPIs, which are defined analogously to the KPIs in Figure 1 except that “jam length” here denotes the total length of the roads whose average speed is less than 20 km/hr during the last 600 simulated seconds.

Intuitively, one might expect that road closures will make traffic congestion worse in general. By comparison of the “jam length” KPI in Figure 2, we can actually observe worse traffic congestion in the first two road closure scenarios. On the other hand, we can observe that “Road closure #3” successfully reduces traffic congestion. We have not evaluated the effectiveness of the road closure of the third scenario in the real world, which is left for future work.

**Rio de Janeiro in Brazil**

More and more people in Rio de Janeiro own cars and thus suffer from heavy traffic congestion. In particular, a commute between the suburbs and the city center is a major problem of the city. We interviewed local residents about the traffic in Rio de Janeiro to select a simulation area and design an OD table. We were informed that the elevated road (Elevado da Perimetral) near the city center causes severe congestion. To simulate the traffic flows in the city center, we generate ODs to go to and go through the city center. We calibrate the number of vehicles to run so that there is moderate traffic congestion. We simulate the traffic of 1 hour in two cases. The first case uses the original map. The second case uses a modified map, where a bypass is
constructed along Elevado da Perimetral. We obtain road network data from the OpenStreetMap analogously to the Sendai and Nairobi cases. The map has approximately 5,600 intersections and 7,000 road segments, and we generate approximately 15,000 trips in each run of simulation. The computation time for each scenario was approximately 50 seconds.

The results of simulation in the first case show traffic congestion along major roads, as the local residents informed us. One of the authors actually experienced heavy congestion in some of the roads where the simulation result shows traffic congestion. The simulation result with a bypass along Elevado da Perimetral suggests that the bypass will flow much more vehicles between the end points of the bypass but will cause heavy traffic congestion around the end points. While the jam length (the total road length whose average speed is less than 18 km/hr) of the lane toward the domestic airport of Elevado da Perimetral suggests that the bypass will flow much more vehicles between the end points of the bypass but will cause heavy traffic congestion around the end points. While the jam length (the total road length whose average speed is less than 18 km/hr) of the lane toward the domestic airport of Elevado da Perimetral is approximately 300 m in the scenario without the bypass, that of the scenario with the bypass increases to approximately 1,800 m. This result implies that we should choose the end points of a bypass carefully. More detailed study about where to construct a bypass is left for future work.

**Numerical experiments**

In this section, we evaluate the scalability of Megaffic2 and compare the results of the simulation against corresponding values observed in real traffic. We use the road network of the Hiroshima area in Japan, which contains 23,734 intersections and 65,518 road segments. Specifically, the latitude ranges from 34 to 35 degrees North, and the longitude ranges from 132 to 133 degrees East. The focus of the study in this section is on the model of route selection. Other models of drivers’ behavior certainly have impacts on the scalability and the accuracy of Megaffic2, but such extended study is beyond the scope of this paper.

**Performance evaluation**

In this section, we evaluate the performance of the running time of Megaffic2. Although we could use a supercomputer as we did in [20], we use a commodity cluster with four nodes in this experiment, because the road network and the number of trips are relatively small. Each node has two Intel Westmere EP 2.93-GHz processors (Xeon X5670 having an L2 cache of 256 KB and an L3 cache of 12 MB), with 12 CPU cores in total and 50 GB of local memory. The operating system is Cent OS 5.6. To enable the parallel
processing, we partition the network into multiple domains. A graph partitioning tool, METIS [34], is used to partition the network into up to four domains so that intersections are distributed evenly to each domain.

Figure 3 shows the results of the performance evaluation. The vertical axis shows the elapsed time to simulate the traffic of 1,000 seconds, where we generate 100,000 trips uniformly at random during the simulation period. The elapsed time is evaluated with a varying number of nodes and CPU cores, as is indicated along the horizontal axis. Specifically, a single node with 12 cores is used in the leftmost result, and four nodes with 48 cores are used in the rightmost result. In each setting of CPU, the simulation is run for two cases, which are labeled with “Time only” and “Time, Distance, Turns.” The case with “Time, Distance, Turns” corresponds to the full model of Megaffic2, for which the route of each vehicle is determined and depends on the personality of its driver, taking into account the three quantities: travel time, travel distance, and the number of turns. The case with “Time only” uses a reduced model, for which the route of each vehicle is determined solely on the basis of the travel time. With the full model, the elapsed time is reduced from 1,170 seconds to 576 seconds by increasing the number of nodes from one to four. With the reduced model, the elapsed time is analogously reduced from 447 seconds to 183 seconds.

Observe that Megaffic2 shows nearly linear scale-up for the range of the number of cores studied in Figure 3. Even though the full model requires significantly longer running time than the reduced model, we can observe that this increased running time can be compensated by the use of sufficiently many CPU cores.

Validation
We now study the accuracy of Megaffic2 by comparing the results of simulation with respect to corresponding values observed in the real traffic. We use the probe-car data from 2,954 taxis in the greater Tokyo area on December 9, 2009, to determine the parameters in the model of route selection and those of OD generation. These parameters are then used in simulation of Hiroshima city, which is more than 400 miles away from Tokyo (i.e., outside the greater Tokyo area). An advantage of our approach is that we can transfer the models estimated in one area to simulation in other areas, because these models consist only of abstract elements such as the personality of the drivers of vehicles and the tendency of a type of a landmark to become an origin or a destination. Details about our probe-car data can be found in [27]. Note that unlike the simulation in the abovementioned case studies, we use the full model of route selection, and the OD is generated by the use of L1-regularized Poisson regression with adaptive baseline. In addition, the number of lanes is set as is specified in our map data. However, the traffic signals are ignored.

Figure 4 compares the volume of the simulated traffic along the y axis with respect to that of the observed traffic along the x axis at the 156 observation points with a scatter plot in the log-log scale. For each of the observed traffic and the simulated traffic, we normalize the volume of traffic such that the total volume at the 156 observation points
becomes 1.0. This normalization allows us to isolate the relative difference from the total volume that can vary day to day. The coefficient of correlation between the simulated traffic volume, and the observed one is 0.72, which is unaffected by the normalization. If the simulated traffic volumes were perfectly proportional to the observed ones, the scatter plot would become a 45-degree diagonal line, and the coefficient of correlation would become 1.0.

**Conclusion**

Building a resilient society involves building a resilient transportation system that enables quick evaluation [22, 35], rescue, distribution [36] of relief supplies, and other activities for mitigating the impact of natural disasters and quickly recovering from them. To maintain a functioning transportation system, the transportation authorities must take appropriate actions when they face a natural disaster. For example, they could close certain sections of a road to redirect evacuating vehicles. However, it is not at all obvious what actions are appropriate and what the consequences of an action will be. In addition, the decisions must be made quickly.

We have demonstrated the use of a traffic simulator, Megaffic2, for potentially accelerating the recovery from disaster and for reducing the traffic congestion in developing cities. These use cases have not been well studied in the literature. We expect that our use cases can be used as guidelines for the transportation authority that wants to maintain the function of the transportation system after a natural disaster or that wants to reduce congestion in developing cities.

Megaffic2 is built with a unique design principle. Namely, we use probe-car data to directly estimate some of the parameters of Megaffic2 without regard to the standard approach of calibration, which is known to be quite difficult for a microscopic model of traffic simulation. In particular, we estimate the personality of individual drivers of vehicles in selecting their routes by taking into account travel time, travel distance, and the number of turns. This paper has presented the results of numerical experiments, but more detailed experiments are needed to precisely understand the advantages and the disadvantages of the proposed design principle and of those of the model of route selection with the estimated personality.

The model of Megaffic2 can be further extended to incorporate more detailed behavior of the drivers of vehicles or to incorporate the dependency of the behavior on time. For example, the drivers of vehicles have different sensitivity to the risk of delay [37]. Our model of route selection can be extended in such a way that the drivers of vehicles select different routes depending on the sensitivities to the risk of delay, where the sensitivity is estimated from the probe-car data. Travel time distribution, which will be needed for such estimation, can be fitted by the use of the techniques introduced in [27].

**Acknowledgments**

This research was supported by CREST of the Japan Science and Technology Agency. Megaffic2 was developed by S. Kato, T. Morimura, R. Raymond, S. Suzuki, and R. Takahashi in addition to the authors of this paper.

**References**


Received November 6, 2012; accepted December 17, 2012

Takayuki Osogami IBM Research - Tokyo, 5-6-52 Toyosu, Koto-ku, Tokyo 135-8511, Japan (osogami@jp.ibm.com). Dr. Osogami is a Research Staff Member at IBM Research - Tokyo. He received his Ph.D. degree in computer science from Carnegie Mellon University in August 2005 and his B.Eng. degree in electronic engineering from the University of Tokyo in 1998. A primary focus of his current research is in modeling sequential decision-making under risk and uncertainty as well as estimating the preference of individual decision-makers from their behavioral data. He has filed more than 24 patents and published more than 40 papers in journals and refereed conference proceedings.

Takashi Imamichi IBM Research - Tokyo, 5-6-52 Toyosu, Koto-ku, Tokyo 135-8511, Japan (imamichi@jp.ibm.com). Dr. Imamichi is a researcher at IBM Research - Tokyo. He is a member of the Analytics and Optimization Group. He received the Bachelor of Engineering, Master of Informatics, and Doctor of Informatics degrees from Kyoto University in 2004, 2006, and 2009, respectively. His research interests include cutting and packing problems and metaheuristics. Dr. Imamichi is a member of the Operations Research Society of Japan and the Information Processing Society of Japan.

Hideyuki Mizuta IBM Research - Tokyo, 5-6-52 Toyosu, Koto-ku, Tokyo 135-8511, Japan (mizuta@jp.ibm.com). Dr. Mizuta is a Research Staff Member of IBM Research - Tokyo. He received
B.S., M.S., and Ph.D. degrees in physics from the University of Tokyo. He is a member of Information Processing Society of Japan and ACM SIGSIM (Association for Computing Machinery, Special Interest Group on Simulation and Modeling). His research interests include dynamic economic-social systems with heterogeneous agents, SSME (services science, management, and engineering), and smarter cities for a sustainable society.

**Toyotaro Suzumura**  IBM Research - Tokyo, 5-6-52 Toyosu, Koto-ku, Tokyo 135-8511, Japan (toyoh@jp.ibm.com). Dr. Suzumura received his B.S., M.S., and Ph.D. degrees from the Department of Mathematical and Computing Sciences of Tokyo Institute of Technology (1999, 2001, and 2004, respectively). During his Ph.D. program, he spent about one and half years as a visiting researcher at the San Diego Supercomputing Center, Innovative Computing Laboratory led by Prof. Jack Dongarra, and the Extreme Laboratory of Indiana University. He had been a research fellow of the Japan Society for the Promotion of Science from 2002 to 2005.

**Tsuyoshi Idé**  IBM Research - Tokyo, 5-6-52 Toyosu, Koto-ku, Tokyo 135-8511, Japan (goodidea@jp.ibm.com). Dr. Idé is a Senior Researcher and the manager of the Analytics and Optimization Group at IBM Research - Tokyo. He received M.Sc. and Ph.D. degrees in theoretical physics from the University of Tokyo in 1997 and 2000, respectively. In 2000, he joined IBM Research as a researcher in the field of display technologies. Since 2003, he has been working on data mining research. His current research interest includes knowledge discovery techniques from time-series data.