

TECHNIQUES FOR RAPID INITIALIZATION IN IN-VEHICLE TRAFFIC SIMULATORS

Vishwanath Palagummi

Michael Hunter

Richard Fujimoto

Computational Science and Engineering

School of Civil and Environmental Engineering

Georgia Institute of Technology
Atlanta, GA 30332, USA

Georgia Institute of Technology
Atlanta, GA 30332, USA

ABSTRACT

On-line in-vehicle traffic simulation has been proposed as a means to provide predictions of future states of a traffic network based on current traffic conditions. The area covered by an in-vehicle simulation may change dynamically during the vehicle's journey. This paper is concerned with the issue of initializing the state of new regions that are added to a running microscopic traffic simulation. A warm-up period is required where the new road segments must be populated with simulated vehicles. Techniques that convert vehicle flow rate and queue length information to vehicle positions and speeds are proposed to minimize this warm-up period. These techniques are evaluated and compared with respect to estimated travel times of vehicles traveling through different paths in the road network under different traffic conditions. Travel time predictions are compared to a microscopic simulation of the entire region.

1 INTRODUCTION

The last decade has seen dramatic advances in the development and deployment of technologies for sensors, mobile computing, and wireless communications. Use of on-line simulation as a means to better manage and control operational systems has been receiving increased attention. This trend is particularly true in transportation systems where vehicles are now routinely equipped with global positioning system (GPS) devices, route planning software, and on-line feeds of current traffic conditions derived from instrumented roadways. On-line simulations, fed by real-time data, can rapidly predict future system states for use in planning and management for individual vehicles, and potentially, for the system as a whole. Here, we are concerned with on-line simulations embedded in vehicles in order to predict future states of portions of the transportation network that are of interest to the traveler.

We envision collections of mobile in-vehicle simulators, each modeling some portion of the road network. The road network may be viewed as being divided into a set of non-overlapping zones or grids, and each in-vehicle simulation selects certain grids to include in its model. For example, the vehicle may choose to model the next few grids in which it intends to travel on its current trip. As the vehicle moves, it may wish to add new grids into the area it is modeling, while deleting grids for areas in which it no longer has an interest, e.g., because it has completed travel through those regions. The addition of a new grid to the running simulation requires specification of the initial state for the new region. The required information includes relatively static information (such as road topology) to dynamic, but somewhat predictable, information (such as signal control) to rapidly changing, less predictable dynamic information (such as vehicle flow rate data or information concerning individual vehicles, i.e. exact position, current acceleration, etc). Here, we are concerned with this latter information, and specifically, determination of initial state information of microscopic traffic simulations based on current flow rate information derived from infrastructure sensors or through other means such as in-vehicle sensing. We assume topology, signal timing and routing information is available from the infrastructure or historical data. The problem addressed here concerns the establishment of initial vehicle position and speed to populate a microscopic traffic simulation based on on-line data reporting vehicle flow rates on segments of the region that is to be added.

2 RELATED WORK

Warm start and warm-up intervals are a well researched area in the field of simulation. For example, (Fishman 2001) explores this problem to determine the length of the warm-up interval and the accuracy of samples obtained in the warm up period in discrete event simulations. This work treats the warm start problem independent of the application domain, and thus does not deal with the problem of initializing transportation simulations.

Microscopic transportation simulations model traffic at the detailed level of individual vehicles. Details concerning traffic signals and routing decisions of individual vehicles are explicitly modeled. Examples include CORSIM (ITT Systems 1998), VISSIM (PTV 2005), MITSIM (Balakrishna et al. 2005), and TRANSIMS (Fisher 2000), among others. Microscopic traffic simulation models are well suited for modeling urban areas such as signalized intersections, conflict areas, interchanges and regions with public transit systems. Microscopic models evolved from car following models. The cellular automaton model proposed in (Nagel and Schreckenberg 1992) and also used in TRANSIMS represents an approach that divides road segments into cells, where each cell could accommodate a single vehicle. Laws of physical motion were applied to quantize the movement of vehicles in terms of the cells. The model was initially developed for a single lane but was extended to multiple lanes. Microscopic simulations have evolved to account for more complex driving behavior models such as psycho-physical car-following models (Wiedemann 1991).

The usage of microscopic simulations for large scale urban traffic scenarios is challenging due to the large number of microscopic elements. One approach to support such large scale simulations is to parallelize the simulation model. A parallelization approach through spatial decomposition is discussed in (Nagel and Rickert 2001), while (Kiesling and Lüthi 2005) discusses using time parallelization techniques.

Ad-hoc simulation systems described in (Fujimoto et al. 2007) are an approach to embedded on-line simulation such as those proposed here. It involves distributing the simulation over autonomous interacting simulators. This approach allows simulators to operate in close proximity to the data collection points, reducing the cost of centralization. The absence of a centralized control point allows simulators to operate on overlapping simulation regions allowing better accuracy and greater robustness. This paper builds upon this approach by focusing on the development of an in-vehicle simulator that would eventually form a component of an ad-hoc vehicle traffic simulation system.

3 SIMULATION COMPONENTS, RESOLUTION AND METHODOLOGY

The in-vehicle simulation is presented in the context of its use in an ad hoc distributed simulation system. However, the initialization problem addressed in this paper is applicable to any in-vehicle simulator that must dynamically add new areas to the model during its execution. When a simulation is started, the available road network is divided into a set of non-overlapping *simulation windows*. In the current road topology model, a *road link* is identified as the fundamental level of resolution within a simulation window. A road link is a road section between intersections with other road links, and is specified by the starting and ending co-ordinates, length and number of lanes. Each simulation window can be defined as a collection of road links that pass through a zone of interest. Each simulation windows maintains the information of road links within the region. The current implementation creates fixed size rectangular grids with the Cohen Sutherland line clipping algorithm (Foley et al., 1990) from the existing road network (defined by the bottom left and top right corners); though grids of regular convex shapes of varying sizes are supported. Considering the direction of traffic, the points at which a road enters and leaves the simulation window defines the entry and exit points of the link. The fragment of road between the entry and exit points specifies the simulated region of the road for a window.

An *active region* in the simulation is defined as a set of simulation windows that are currently simulated by the vehicle. The road links that are a part of the active region (entirely within the region or passing through the region) are *active links*. The simulation model subscribes for input data pertaining to the active links. Since a link might span across simulation windows, the related links are combined, i.e. the entry and exit points for an active link are calculated for the entire active region. For active links that lead outside the simulated region, vehicular traffic beyond the exit point of a link is periodically removed from the simulation model. Similarly, input traffic is generated only at the entry point of an active link.

The overall in-vehicle simulation system consists of the following components shown in Figure 1:

- The Space Time Memory (STM) is embedded in the infrastructure and provides updated information concerning the flow rates, queue length and mean vehicular speed on links (road segments) in the road network. In addition to current state information such as flow rates, it includes a prediction of future flow rates on links computed by other in-vehicle simulators.
- The Navigational Unit (NU) provides the current position of the vehicle in the road network, from which the portion of the network modeled by the in-vehicle simulator is derived.

- The Simulation Engine receives input from the STM and NU and performs a microscopic traffic simulation; here, a commercial transportation simulation VISSIM is used.

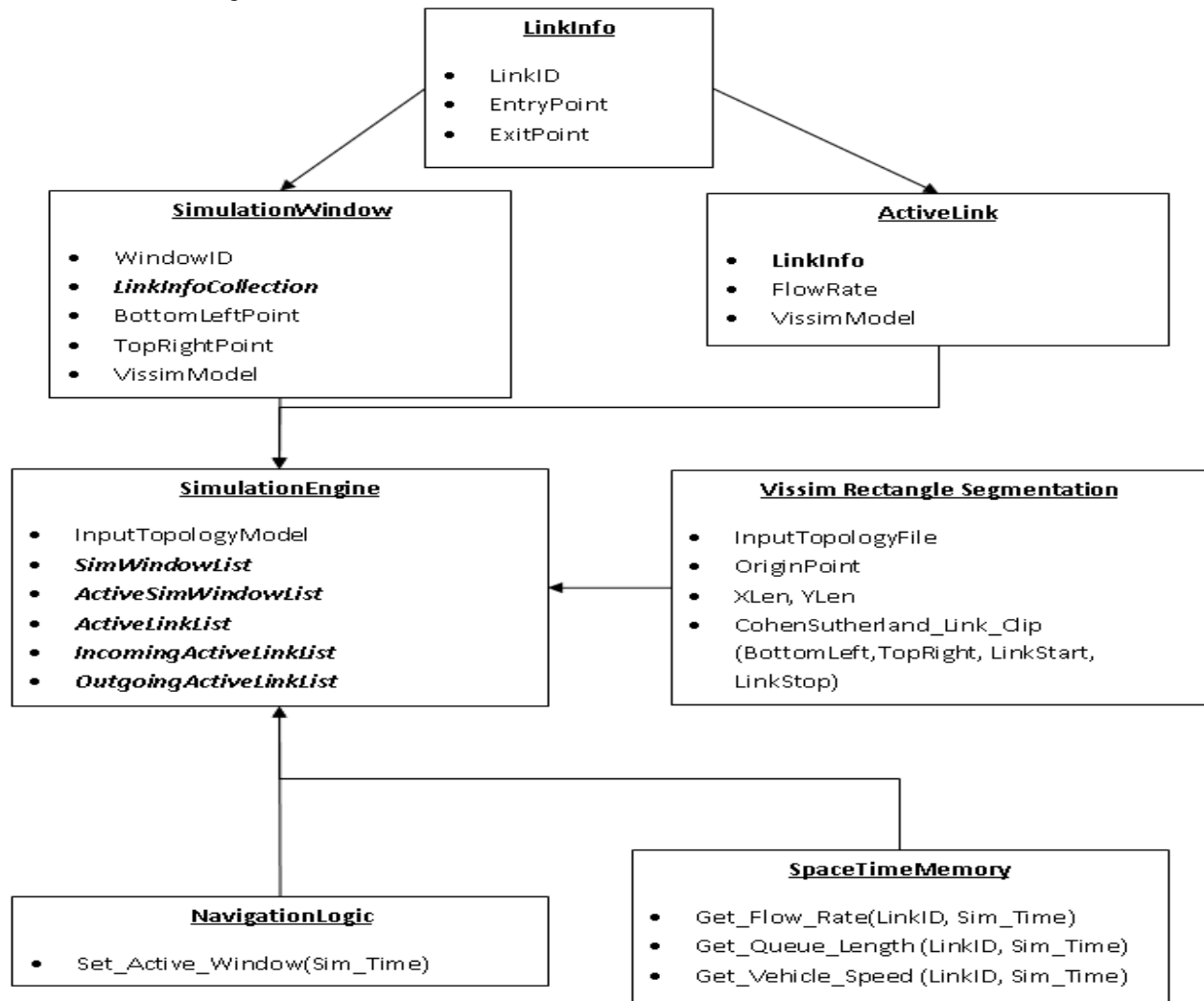


Figure 1: Simulation components

Other data structures shown in Figure 1 describe the simulation windows, links and active links, and specification of the segmentation scheme.

The in-vehicle simulation system computes the future state of the portion of the transportation network it is modeling based on the current state of the system. The space-time memory contains statistics that capture the current state of the network. The navigation logic holds the current position of the vehicle, from which the topology segmentation component constructs the portion of the network modeled by this simulator. Specifically, segmentation windows are created, one for each grid sector modeled by the simulator. The simulation engine is assumed to have projected the future state of this portion of the system over the desired period of time into the future. The simulator then performs the following steps:

1. Update the set of active windows (grids) for the current simulation time
 - (a) Remove any previously active windows that no longer need to be simulated
 - (b) Add any new active windows to be simulated
 - (c) Update the set of active links and aggregate them.
 - (d) Populate new portions of active links.
2. Receive updated input simulation parameters such as flow rates, vehicle speeds for all active links from the STM.
3. Execute the simulation for a short, fixed interval of time, generating traffic on incoming links of the simulated region based on the input simulation parameters collected in the previous step.

4. Repeat steps 1-3 until the simulation end time is reached.
5. Log results and stop the simulation.

4 TECHNIQUES FOR RAPID INITIALIZATION OF ACTIVE WINDOWS

The simulation engine is instantiated with a road topology map for the entire network. The road topology network includes elements such as signal control plans, routing decisions (i.e. distribution of vehicle paths through intersections), etc. in addition to the network topology. At any instant of time, these values such as the length of the signal cycle or percentages of traffic making a turn at an intersection are known a priori (e.g., derived from historical data) and can be used as inputs to the simulation. A principal question concerns how one should initialize active windows by populating road links with vehicles.

When a simulation window is activated, the links in the window have no vehicles assigned to them. If vehicles are added to the links from at the entry points of the grid an extensive warm up period is required before the new grid sector reaches a steady state. Statistics must be turned off during this period. This is illustrated in Figure 2. A lengthy warm up period reduces the response time of the simulation to report statistics that include the new grid sector. In this experiment a warm period of ten minutes is required for a network wide simulation.

The warm up time can be reduced by initially populating the new window with vehicles along each of its links. The vehicles should be distributed over each link in a realistic manner before the simulation window is activated. However, it is not immediately clear how the links should initially be populated with vehicles.

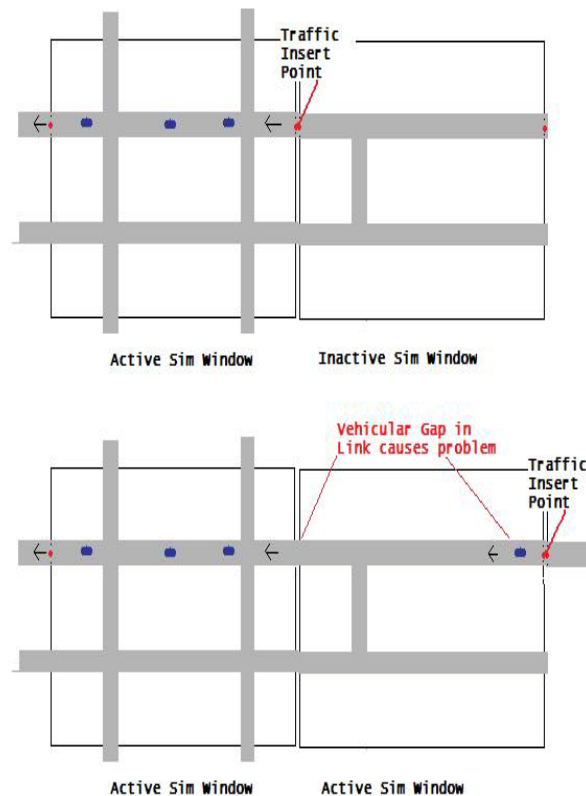


Figure 2: Vehicular gap during instantiation of simulation windows

The following two approaches are investigated to determine the initial distribution of vehicles across the links:

- Vehicle positions are calculated on road links based on input parameters.
- Vehicle positions are derived from results from a previous simulation of the window or direct field data.

The first approach requires a selection of traffic parameters that would yield sufficiently accurate vehicle positions. Several variations of this approach can be defined, depending on how one translates dynamic vehicle information into individual vehicle positions. The second approach is not specific to the vehicular traffic simulation domain. The first approach requires less computation than a pre-simulation and requires less memory as it computes vehicle positions using the input parameters for the simulation. The second approach requires previously computed results for the simulation window or sufficient ve-

hicle instrumentation or in-field sensor data to capture current vehicle positions. Furthermore, where in-field vehicle location data is not directly measurable, the selected previously computed results will depend on the input parameters, thus multiple sets of results must be maintained for different combinations of input parameters.

4.1 Initialization based on vehicular traffic parameters

We consider three different approaches for the initialization of a new active window based on traffic parameters: empty grid initialization (E), simulation with flow rates (FR), and simulation with flow rates and queues (FRQ). These initialization techniques represent trade-offs between the ease of implementation, and computational, communications, and storage overhead and accuracy:

- Empty grid initialization (E): When a simulation window is initialized, the road links within the window are left unpopulated and vehicles enter from the window boundaries and gradually fill the road links in the middle over a period of time. This approach has the advantage of being simple and easy to implement, but clearly incurs a significant warm up time to populate links with vehicles and is inaccurate for short executions of simulation time.
- Simulation with flow rates (FR): The flow rates and vehicle speeds are used to compute the mean vehicular headway and the vehicles are distributed uniformly across the road link based on the mean vehicular gap.
- Simulation with flow rates and queue lengths (FRQ): Queues of slow moving or standing vehicles are created on all road links leading to traffic signals. This technique requires significantly more information than the FR method, including an ability to field measure queue lengths and queued vehicle headways. Currently, mean link queue lengths are used to determine appropriate queue lengths. The remaining road segment is populated using the mean vehicular headway. The headway distance is based on the queue counter configuration. During initial experiments, it was observed that on highly congested links, a small increase in the percentage of vehicles causes large errors in travel times. This can be mitigated by setting the inter vehicle spacing in vehicle queues to the headway configuration in queue counters.

The methods for E and FR are straightforward and implemented as stated above. When populating vehicles in a new active window using FRQ, the vehicular density on a road link, speed of the vehicles on the road link and the distribution of vehicles (i.e. presence of a queue) on the road link must be determined. In the current simulation, we assume that the space time memory provides the mean vehicular speed and flow rate measured in vehicles per unit time on each link. We further assume for FRQ that the space time memory unit additionally provides the queue length on each road link, estimated either through simulations conducted by other vehicles or measured directly in the field. To determine vehicle distribution across a road link, we classify road links into two categories: road links where a queue of closely spaced, slow moving or stationary vehicles exists, and road links where vehicles move at an almost uniform (or desired) speed without stopping.

Using the following parameters, the vehicular density and distribution can be determined for populating the active windows:

On a road link with no queues, a vehicular headway is given by

$$\text{Mean Vehicular Headway} = \text{Mean Vehicular Speed} / \text{Link Flow Rate} \quad (1)$$

The vehicle density on a link is given by

$$\text{Vehicle Density} = \text{Link Length} / \text{Mean Vehicular Headway} \quad (2)$$

However on a road link where a queue exists, vehicles queue up on one portion of the road link, while the remaining portion of the road has vehicles distributed uniformly across it. Under normal operating conditions (i.e. there is not an accident or other cause for a mid-block queue) vehicle queues exist on the exit points of the road link that leads to a traffic signal controller. Queue counters that measure vehicle queue lengths are positioned near traffic signals on links leading to a traffic signal.

We use the following procedure for distributing vehicles across a road link in FRQ:

- If the road link has no queues, compute the vehicular headway and insert the vehicles uniformly at fixed intervals equal to the vehicular headway
- If the road link has a vehicle queue, insert stationary vehicles at the exit point based on the queue length at a small, standard inter vehicle distance. The remaining distance between the end of the queue and the entry point of the link is taken as the effective uncongested link length and traffic is evenly distributed as in the uncongested case over this portion of the link.

4.2 Initialization based on existing simulated window or field data

In this approach, the simulator initializes each road link in a new active window by directly using estimated or field measured vehicle positions. The space time memory unit maintains the link position and speed data for each vehicle in an active road link. To estimate these data, it is realized that vehicle distribution and positions on a road link can vary based on link flow

rate, mean vehicle speed and signal timings used in the simulation. Thus, estimating the vehicle speed-position from a simulated historical database could potentially give rise to a large number of scenarios in order to simulate all possible states. Also, a method must be developed to select the appropriate saved state for the current initialization.

In an in-vehicle simulator belonging to an ad hoc distributed simulation system, initialization information may be obtained by using real vehicle positions (assuming a 100% penetration of instrumented vehicles) or by using previously computed results from other in-vehicle simulators simulating the same region, eliminating the need for a large database of possible system states. In either case, the simulation state set would be limited. In our implementation, we use vehicle positions obtained from a previously simulated network wide simulation which uses the same link flow rate, mean vehicular speed and signal timings as input. This is an idealized scenario equivalent to the assumption that all vehicles contain some minimal level of instrumentation and that it is possible to obtain a real time “snapshot” of all vehicle locations along the area of interest.

5 EXPERIMENTS

5.1 Methodology

Using the design described in Figure 1, we implemented an in-vehicle simulator. The simulation engine was implemented using VISSIM 5.1 (VISSIM 2005) by PTV Vision. VISSIM is a microscopic, multi-modal, time stepped simulation. It uses a driver behavior model based on the physical and psychological conditions experienced by a driver. It allows construction of road networks that allow configuration of multiple lanes, signal timings, traffic distribution, parking lots and transit stops. For evaluation purposes it allows measurement of characteristics such as delay times, travel times, flow rate, queue lengths and vehicle volumes. Furthermore, it provides APIs that allow easy integration with a window based simulator. The simulation engine component contains implementations of the initialization techniques described in the previous section.

In the following experiments we compare results using the windowing technique for different initialization approaches. We use an equivalent simulation of the entire transportation network using VISSIM, (without windowing) to define the base case (ground truth). VISSIM supports travel time segments that measure the time taken for a vehicle to travel from one point in the road network to another. We use travel times measured across different paths in the network as an output metric to verify the simulation. The length of travel times paths are largely restricted to the window size, because a path that spans multiple simulation windows may not be simulated in its entirety.

In these experiments the in-vehicle simulator models two geographically neighboring simulation windows at a time. The window in which the vehicle currently resides and the window that the vehicle will enter next are simulated simultaneously. The in-vehicle simulator uses the flow rates and vehicle speeds obtained from the ground truth simulation as input, representing sensor data that would be stored in the STM. Queue lengths are used only while instantiating new simulation windows. The travel times obtained from the simulation active window approaches outlined in section 4 are compared with the travel times from the ground truth. Vehicles when added during instantiation are set to the mean desired speed of the ground truth simulation. The current simulations were carried out for a single class of vehicles which have a common mean desired speed of 50 km/h.

We imagine that the vehicle carrying the simulator itself is moving across the road network and simulating the local region. In this experiment, the simulator activates at most two geographically neighboring simulation windows at a time. The windows are located in the path of the intended direction of the vehicle. The time taken for a vehicle to move across the road network is measured by aggregating the travel time segments that lie along its path. Since the simulator executes at a much faster pace than the actual time taken for traversal, the simulation is able reach steady state prior to obtaining travel time estimates. The four initialization schemes discussed in the previous section are implemented.

The simulation is configured as follows:

- The road network consists of a square 6X6 rectangular grid, approximately 3 Km in each dimension. It is divided into four simulation window regions (see Figure 3). An in-vehicle simulator travels from the South West quadrant to the North East quadrant.
- The simulation lasts for 1200 seconds (20 minutes) of simulation time. The South West quadrant is activated at the start of the simulation, and the South East quadrant is activated 300 seconds into the simulation. The South West quadrant is deactivated after 600 seconds. The North East quadrant is activated after 600 seconds. The North West Quadrant is not simulated.
- Vehicular traffic enters the road network from the road links bordering the four edges. We use a single vehicle class in this simulation. The flow rates, mean vehicle speeds and queue lengths are aggregated across lanes and collected

for each road link from the ground truth simulation. These values are used to inject traffic into the boundary road links of a simulation window. The traffic signals have a 120 second cycle.

- The queue counter is set to measure queues of vehicles that are moving between 0-5 Km/hr, with a headway or inter vehicle distance less than 10m.

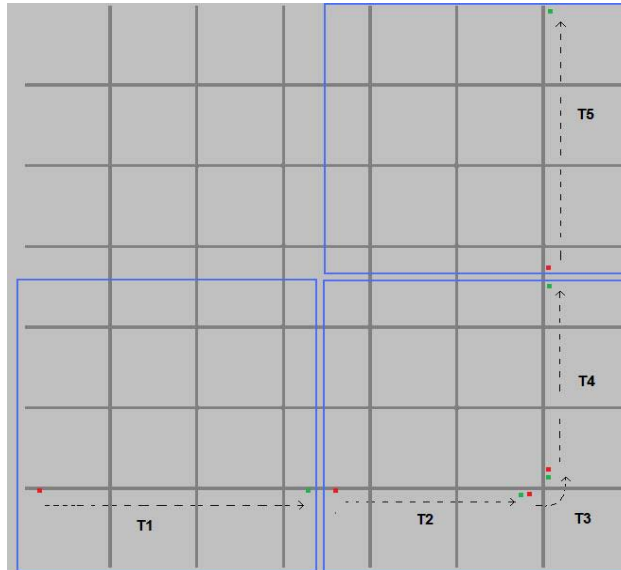


Figure 3: Navigation with window instantiation in a 6X6 road network. The blue grids represent the window boundaries of the road network. The black dotted line indicates the direction of movement of the in-vehicle simulator. A travel time segment consists of a road segment between a consecutive red and green spot. The travel time segments T1-T5 measure the time taken for the vehicle to move from the South West to the North East corner of the grid. (For the complete experiment a total of 16 different travel time segments were utilized; however, only the first five are shown to allow for clarity within the figure.)

The simulator produces quantitative values for the following metrics:

- Travel Times across each of the sixteen different segments. The travel times are computed by averaging the individual travel times of vehicles across the segments over a fixed time interval (240 second intervals).
- Simulation results from the ground truth are collected periodically at short intervals (10 seconds). The mean vehicle volume and vehicle speed are averaged over the duration of the simulation. Similarly, the vehicle queue length counters are averaged over 10 second intervals.

5.2 Comparison of Initialization Techniques

An initial experiment was conducted with an input flow rate of 500 vehicles/hour along all the border road links. The comparison of mean error percentages from the different initialization techniques for each travel time segment is shown in Figure 4. The mean error percentage is computed for each travel time segment using (3). $GTTravelTime_t$ is the ground truth travel time and $SimTravelTime_t$ is the simulated travel time; both of which are measured at the simulation time of t seconds.

$$MeanErrorPercentage = \sum_{t=StartTime}^{EndTime} ((GTTravelTime_t - SimTravelTime_t) / GTTravelTime_t) / (EndTime - StartTime) \quad (3)$$

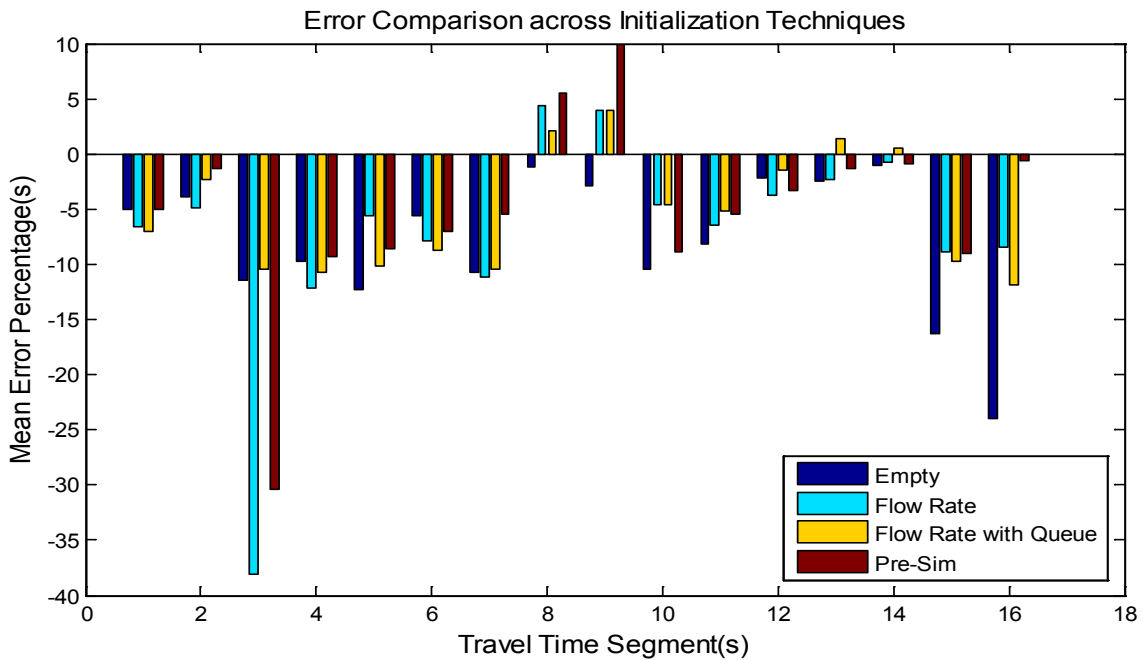


Figure 4: Comparison of mean error percentages across 16 travel time segments. The flow rate across each input link into the 6x6 topology is 500 vehicles/hr.

The initial results are mixed, with no one initialization technique consistently providing the best or worst performance. All methods perform reasonably well over most segments with mean error percentages typically under ten percent. Intuitively, it would seem that FRQ and Pre-Sim should offer the best performance. However, it is likely that any difference between the methods is being overwhelmed by the variability due to the signal control. This begins to indicate that in the transportation arterial environment, one must account for traffic control synchronization as well as vehicle traffic at model initialization, and potentially throughout the run. It is noted that travel time segment 3 has a high error percentage under all four initialization techniques. This segment is an extremely short segment, that cuts across a single intersection. The high rate of variability is due to the short nature of the path relative to the variability introduced by the signal control.

Figure 5 shows the travel times generated by different initialization techniques in segments 1 and 2. The Ground Truth travel time is measured during the entire simulation while the other travel times are measured only when the particular simulation window is activated and thus are not seen across an entire simulation window. We observe that in travel time segments 1 and 2, and in other segments, that the effects of signalization can be clearly identified.

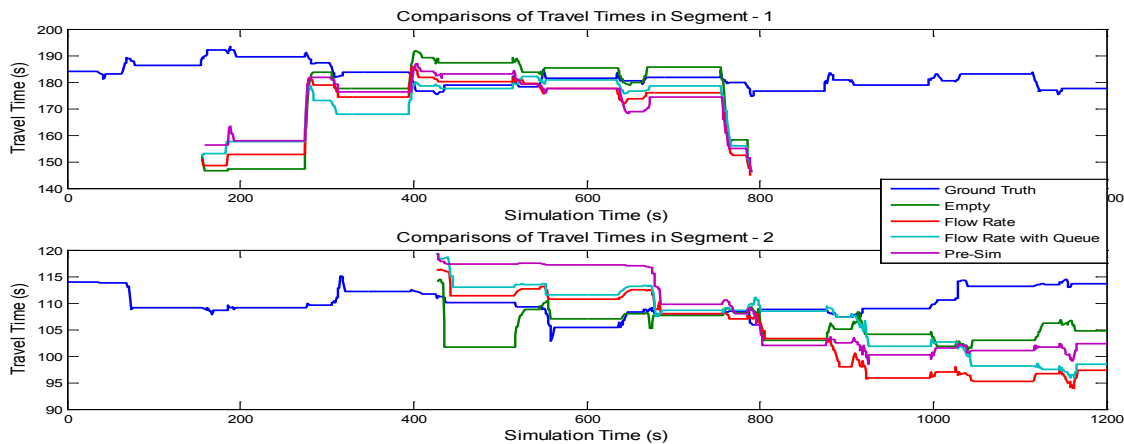


Figure 5: Comparison of Travel Times between Ground Truth and each simulation initialization technique. The flow rate across each input link into the 6x6 topology is 500 vehicles/hr.

It is noted in Figure 5, segment 1, that all of the techniques initially estimate travel times significantly lower than the Ground Truth. Each method then provides travel time estimates within five to ten seconds of the Ground Truth. This phenomenon is a result of the cyclic nature of arterial signal control. It is not until a complete set of signal cycles is experienced that the travel time estimate incorporates vehicles covering the full range of delays at the intersection. A similar phenomenon is seen at the end of the segment 1 prediction time period (i.e. slightly under 800 seconds simulation time).

Segment 2 in Figure 5 demonstrates a different behavior. In this instance the initial portion of the cycle experienced by the vehicles at the start of the simulation results in a travel time closer to the average value, thus, the initialization bias found is not as severe as that found on segment 1. However, it is seen that after approximately 800 seconds the segment 2 predictions for all initialization techniques begins to degrade, with every technique providing ongoing estimates lower than the ground truth. To understand this behavior it is important to note that segment 2 is in the second simulation initiated. The first simulation initiated (which includes segment 1) is ended at approximately 600 seconds. Until time 600 seconds the arrival pattern on segment 2 is that of platooned vehicles departing segment 1. When the segment 1 simulation is ended, the arrival pattern entering segment 2 must be estimated by the segment 2 simulator, as the first link of segment 2 is a boundary link. A uniform flow at the mean arrival rate is utilized for this arrival estimate. The impact of this change in arrival pattern is distinct, witnessed at approximately 800 seconds. It is noted that the effect is not immediate at 600 seconds, as time is required for the arriving vehicles to progress through segment 2 and begin to influence the estimated travel time.

5.3 Test with varying levels of congestions

We repeat the experiment for varying levels of congestion by subjecting the incoming road network links to flow rates of 200, 500 and 1000 vehicles/hr. Figure 6 shows the mean error percentages seen in all the cases.

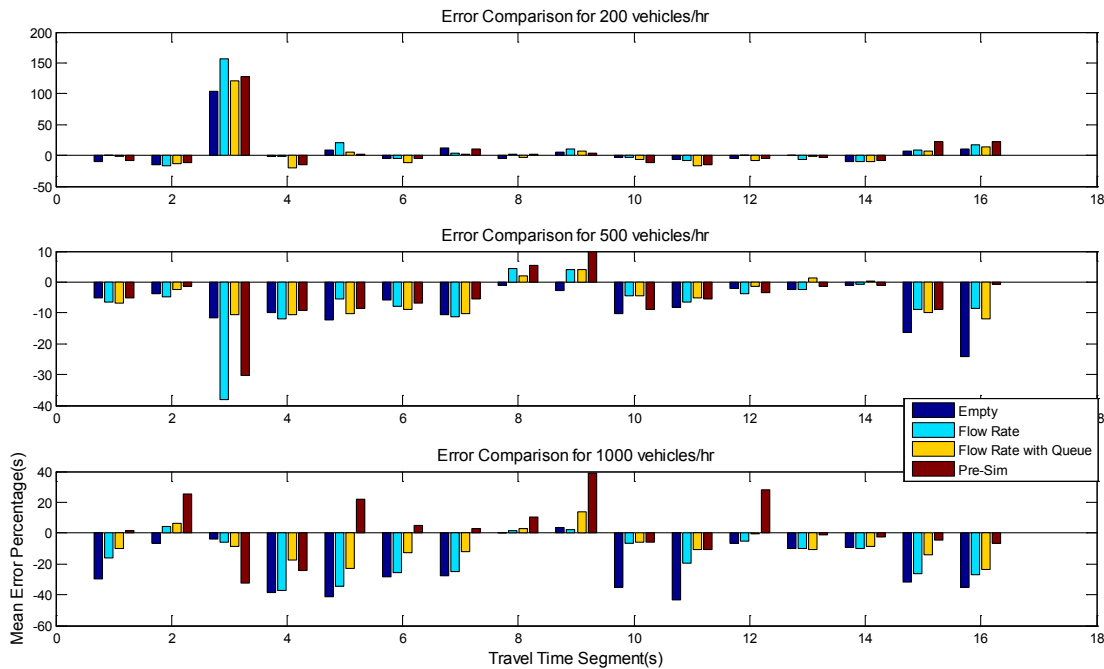


Figure 6: Comparison of mean error percentages with varying levels of congestion. The flow rate across each input link into the 6x6 topology is 200, 800 and 1000 vehicles/hr.

The graphs in Figure 6 indicate similar performance among the initialization techniques at the 200 vehicles/hr and 500 vehicles/hr levels. However, the error of the empty initialization technique increases as the traffic level is increased to 1000 vehicles/hour. With an empty grid, vehicles initially travel unencumbered to their destination, recording lower travel times.

Once a sufficient warm up period has been achieved, however, the network is well populated with vehicles and travel times approaching that of the steady state situation. This suggests that as congestion increases, the empty initialization approach yields larger errors than the others, with the drawbacks of the method beginning to outweigh the signalization effects. In addition, under the higher demands, the FR method also seems to result in higher errors than the FRQ and Pre-Sim techniques. Under the higher demand conditions an initialization state that more closely reflects real world conditions appears to begin to show definite advantages.

6 CONCLUSION

This paper has examined the challenge of rapidly initializing simulation windows by populating links with vehicles on under varying levels of congestion. The results of this experiment suggest that the explored techniques provide similar performance under lower demand levels, with the Empty Initialization and Flow Rate (FR) techniques degrading under higher demands. It is also noted that in the transportation arterial environment, in addition to the vehicle network loading characteristics, synchronization of intersection traffic control parameters with the real time signalization may be critical to the performance of the initialization strategy.

Areas of future research include experiments to determine the optimum amount of simulation time require to allow travel time readings to converge. Also, future efforts will concentrate on experimentation with different arrival patterns on the boundary links. Finally, future efforts will explore the impact of the demand level (i.e. low, moderate, congested) on the reliability of the initialization techniques, potentially resulting in technique selection guidance based on the real world demand.

ACKNOWLEDGMENTS

The research described in this paper was supported under NSF grants EFRI-0735991 and CNS-0540160.

REFERENCES

- Balakrishna, R., H. N. Koutsopoulos, M. Ben-Akiva, B.M. Fernandez Ruiz and M. Mehta. 2005. Simulation-Based Evaluation of Advanced Traveler Information Systems. In TRB 84th Annual Meeting.
- Fellendorf, M. and Vortisch, P. 2001. Validation of the microscopic traffic flow model VISSIM in different real-world situations. Transportation Research Board
- Fisher, K.M. 2000. *Transims is Coming!*. Public Roads, 63(5): 49-51.
- Fishman, G.S. 2001. *Discrete Event Simulation*. Springer Verlag New York.
- Foley, J.D., A. van Dam, S. K. Feiner, and J.F. Hughes. 1990. *Computer Graphics*, Addison-Wesley
- Fujimoto R., and K. Ghosh. 1991. *Proceedings of the International Conference on Parallel Processing*.
- Fujimoto, R. 2000. *Parallel and Distributed Simulation Systems*. John Wiley & Sons, New York.
- Fujimoto R., M. Hunter, J. Sirichoke, M. Palekar, H. Kim, and W. Suh. 2007. Adhoc Distributed Simulations. *Proceedings of the 21st Workshop on Principles of Advanced and Distributed Simulation*:15-24.
- ITT Systems & Sciences Corporation, 1998. *CORSIM User's Manual, Version 1.04.*, Federal Highway Administration, U.S. Department of Transportation.
- Kiesling, T. and J. Lüthi. 2005. Towards Time-Parallel Road Traffic Simulation. *Proceedings of the 19th Workshop on Principles of Advanced and Distributed Simulation*. 7-15.
- Law, A. M., and W. D. Kelton. 2000. *Simulation modeling & analysis*. 3rd ed. New York: McGraw-Hill, Inc.
- Nagel, K., and M. Schreckenberg. 1992. A Cellular Automaton Model for Freeway Traffic. *Journal de Physique*, 2:2221-2229.
- Nagel, K., and M. Rickert. 2001. Parallel implementation of the TRANSIMS micro-simulation. *Parallel Computing*.
- PTV. 2005. *VISSIM User Manual 4.10*. PTV Planung Transport Verkehr AG: Karlsruhe, Germany.
- VISSIM [accessed 11th April 2009] <<http://www.ptvamerica.com/vissim.html>>
- Wiedemann R. 1991. Modeling of RTI-Elements on multi-lane roads, *Advanced Telematics in Road Transport*, Brussels.

AUTHOR BIOGRAPHIES

VISHWANATH PALAGUMMI is a graduate student pursuing his Master of Science degree in the College of Computing at Georgia Institute of Technology. His areas of research include computer networking and simulations. His email is <vpalagum@cc.gatech.edu>

RICHARD FUJIMOTO is a Professor and Chair of the Computational Science and Engineering Division of the College of Computing at the Georgia Institute of Technology. He received the Ph.D. and M.S. degrees from the University of California (Berkeley) in 1980 and 1983 (Computer Science and Electrical Engineering) and B.S. degrees from the University of Illinois (Urbana) in 1977 and 1978 (Computer Science and Computer Engineering). His research is concerned with the execution of discrete-event simulation programs on parallel and distributed computing platforms. This research has included work on platforms ranging from mobile distributed computing systems to cluster computers to supercomputers. This work has included several application areas including transportation systems, telecommunication networks, multiprocessor systems, and defense systems. He lead the working group that was responsible for defining the time management services for the Department of Defense High Level Architecture (HLA) effort (IEEE Standard 1516). His email is [<fujimoto@cc.gatech.edu>](mailto:fujimoto@cc.gatech.edu)

MICHAEL HUNTER is an Assistant Professor at the School of Civil and Environmental Engineering at Georgia Institute of Technology. His primary teaching and research interests are in transportation operations and design, specializing in adaptive signal control, traffic simulation, freeway geometric design, and arterial corridor operations. Dr. Hunter obtained his B.S. in Civil Engineering from Rensselaer Polytechnic University (1992), his M.S. in Civil Engineering from the University of Texas at Austin (1994), and his Ph.D. in Civil Engineering from the University of Texas at Austin (2003). After obtaining his M.S. he worked as a transportation engineer for several years at the Sear-Brown Group in Rochester, NY. He has conducted numerous traffic impact studies, signal timing projects, freeway operation evaluations, toll plaza analyses, etc. His email is [<michael.hunter@ce.gatech.edu>](mailto:michael.hunter@ce.gatech.edu)