

DYNAMIC DATA DRIVEN EVENT RECONSTRUCTION FOR TRAFFIC SIMULATION USING SEQUENTIAL MONTE CARLO METHODS

Xuefeng Yan

College of Information and Technology
Nanjing University of Aeronautics and Astro-
nautics
Nanjing, Jiangsu 210016, CHINA

Feng Gu

Department of Computer Science
The College of Staten Island, The City University
of New York
Staten Island, NY 10314, USA

Xiaolin Hu
Carl Engstrom

Department of Computer Science
Georgia State University
Atlanta, GA 30303, USA

ABSTRACT

Simulation models are commonly used to study traffic systems. Accurate traffic predictions need proper characterization of the traffic flow and knowledge of related parameters representing the state of the traffic flow in the models. To correctly estimate the traffic flow in real time, we need to reconstruct the event by answering such critical questions as the source of the congestions. The availability of sensor data from the real traffic provides information that can be assimilated into a traffic simulation model for improving predicted results. In this paper, we use the sequential Monte Carlo methods to assimilate real time sensor data into the simulation model MovSim, an open-source vehicular-traffic simulator, to reconstruct events such as the slow vehicles that cause the traffic jam. Related experimental results are presented and analyzed.

1 INTRODUCTION

One of the most prevalent transportation problems that exist is traffic congestion in urban areas. It causes traffic delays, increased accidents, and higher emissions; furthermore it leads to economic losses. The estimated costs of traffic congestions for some big cities reflect the economic burden on U.S. In 2007, the wasted time and fuel were worth about 9.3 billion dollars in greater Los Angeles region, and the New York metropolitan wasted approximately 13 billion dollars in 2006 (Lewis 2008). Therefore, it is essential to understand the traffic flow and develop road networks with efficient traffic movements that will reduce traffic congestion and decrease its economic costs. In the past decades, researchers have developed various theories and models to study the behaviors of the traffic flow. Most of the models fall into two main categories, including microscopic modeling and macroscopic modeling according to the level of details. The microscopic approach models the traffic flow by analyzing driver-driver interactions and driver-road interactions. By contrast, the macroscopic approach formulates the relationships among traffic flow parameters, such as the density and the average speed of the traffic stream from a global aspect. Some of

these models include car-following models (e.g., Viedemann model), cellular automaton models (e.g., Nagel–Schreckenberg model), and LWR model. Comparisons of related traffic flow models were discussed by Hoogendoorn and Bovy (2001).

Although the abovementioned modeling approaches may provide useful information for decision makers of traffic management to some degree, accurate predictions need proper characterization of the traffic flow (e.g., the location of congestion) and knowledge of related parameters representing the state of the traffic flow in the models. To correctly estimate the traffic flow, we have to reconstruct the event by knowing the critical situations below. Where does the congestion happen? What leads to the congestion? How severe is the congestion? Inaccurate estimation of the information above can lead to delay to reduce the traffic jams. The availability of sensor data from the real traffic provides the measurement to match the output to improve predicted results. For example, a slow vehicle in the road can lead to the congestion. By collecting the real time data (such as, velocity, acceleration, and position of vehicles) and assimilating them into the simulation model, the information of this event of congestion can be obtained (the location of this vehicle). This provides valuable information for decision makers to manage the traffic.

The rapidly developed sensing technologies are providing highly convenient availability of real time data from the system under study like never before. The data from the real system is continually assimilated into the simulation model to improve the simulation results. The assimilated data is a kind of measurement used to evaluate the simulation's output, and then adjust the state of the simulation model. This makes it possible to reconstruct events like locating the vehicle to cause traffic congestions. In this work, we assume that a slow vehicle in the road network results in the traffic jam. The collected real time data is assimilated into the simulation model. By analyzing the difference between the real time data and the output from the simulation model, we locate the vehicle causing traffic congestions. The used simulation model for traffic flow is MovSim, a microscopic lane-based traffic simulator. It implements various car-following models like time-continuous models, iterated maps and cellular automata. Due to the complex, nonlinear, and chaotic behavior of the traffic flow system, the sequential Monte Carlo (SMC) methods are used for data assimilation.

The rest of the paper is organized as follows. Section 2 introduces the related work in various domains. Section 3 formulates the problem of the event reconstruction. Section 4 presents the experiments and their results. Finally, Section 5 draws conclusions and points out the future work.

2 RELATED WORK

In the last fifty years, many scholars have developed a large number of traffic flow models. The developed microscopic models include the Wiedemann model (Wiedemann 1974), Gipps' model (Gipps 1976), Nagel-Schreckenberg model (Nagel and Schreckenberg 1992), and LWR model (Lighthill and Whitham 1955, Richards 1956). The Wiedemann model is a car-following model, which assumes that traffic flow is the process of one vehicle following another. It uses thresholds to decide change of the drivers' behaviors, and incorporates lane-changing and overtaking. The Wiedemann model has been used in extensively used microscopic multi-modal traffic flow simulation software VISSIM (Fellendorf and Vortisch 2010). Gipps' model is another car-following model, which defines a number of parameters to model different features of drivers' behaviors. Since it involves multiple parameters, it is very challenging to choose appropriate values of those parameters for particular road networks. However, the computation costs will be highly increased if repeating the simulations. The road network simulation software PARAMICS (Smith, Duncan, and Druitt 1995) is based on Gipps' model. Comparing to the Wiedemann model and Gipps' model, Nagel-Schreckenberg model is a different microscopic traffic model, in which the cellular automata are used to model the road. In this model, a road is composed of connected cells. Each cell is either empty or contains a vehicle with the velocity. With time advances, the states of the cells evolve. This discrete space and time evolution forms a cellular automaton. It is the basis of the transportation analysis simulation system TRANSIMS (Ley 2009). MovSim is a microscopic road traffic simulator, which im-

plements different car-following models, and even time-continuous models, iterated maps, cellular automata, and lane changes. Its ultimate goal is to develop a generic tool for all implemented models and simulate all basic traffic situations and discrete decision making. It is very convenient for researchers and educators' extension since it is open-source. We also consider MovSim as the simulator in this work. The related information about MovSim can be found at <http://www.movsim.org>. Different from the above microscopic models, macroscopic models attempt to study the average behavior of the traffic flow system instead of individual vehicles. LWR model is one kind of these models, which considers a scalar time-varying, non-linear, hyperbolic partial differential equation. One important assumption is that the velocity depends on the traffic density (Miller 2011).

Event reconstruction has been widely used in various application domains, including hazardous atmospheric releases, radionuclide networks, power systems, and digital investigations. Kosovic et al. (2007) developed an event reconstruction methodology to estimate the probabilistic source terms from field measurement data for accidental and clandestine releases. Brown and Kozlowski (2006) presented the technology and necessity of its application of event reconstruction for the power systems in modern data centers. Koohkan et al. (2012) discussed the potential usage of event reconstruction in international monitoring system radionuclide networks. The digital event reconstruction was examined, and a process model used for digital crimes was proposed by Carrier and Spafford (2004). The model was designed to apply to physical crime scenes and support unique aspects of a digital crime scene. Traditionally, regression, inversion, and optimization are widely used although they have limitations for complex event reconstructions. Especially it is difficult for them to solve the large-scale non-linear systems. Therefore, many researchers seek other approaches for the solutions, such as sampling methodologies. The initial method of Markov Chain Monte Carlo stochastic methodology and the improved methods of SMC methods were adopted for the dynamic data driven event reconstruction for atmospheric releases by Kosovic et al. (2007). Trung (2009) used Kalman filter for the implementation of event reconstruction and energy calibration using cosmic muons for the T2K pizero detector. Due to its convergence to the true posterior density in non-Gaussian, non-linear dynamic systems, SMC methods are employed in numerous dynamic systems. Gu and Hu (2008), Gu, Yan, and Hu (2009), and Yan et al. (2009) explored the dynamic data driven application system for wildfire spread simulation using a discrete event simulation model DEVS-FIRE. Gu (2010) and Xue, Gu, and Hu (2012) presented the overall framework of dynamic data driven application system including associated models and algorithms of data assimilation using the discrete event wildfire spread simulation model of DEVS-FIRE. The underlying technique for their work is SMC methods. In traffic flow systems, SMC methods are a promising technique to assimilate the real time data into the models to improve the estimations. Mihaylova, Boel, and Hegyi (2006) used SMC methods to estimate the traffic on a freeway. They also validated and evaluated the SMC performance using the real traffic data. In addition, the unscented Kalman filter was implemented and its results were compared to those of SMC methods. In our work to be presented in this paper, we attempt to explore event reconstruction for the traffic flow simulation by assimilating real time data into a sophisticated model. Because of their advantages mentioned above, we implement the event reconstruction of the traffic flow system using SMC methods.

3 DYNAMIC DATA DRIVEN EVENT RECONSTRUCTION FOR TRAFFIC SIMULATION

3.1 Structure of Dynamic Data Driven Event Reconstruction for Traffic Simulation

As discussed above, we attempt to assimilate the real time data into traffic flow simulation models to reconstruct events like a slower vehicle in the road to cause the traffic jam. Figure 1 shows the proposed structure of dynamic data driven event reconstruction for the traffic flow simulation. From the figure, we know that several components are needed for the system, including the traffic flow simulation (e.g., the MovSim simulation model), the road networks and vehicle data, the traffic data, and data assimilation methods. The traffic sensors are used to collect real time data from the traffic flow system, such as the

vehicles' speeds, locations, accelerations, and densities. The above data is assimilated into the traffic flow simulation models, such as MovSim, during the simulation of the traffic flow models, using different data assimilation methods (e.g., SMC methods). Therefore, the essential parts of the whole system are the simulation models and data assimilation methods. As mentioned above, although conceptually any traffic flow simulation models and data assimilation methods can be used, the MovSim and sequential Monte Carlo methods are the simulation platform and the approach to explain the structure of dynamic data driven event reconstruction for the traffic flow simulation in this paper. It is noted that although we deal with the general problem of event reconstruction, in this specific paper we consider the event of traffic congestion caused by slowly moving vehicles. To reconstruct the event, we need to model this event explicitly (that is, we introduce a slow vehicle in our simulation model), and then use the SMC methods to reconstruct the location and the moving speed of the slow vehicle. Under this context, the details of related components for the event reconstruction for the traffic flow simulation are provided below.

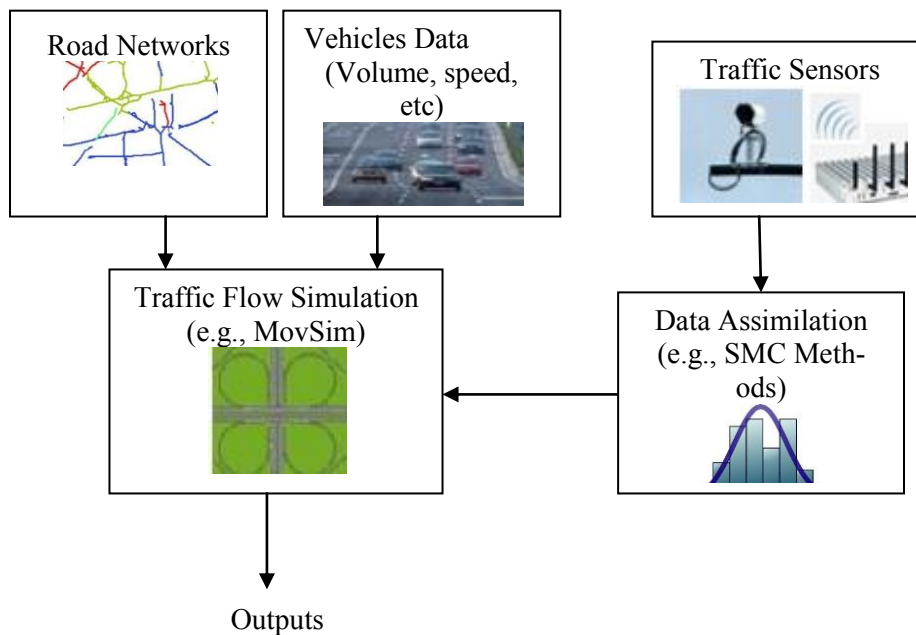


Figure 1: Structure of dynamic data driven event reconstruction for traffic simulation.

3.2 MovSim Simulation Software

MovSim is a microscopic lane-based traffic simulation platform, initiated and contributed by Arne Kesting, Martin Budden, Ralph Germ, and Martin Treiber respectively. MovSim implements different car-following models including time-continuous models, iterated maps, and cellular automata. The time-continuous models include Intelligent Driver Model (IDM) (Treiber, Hennecke, and Helbing 2000), Improved IDM (Kesting, Treiber, and Helbing 2010), Optimal Velocity Model (OVM) (Bando et al. 1995), and Full Velocity Difference Model (FVDM) (Jiang, Wu, and Zhu 2001). The iterated maps contain Gipps' model (Gipps 1976) and Krauss model (Krauss 1998). The cellular automata models consist of Nagel-Schreckenberg model (Nagel and Schreckenberg 1992), improved Nagel-Schreckenberg model, and Kerner-Klenov-Wolf model (KKW) (Kerner, Klenov, and Wolf 2002). It not only models and simulates the basic traffic flow simulations, but also incorporates the discrete decisions, such as lane changes, traffic lights reacting, and yielding. The discrete decision models include lane change with MOBIL

(Kesting, Treiber, and Helbing 2007) and approaching traffic lights. The software can be run from command line or with a graphical user interface. It is an open-source project written in Java. Figure 2 shows a screenshot of a defined example using MovSim. In the figure, a straight two-lane road and a loop two-lane road are defined, and they are connected by two connecting roads. The roads are shown in gray. The cars (shown in black) with their configurations are employed in the road network. When the simulation is run, the cars will behave according to the used models, such as the car-following model. The related output parameters can be captured for analysis. Since it is an open-source software package, MovSim can be used for further development and researchers' use. In addition, the different implemented models provide convenience and flexibility for scholars to meet their requirements in the research. In this work, the real time data is assimilated into the traffic simulation model, therefore, the MovSim simulator will be called in different simulation time steps.

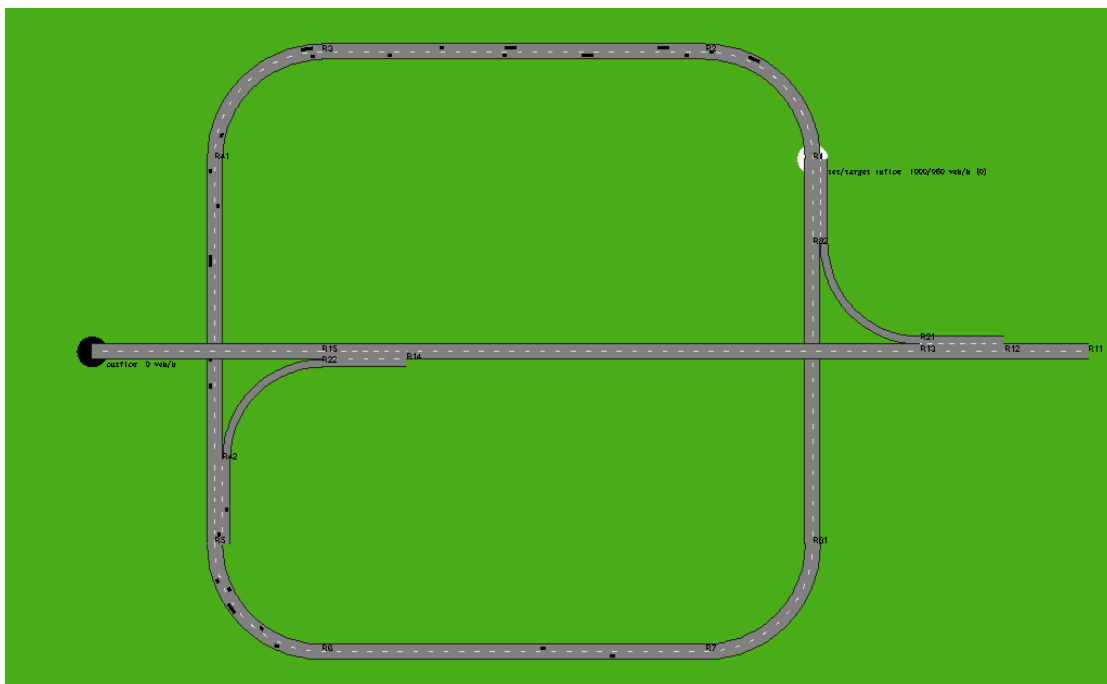


Figure 2: Interface of MovSim.

3.3 SMC Methods

In SMC methods, particles and associated weights are used to approximate the dynamic systems. SMC methods are based on samples that use Bayesian inference and stochastic sampling techniques to recursively estimate the state of dynamic systems from given observations. A dynamic system is defined as a discrete dynamic state-space model, which contains the system transition model in Equation (1) and the measurement model in Equation (2) (Jazwinski 1970) as shown below.

$$s_{t+1} = f(s_t, t) + \gamma_t. \quad (1)$$

$$m_t = g(s_t, t) + \omega_t. \quad (2)$$

t : time step;

- s_t and m_t : the state variable and the measurement variable respectively;
- f : the evolution of the state variable;
- g : the mapping from the state variable to the measurement variable;
- γ_t and ω_t : independent random variables representing the state noise and the measurement noise.

The estimate of the state s_t is based on the set of all measurements $m_{1:t} = \{m_i, i=1,2,\dots,t\}$. In Bayesian filtering, both the state and the measurement variables are stochastic, and the posterior density $p(s_t | m_{1:t})$ is recursively obtained. Given the posterior probability density $p(s_{t-1} | m_{1:t-1})$ at time step $t-1$, the prior probability density $p(s_t | m_{1:t-1})$ of the state at time step t can be computed since the needed system transition density is constructed from the system transition model as shown in Equation (1). If the measurement at time step t is available, the posterior probability density function can be calculated according to Bayes theorem from the measurement model in Equation (2). SMC methods approximate the posterior probability density function $p(s_t | m_{1:t})$ by a set of particles (samples) and their corresponding weights.

One of the most popular algorithms for SMC method is called sequential importance sampling with resampling (SISR), which is the adopted method in this paper. In summary, a basic SMC algorithm that implements the SISR procedure goes through multiple iterations. In each iteration, the algorithm receives a sample (particle) set S_{t-1} and an observation m_t . S_{t-1} represents the previous belief of the system state where $|S_{t-1}| = n_p$ and n_p is the number of samples. In the importance sampling step, each sample $s_{t-1}^{(i)} \in S_{t-1}$ is used to predict the next state. This is done by sampling from the proposal density $q(s_t^{(i)} | s_{t-1}^{(i)}, m_t)$. The importance weight of each particle is then updated and normalized. In the resampling step, n_p offspring samples are drawn with a probability proportional to the normalized sample weights. These samples represent the posterior belief of the system state and are used for the next iteration. With SMC methods, it has been shown that a large number of particles are able to converge to the true posterior density even in non-Gaussian, non-linear dynamic systems (Crisan 2001). For systems with strongly non-linear behaviors, SMC methods are more effective than the widely used Kalman filter and its various extensions. More details about the algorithm were presented by Gordon, Salmond, and Smith (1993).

3.4 Formulating Event Reconstruction for Traffic Flow Simulation

Based on the context of reconstructing an event of a slow vehicle, we describe the system model below. Assume there is a slow vehicle whose position and velocity are $p_slowvehicle(t-1)$ and $v_slowvehicle(t-1)$ at time step $t-1$ in a road network, which is composed of n road segments. The densities of the road segments at time step $t-1$ are $d1(t-1), d2(t-1), \dots, dn(t-1)$ respectively. Given the system status at time step $t-1$, $\mathbf{RW}(t-1) = \{d1(t-1), d2(t-1), \dots, dn(t-1), p_slowvehicle(t-1), v_slowvehicle(t-1)\}$, the MovSim simulator is run for one time step, and we can obtain the system status at time step t , $\mathbf{RW}(t) = \{d1(t), d2(t), \dots, dn(t), p_slowvehicle(t), v_slowvehicle(t)\}$. The system transition is denoted in Equation (3).

$$\mathbf{RW}(t) = \text{MovSim}(\mathbf{RW}(t-1)). \tag{3}$$

To evaluate the status of vehicles, the measurement model is needed to convert from the given status to comparable data with the real time data. The sensors can capture the real time data, such as densities of the vehicle on the road segments. In this problem, we assume k sensors are deployed in the predefined locations of the road network, and their readings at time step t are $\mathbf{S}'(t) = (s1'(t), s2'(t), \dots, sn'(t))$. These data are used to compare with the data from the measurement model, $\mathbf{S}(t) = (s1(t), s2(t), \dots, sn(t))$. The measurement model maps the system status $\mathbf{RW}(t)$ to the data $\mathbf{S}(t)$ comparable with the sensor data from distributed sensors $\mathbf{S}'(t)$. Therefore, we define the measurement model in Equation (4). In this particular context, it is easy to calculate the vehicle densities in the predefined k locations if we know the vehicle positions on road segments. This is implemented by the measurement model MM .

$$\mathbf{S}(t)=MM(\mathbf{RW}(t)). \quad (4)$$

Combining the system model, measure model, and the related noises, the discrete dynamic state-space model is defined in Equation (5) and Equation (6). In the equations, $\mathbf{RW}(t)$ and $\mathbf{RW}(t+1)$ are the statuses of the vehicles at time t and $t+1$ respectively; $MovSim$ is the system model; $\mathbf{S}(t)$ is the measurement data computed from the system status; MM is the measure function mapping from the system status to the measurement data; $\gamma(t)$ and $\omega(t)$ are the system model status noises and the measurement noises respectively.

$$\mathbf{RW}(t+1)=MovSim(\mathbf{RW}(t))+\gamma(t). \quad (5)$$

$$\mathbf{S}_t=MM(\mathbf{RW}(t))+\omega(t). \quad (6)$$

Based on the discrete dynamic state-space model for the traffic flow simulation, the related algorithms of SMC methods can be applied, recursively assimilating the real time data into the system model of $MovSim$ to improve the state estimations.

Weight computation is a very important step in SMC methods since it provides the measure to keep the optimal particles for the future steps. In this paper, we assume n sensors are uniformly distributed on the road network, which can detect the densities of the vehicles within L meters at their left side and right side respectively. For any sensor i at time step t ($t > m$), the history information from time step $t-m+1$ to $t-1$ and the current information at time step t are considered. The differences between the real time data and the measurement data are computed, and their average absolute values are used for measuring the importance of this sensor as shown in Equation (7), where $as(i, t)$ is the weight of sensor i at time step t , $d'(i, t)$ is the real density from the sensor i at time step t , and $d(i, t)$ is the calculated density of sensor i from the measurement model at time step t . Therefore, the weight of a particle $w(t)$ at time step t is defined in Equation (8). In the equation, if $as(i, t)=0$, we directly set $1/as(i, t)$ as 1, where $0 \leq i \leq n$.

$$as(i, t)=(|d'(i, t)-d(i, t)|+|d'(i, t-1)-d(i, t-1)|+\dots+|d'(i, t-m+1)-d(i, t-m+1)|)/m. \quad (7)$$

$$w(t)=\text{Max}(1/as(1, t), 1/as(2, t), \dots, 1/as(i, t), \dots, 1/as(n, t)). \quad (8)$$

4 EXPERIMENTS AND RESULTS

The identical-twin experiment is widely used in dynamic data driven event reconstruction and data assimilation research. It is also adopted in this paper to evaluate the dynamic data driven event reconstruction for traffic flow simulation. It aims to study the dynamic data driven event reconstruction in ideal situations and evaluate the proximity of the prediction to the true states in a controlled manner. A simulation is run, its related data is recorded, and the results are believed as “true” and regarded as the real time data. This “real” time data is assimilated into the dynamic data driven event reconstruction system to predict the system states. Consequently, we check whether the predicted states are close to the “true” states.

We assume there is a slower vehicle on a ring road network (e.g., 1/3 of the other vehicles' speed in the traffic flow), which causes the possible traffic congestion. We run a simulation, and record the data of positions at each time step. This data is treated as “real” data, which will be assimilated into the dynamic data driven event reconstruction system. We name this data as the real traffic. Although we don't know the information of the slower vehicle, the real time data is assimilated into the traffic model for event reconstruction to reconstruct the traffic flow. The goal of the experiment is to show that the reconstructed traffic is proximate to the real traffic.

We use a ring road with length of 1,000 m, and there are 44 vehicles travelling on it as shown in Figure 3. There are two types of vehicles on the road. One type is the normal vehicles (displayed in black), whose maximum velocity is 35 km/h. Another type is the slower vehicle (displayed in red), whose maximum velocity is 10 km/h. The road is divided into 20 road segments with the same length. In Figure 3, the slower vehicle is longer than the normal vehicles. Three density sensors A, B, and C are uniformly deployed on the road, which can detect the vehicle densities within some distance (e.g., 50 m) left and right

respectively. From the figure, we know that because of the slower vehicle, the traffic congestion occurs in the road segment displayed in red. By assimilating the real time data, we reconstruct the event of the slower vehicle on the ring road. In the simulation, the number of particles is set to be 30. We run the simulation for 180 time steps (2 seconds each time step). Figure 4 and Figure 5 display the simulation results at time step 120 and time step 180 respectively. In both of the figures, the horizontal axis and vertical axis represent the road segments and number of vehicle per 50 meters respectively. The real traffic is obtained by assuming we know the information of the slower vehicle denoted as TFS. From Figure 4, we can see that the number of vehicles in segment 4 is relatively high, and probably leading to the congestion. In Figure 5, there are more vehicles in segment 13, 14, 15, and 18, which are the possible congestion parts of the road. In both figures, the vehicles densities of the reconstructed traffic flow (DDDTFS) are proximate to those of the real traffic (TFS). Particularly, the results at time step 180 are better than those at time step 120. From the results of both Figure 4 and Figure 5, we can conclude the reconstructed traffic is very close to the real traffic. Therefore, according to the reconstructed traffic without knowing the slower vehicle, we can estimate its information by assimilated the real time data.

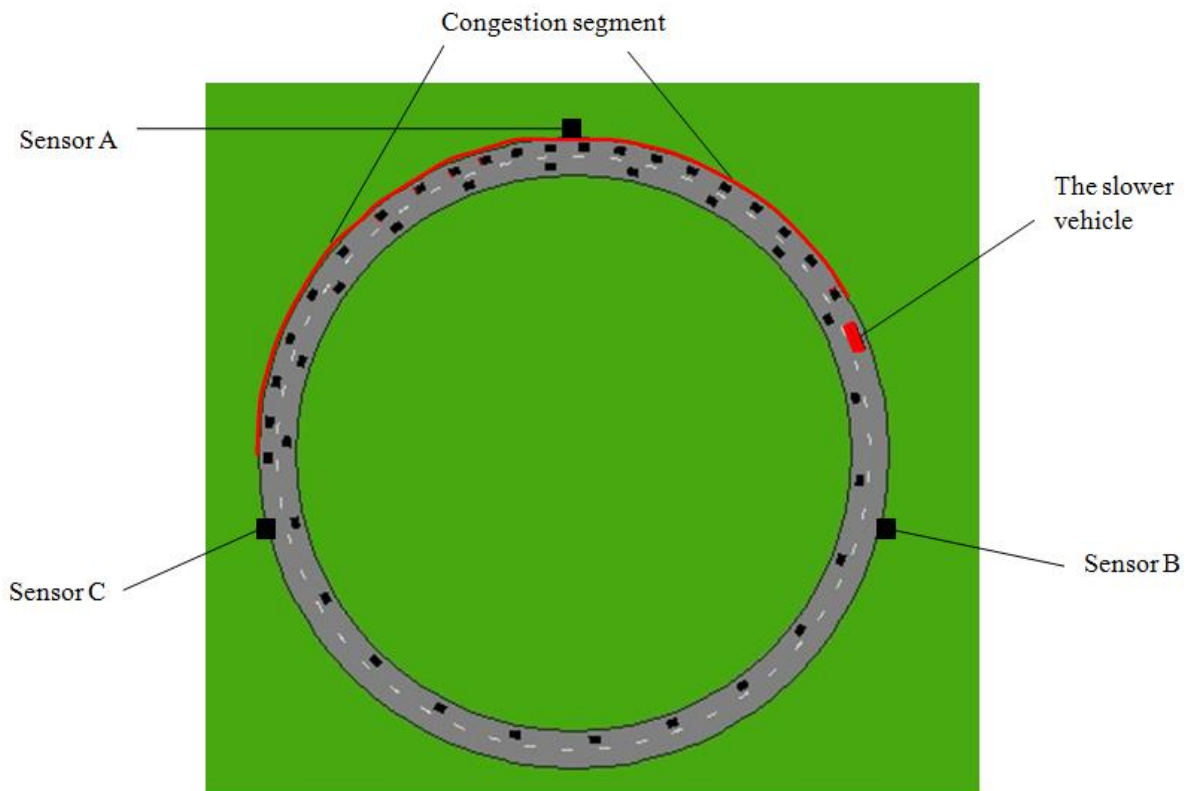


Figure 3: Ring road network with vehicles.

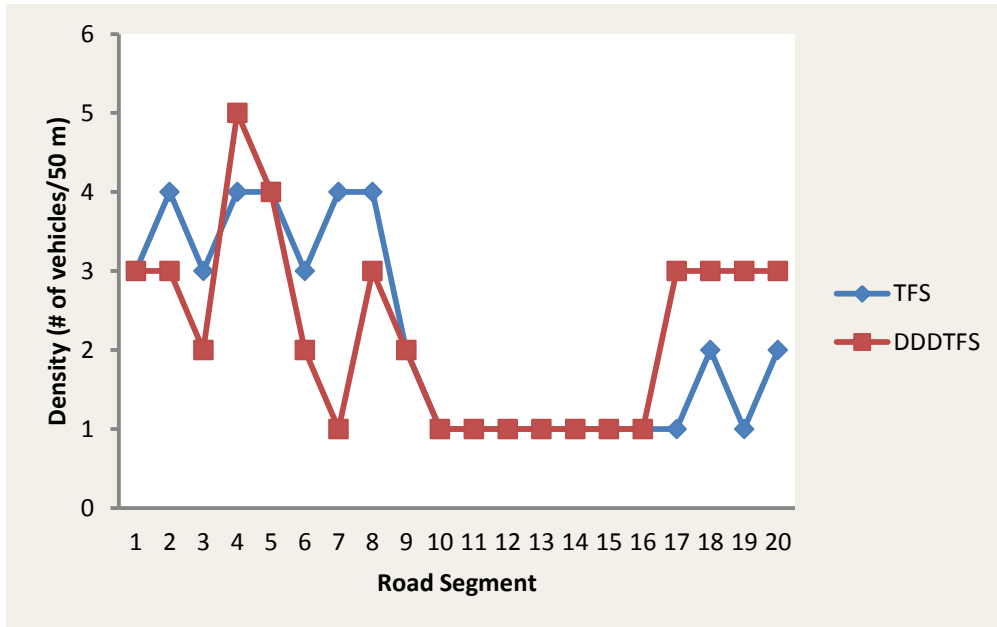


Figure 4: Density of vehicles at time step 120.

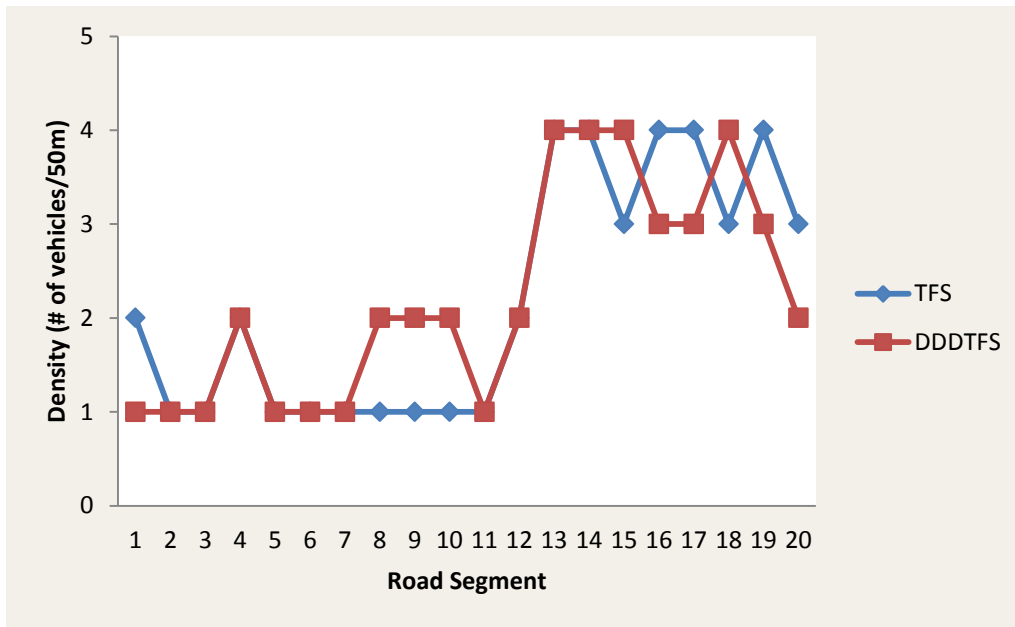


Figure 5: Density of vehicles at time step 180.

5 CONCLUSIONS AND FUTURE WORK

In the paper, the dynamic data driven event reconstruction for traffic flow simulation is presented. Based on the traffic flow simulator MovSim, the structure of dynamic data driven event reconstruction system is described. Finally, the experimental results show that the event (a slower vehicle) can be reconstructed by assimilating the real time data into the traffic flow simulation model. Future work will be focused on the following aspects: (i) Explore other dynamic data driven event reconstruction applications in traffic flow

simulations, such as parameters estimation; (ii) Develop more advanced data assimilation methods to improve the simulation results; (iii) Conduct quantitative analysis about the effects of different parameters of SMC methods on the quality simulation results.

REFERENCES

- Bando, M., K. Hasebe, A. Nakayama, A. Shibata, and Y. Sugiyama. 1995. "Dynamical Model of Traffic Congestion and Numerical Simulation." *Phys. Rev. E* 51: 1035.
- Brown, B., and M. Kozlowski. 2006. "Power System Event Reconstruction Technologies for Modern Data Centers." Square D Critical Power Competency Center.
- Carrier, B. D., and E. H. Spafford. 2004. "Defining Event Reconstruction of Digital Crime Scenes." *Journal of Forensic Science* 49(6): 1291-1298.
- Crisan, D. 2001. "Particle Filters—A Theoretical Perspective." *Sequential Monte Carlo Methods in Practice*. New York: Springer-Verlag.
- Fellendorf, M., and P. Vortisch. 2010. "Microscopic Traffic Flow Simulator VISSIM." *Fundamentals of Traffic Simulation*. New York: Springer.
- Gipps, P. G. 1976. "Computer Program MULTSIM for Simulating Output from Vehicle Detectors on a Multi-Lane Signal Controlled Road." Transport Operations Research Group Working Paper No. 20, University of Newcastle-Upon-Tyne.
- Gordon, N. J., D. J. Salmond, and A. F. M. Smith. 1993. "Novel Approach to Nonlinear/Non-Gaussian Bayesian State Estimation." In *IEE Proceedings on Radar and Signal Processing* 140: 107-113.
- Gu, F. 2010. "Dynamic Data Driven Application System for Wildfire Spread Simulation. Ph.D. thesis, Department of Computer Science, Georgia State University, Atlanta, Georgia.
- Gu, F., and X. Hu. 2008. "Towards Applications of Particle Filters in Wildfire Spread Simulation." In *Proceedings of the 2008 Winter Simulation Conference*, edited by S. J. Mason, R. R. Hill, L. Mönch, O. Rose, T. Jefferson, J. W. Fowler, 2852-2860. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.
- Gu, F., X. Yan, and X. Hu. 2009. "State Estimation Using Particle Filters in Wildfire Spread Simulation." In *Proceedings of the 2009 Spring Simulation Multiconference*, No. 34. Society for Computer Simulation International.
- Hoogendoorn, S. P., and P. H. L. Bovy. 2001. "State-of-the-art of Vehicular Traffic Flow Modeling." *Journal of Systems and Control Engineering – Special Issue on Road Traffic Modeling and Control*: 283-303.
- Jazwinski, A.H. 1970. "Stochastic Processes and Filtering Theory." *Mathematics in Science and Engineering*. Academic Press, New York, USA.
- Jiang, R., Q. Wu, and Z. Zhu. 2001. "Full Velocity Difference Model for a Car-following Theory." *Physical Review E* 64 1: 017101.
- Kerner, B., S. L. Klenov, and D. E. Wolf. 2002. "Cellular Automata Approach to Three-phase Traffic Theory." *Journal of Physics A* 35: 9971-10013.
- Kesting, A., M. Treiber, and D. Helbing. 2007. "General Lane-changing Model MOBIL for Car-following Models." *Transportation Research Record: Journal of the Transportation Research Board* 1999 (1): 86-94.
- Kesting, A., M. Treiber, and D. Helbing. 2010. "Enhanced Intelligent Driver Model to Access the Impact of Driving Strategies on Traffic Capacity." *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 368 (1928): 4585-4605.
- Koohkan, M. R., M. Bocquet, L. Wu, and M. Krysta. 2012. "Potential of International Monitoring System Radionuclide Network for Inverse Modelling." *Atmospheric Environment* 54: 557-567.
- Kosovic, B., R. Belles, F. K. Chow, L. D. Monache, K. Dyer, L. Glascoe, W. Hanley, G. Johannesson, S. Larsen, G. Loosmore, J. K. Lundquist, A. Mirin, S. Neuman, J. Nitao, R. Serban, G. Sugiyama, and

- R. Aines. 2007. "Dynamic Data-Driven Event Reconstruction for Atmospheric Releases." No. UCRL-TR-229417. Lawrence Livermore National Laboratory (LLNL), Livermore, CA.
- Krauss, S. 1998. "Microscopic Modeling of Traffic Flow: Investigation of Collision Free Vehicle Dynamics." PhD dissertation. Universitat zu Koln.
- Lewis, D. 2008. "America's Traffic Congestion Problem: Toward a Framework for National Reform." Discussion Paper. Washington, D.C.: Brookings Institution, The Hamilton Project.
- Ley, H. 2009. TRANSIMS Training Course at TRACC Transportation Research and Analysis Computing Center. Accessed February 24. [http://transims.googlecode.com/svn/v4/trunk/documentation/training/TRANSIMS%20-%201%20-%20Introduction%20\(final\).pdf](http://transims.googlecode.com/svn/v4/trunk/documentation/training/TRANSIMS%20-%201%20-%20Introduction%20(final).pdf).
- Lighthill, M. J., and G. B. Whitham. 1955. On Kinematic Waves II: A Theory of Traffic Flow on Long, Crowded Roads." In *Proceedings of The Royal Society of London Ser, A* 229, 317-345.
- Mihaylova, L., R. Boel, and A. Hegyi. 2006. "Freeway Traffic Estimation within Particle Filtering Framework." *Automatica* 43: 290-300.
- Miller, S. 2011. Traffic Modeling. Accessed February 14. http://home2.fvcc.edu/~dhicketh/DiffEqns/Spring11projects/Scott_Miller/Project.pdf.
- Nagel, K., and M. Schreckenberg. 1992. "A Cellular Automaton Model for Freeway Traffic." *Journal de Physique I* 2(12): 2221-2229.
- Richards, P. I. 1956. "Shock Waves on the Highway." *Operation Research* 4: 42-51.
- Smith, M., G. Duncan, and S. Druitt. 1995. "PARAMICS: Microscopic Traffic Simulation for Congestion Management." In *Dynamic Control of Strategic Inter-Urban Road Networks, IEE Colloquium on*, 8-1-8-3.
- Treiber, M., A. Hennecke, D. Helbing. 2000. "Congested Traffic States in Empirical Observations and Microscopic Simulations", *Physical Review E* 62 (2): 1805-1824.
- Trung, L. 2009. "Event Reconstruction and Energy Calibration Using Cosmic Muons for the T2K Pizero Detektor." Ph.D. Thesis. Stony Brook University.
- Wiedemann, R. 1974. "Simulation des Straßenverkehrsflusses." Schriftenreihe des Instituts für Verkehrswesen der Universität Karlsruhe, Heft 8.
- Xue, H., F. Gu, and X. Hu. 2012. "Data Assimilation Using Sequential Monte Carlo Methods in Wildfire Spread Simulation." *ACM Transactions on Modeling and Computer Simulation (TOMACS)* 22(4): 23.
- Yan, X., F. Gu, X. Hu, and S. Guo. 2009. "A Dynamic Data Driven Application System for Wildfire Spread Simulation." In *Proceedings of the 2009 Winter Simulation Conference*, edited by M. D. Rossetti, R. R. Hill, B. Johansson, A. Dunkin, and R. G. Ingalls, 3121-3128. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

AUTHOR BIOGRAPHIES

XUEFENG YAN is an Associate Professor in the College of Information Science and Technology at Nanjing University of Aeronautics and Astronautics in China. His research interests include modeling and simulation, computer networking, and grid computing. His email is xuefeng.yan@gmail.com.

FENG GU is an Assistant Professor in the Department of Computer Science at The College of Staten Island, The City University of New York in USA. He received his Ph.D. degree in Computer Science from Georgia State University. His research interests include modeling and simulation, complex systems, and high performance computing. His email address is fenggu9083@gmail.com.

XIAOLIN HU received his Ph.D. degree from the University of Arizona. He is an Associate Professor in the Department of Computer Science and Director of the Systems Integrated Modeling and Simulation (SIMS) Lab at Georgia State University. Dr. Hu's research interests include modeling and simulation,

Yan, Gu, Hu, and Engstrom

agent and multi-agent system, and complex systems science. His email address and website are xhu@cs.gsu.edu and <http://www.cs.gsu.edu/xhu/> respectively.

Carl Engstrom is a Master student in the Department of Computer Science at Georgia State University. He received his B.S. degree from Georgia State University. His research interest is modeling and simulation. His email address is cengstrom1@student.gsu.edu.