

## MULTI-MODEL TRAFFIC MICROSIMULATIONS

Rutger Claes  
Tom Holvoet

DistriNet Labs  
Celestijnenlaan 200A  
Katholieke Universiteit Leuven  
Leuven, B-3001, BELGIUM

### ABSTRACT

Microscopic simulation of traffic, while often necessary to capture the effects of interest, is a computationally expensive simulation strategy. What can be observed, however, is that the accuracy required from the simulation for post simulation analysis can depend on the simulated world and vary over simulated time - roads becoming crowded may be simulated differently than sparsely used roads. In this paper we explore multi-model simulation as an adaptive simulation strategy. Multi-model simulations are capable of selecting and switching to a suitable simulation model at runtime, based on the state of the simulated world. This simulation strategy reduces the computational complexity of traffic microsimulations while still maintaining the desired level of accuracy needed to produce meaningful results. We illustrate the approach via an experimental set-up that allows switching between a road queue model and a fully detailed road simulation model.

### 1 INTRODUCTION

In the context of a multidisciplinary project called MASE, we aim at evaluating the effects of traffic policy measures such as road pricing schemes and car pool incentives on road safety, public health and environment. Microscopic simulations are used for two reasons. First, we incorporate activity based models in the simulations. Activity based models describe individual driver behavior based on their current activity ([Ben-Akivai, Bowman, and Gopinath 1996](#)). Second, the analysis models for environment and safety require detailed rather than aggregated data about the traffic state.

Unfortunately, the accuracy of microscopic simulations comes at a cost. Microscopic simulation of traffic is computationally expensive since every vehicle is represented and simulated individually. Aiming at evaluating effects of policy measures in large traffic networks (large urban regions), it is clear that the management of computational complexity is a key challenge in this project.

Many approaches are being investigated for reducing the computational cost of traffic microsimulations, including parallel and distributed simulations ([Charypar, Axhausen, and Nagel 2007](#), [Cameron and Duncan 1996](#), [Mabry and Gaudiot 1994](#)). Virtually unexplored, however, is runtime adaptation of simulation models - which we call multi-model simulations henceforth. Multi-model simulations allow switching to a more accurate or more performant simulation model for part of the simulation, during simulation.

Microscopic simulation models can differ in their capability of modelling complex vehicle-to-vehicle interactions. A switch to a less accurate and more performant simulation model can e.g. be justified when the current simulation state indicates that the switch will not affect the simulation analysis significantly. The level of detail necessary in the simulation is determined by the requirements of the analysis models that interpret the resulting data. E.g. for air pollution, and more specifically for the exposure of people to vehicle emissions, analysis models such as ([Künzli et al. 2000](#), [Hoek et al. 2002](#)) need detailed data on vehicle emissions including the vehicles positions and acceleration profiles. If, during a traffic microsimulation, a road is only used very sparsely the timing and speed of the vehicles on that road will depend very little on vehicle interactions due to the large inter vehicle distances. The speed of vehicles able to drive freely is easily determined without taking into account vehicle interactions. On a congested road, on the other hand, vehicle interactions become the main factor determining the vehicles speed. The vehicle interactions also become predictable: every vehicle will be affected

by the movement of the vehicle directly in front. Here we can also determine the speed of the vehicles without a very detailed simulation by looking at the overall speed of the traffic jam.

In this paper, we want to explore multi-model simulations. We first discuss two traffic simulation models, each with its own performance and accuracy characteristics. Secondly, we investigate multi-model simulations using both simulation models. We describe a simulation protocol that allows both simulation models to coexist during a single simulation, where each road is simulated by one of the simulation models. Then we describe a strategy that allows online switching of simulation models for roads. In section 4, we present experimental results with multi-model simulations to evaluate the approach. Finally, this evaluation is followed by some concluding remarks.

## **2 SIMULATION STRATEGIES FOR SIMULATING TRAFFIC**

In this section we describe two simulation models. Both simulation models are able to simulate vehicles travelling down a road. The assumptions that both models make differ, causing a difference in both the performance and accuracy of the road simulation. Further in this paper, we describe how we exploit these differences in order to gain performance. This section is by no means meant to highlight the best, let alone all usable simulation models. Instead, we put forward two simulation models with clearly distinguished characteristics. This allows us to explore multi-model simulation further on.

### **2.1 Queue Based Model**

A first, performant simulation model treats roads as queues. This simulation model is based on work done in (Gawron 1998). The authors of (Balmer et al. 2008) and (Simon and Nagel 1998) continued work on this queue based model and provided the basic algorithm on which this simulation model is based.

In (Gawron 1998, Balmer et al. 2008, Simon and Nagel 1998) the simulation is part of an iterative approach to find the dynamic user equilibrium in traffic assignment. The goal of the queue based model is to simulate travel times, not travel behavior. In general, the queue based model would lack the accuracy needed for the analysis models. This is due to the assumptions the queue based model makes in the trade-off between accuracy and performance.

The implementation of a queue based simulation model is straightforward. Whenever a vehicle enters a queue, the simulation calculates the earliest time the vehicle is able to leave the road. The simulator determines the average speed for the vehicle based on the traffic density on the road. Using this average speed together with the length of the road, the simulation can determine the earliest departure time for the vehicle. If this minimal departure time is later than the departure time of the vehicle in front of the arriving vehicle, it becomes the departure time for the vehicle. If, however, the minimal departure time is earlier than that of the vehicle in front, the arriving vehicle will be scheduled to leave some time after the departure of the vehicle in front.

Under certain conditions, the errors caused by the assumptions made in the queue based model no longer contribute to the overall results. When traffic density is sufficiently low to allow free flow traffic, the assumption that a vehicle will not diverge from its average speed becomes plausible. At this point the queue based model not only offers a good prediction for the travel time of a vehicle, it also provides a good description of the vehicles behavior on the road.

### **2.2 Continuous Vehicle Centric Model**

The second road simulation model is a more accurate simulation model that models road lanes as one dimensional continuous spaces. Every vehicle has a position in this one dimensional space similar to the simulation model described in (Paruchuri, Pullalarevu, and Karlapalem 2002). In order to simulate a road, this simulation model queries all vehicles of the road starting with the one that is furthest on the road. The simulation will ask the vehicle how far it intends to drive given speed limits on the road and the position of the vehicle in front. This simulation model thus hands over control to the vehicle. Vehicles are never allowed to drive further than the vehicle in front of them, nor are they allowed to exceed the speed limits. At regular time intervals, vehicles are allowed to switch road lanes in order to overtake the vehicles in front of them.

This model allows the modelling of car following behaviour, start-and-stop waves in congested traffic and other traffic phenomenon the queue based model is unable to model. This accuracy comes at a cost. Where the queue based model only needs to look at a single vehicle in each iteration, this model needs to evaluate every vehicle on every road in each iteration.

### 2.3 Difference between both Simulation Models

Both simulation models were evaluated using the road network depicted in Figure 1. Vehicles are injected into the network at points A and B. The two traffic streams meet at C, which forms a bottle-neck. Junctions D and E further disturb the traffic flow by limiting the rate vehicles can pass.

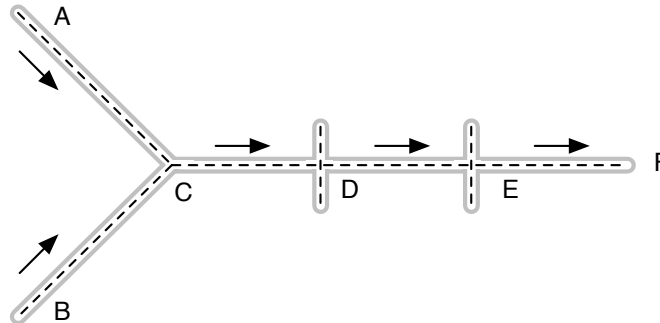


Figure 1: Road network for the simulations. Vehicles enter the network at points A and B. They drive across C, D and E to F where they leave the network.

One of the important metrics of the simulated traffic situation is the traffic density on the roads. Traffic density is not a parameter of the simulation, it is the result of different parameters such as road length, road speed and the pattern at which vehicles are injected into the network. This last parameter, the vehicle injection profile, is easily altered between different runs of the simulation and strongly influences the traffic densities of the simulated roads. Therefore it will be used to explore the simulation outcomes for different traffic densities.

To compare the outcome of different simulation runs we will focus on the average speed of individual vehicles. The average speed of vehicles is easily determined for both simulation models and is a good representation of the vehicles behavior during the simulation.

Figure 2 demonstrates the difference in simulation outcome between both models when the roads are getting crowded. The graphs represent the vehicle density according to their average speed. The top row shows the distribution for high traffic densities. The vehicle density of both the queue and the detailed continuous model are very similar. Both have a peak around  $25\text{ km/h}$ , indicating the majority of vehicles reached an average speed of around  $25\text{ km/h}$ . The long tail on the right represents vehicles that managed to stay ahead of the congestion.

The bottom row of Figure 2 also features two similar vehicle distributions. The vehicle distributions on the bottom represent simulation outcomes of tranquil traffic situations for both simulation models. For these tranquil traffic situations the simulation outcomes are similar. Both feature a strong peak around  $83\text{ km/h}$ , meaning almost all vehicles were able to reach the speed limit of  $90\text{ km/h}$  and were only delayed by the junctions.

The middle row of Figure 2 shows the vehicle distribution for a medium traffic density. Here both vehicle distributions have different shapes. The right vehicle distribution, the one based on the detailed continuous model, resembles the crowded distribution just above. The left vehicle distribution, the one based on the queue model, resembles the tranquil distribution just below. Vehicles in the detailed continuous model have lower average speeds and already appear to be congested at this traffic density while the vehicles in the queue based model still reach their free flow speeds. This observation is confirmed in Figure 3. The delays introduced by vehicle interactions occurring at this traffic density slow the vehicles down in the detailed model, but not in the queue model. In conclusion, we can say that both models produce similar simulation outcomes for crowded and sparsely used roads, but produce significantly different outcomes for roads where congestion is starting to build.

## 3 COMBINING DIFFERENT SIMULATION STRATEGIES

Runtime adaptation of simulation models requires several simulation models to coexist within one microsimulation. We describe a protocol that is able to connect roads using different simulation models, which we call the schedule-pull protocol. This protocol allows considering an adaptive strategy, in which a suitable simulation model can be selected dynamically

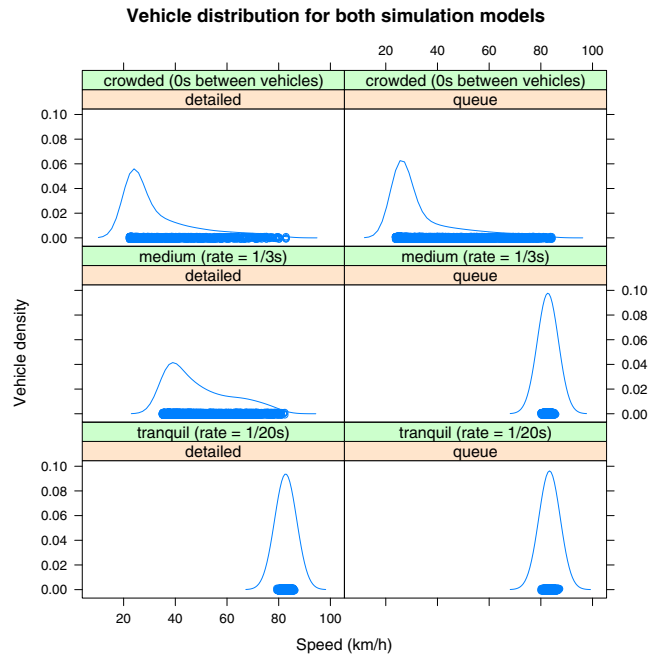


Figure 2: Distribution of the vehicles speed for various levels of vehicle injection rates and simulation models

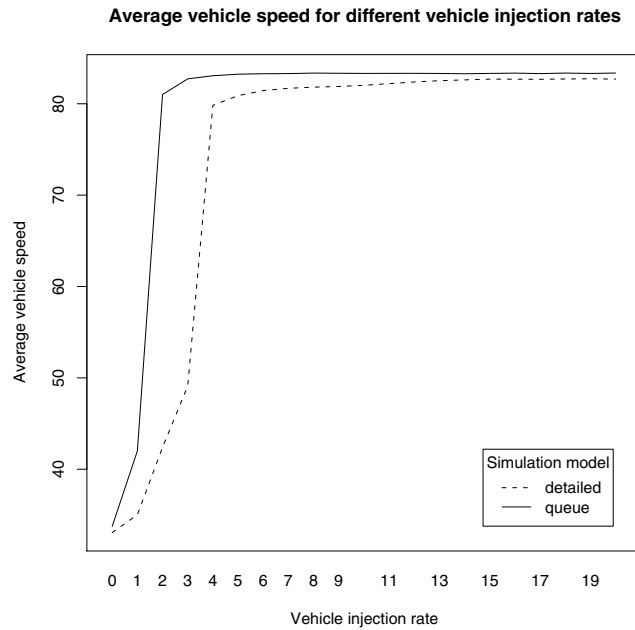


Figure 3: Average vehicle speed in simulation outcome for various levels of vehicle injection rates and both simulation models

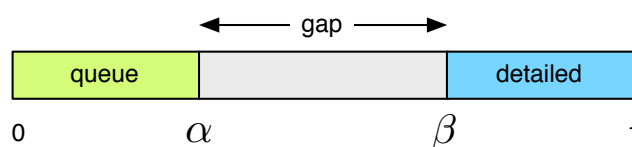


Figure 4: Whenever the road density drops below  $\alpha$  into the green range, the road will be modelled using the queue based simulation model. If the road density rises above  $\beta$  into the blue range, the detailed model will be used. As long as the density stays between  $\alpha$  and  $\beta$  the current simulation model is kept.

for individual roads. This section describes how to switch between different simulation models at runtime and describes a selection strategy to select the model to use.

### 3.1 Schedule-Pull Protocol

For both simulation models described above it is possible to determine whether a vehicle has travelled across the entire road. For the queue based model a vehicle is ready to leave whenever its departure time has passed and it is the last vehicle in the queue. For the detailed model, a vehicle is ready to leave when its position equals the end of the road. In both cases, vehicles able to leave are *scheduled* on the next road. While a vehicle is scheduled on the next road, it remains occupying space at the end of the current road preventing vehicles behind it to advance and possibly new vehicles to enter. Whenever the next road has enough space to accommodate the scheduled vehicle, it will *pull* the vehicle off the previous road. This frees the space occupied by the vehicle on the previous road and reduces the available space on the next road.

This *schedule-pull* protocol allows the coexistence of both models. By not removing a scheduled vehicle from its road, the protocol can account for spill-overs due to congestion on the next road.

### 3.2 Runtime Switching between Simulation Strategies

The runtime switching between both models is governed by runtime simulation metrics. These metrics are monitored throughout the simulation. Early evaluations of both simulation models revealed that their outcome differ significantly when traffic density starts to increase. For very high and very low traffic densities, both models produce the same outcome. Based on this observation the traffic density of a road was chosen as the metric to trigger a change in the simulation model used. The density of a road is defined as the space currently occupied by vehicles on the road divided by the total length of that road. As we make no assumption on the validity of either of the models, a more sophisticated set of metrics is not required.

Initializing the queue model according to the state of a more detailed continuous model and vice-versa is straightforward. The position of every vehicle on the road is calculated and used to initiate the vehicle in the new model. When using the queue model to initialize the detailed model, the position of the vehicle is determined by looking at the time it spend on the road and the time it still needs to complete. When using the detailed model to initialize the queue model the queue simulation model calculates the departure time of the vehicles based on their current position. Using these naive transformation rules introduces transient inconsistencies in the vehicles position. These inconsistencies are inherent to the queue model as it does not model the position of the vehicle. The inconsistencies are not a problem as none of the analysis models can depend on the queue model for vehicle positions.

### 3.3 Selecting a Suitable Simulation Strategy

As described in the previous section, the runtime switching between both models is controlled using the traffic density of a road. The rules for switching between both models is depicted in Figure 4. We define two levels,  $\alpha$  and  $\beta$ , to guide the switching process. Whenever the road density falls below  $\alpha$  the road will be mapped onto the queue simulator. If the road density rises above  $\beta$  the road will be simulated using the detailed simulation model. Whenever the road density is in the  $]\alpha, \beta[$  interval, the current simulation model is preserved. The gap between  $\alpha$  and  $\beta$  prevents thrashing between both simulation models.

An  $\alpha$  value of 0.2 combined with a  $\beta$  value of 0.3 means the simulation will use the queue model to simulate roads with a traffic density below 20% and a more detailed, continuous model to simulate roads with a traffic density higher than 30%. The 10% gap in between will prevent the simulation from thrashing between both models.

#### 4 EVALUATION OF RUNTIME ADAPTATIONS

To evaluate the proposed approach we will start by examining the influence of runtime adaptations to the simulation outcome. Then we will continue by examining the influence of runtime adaptations on the performance of the simulation. Finally, we will combine both influences and determine a suitable adaptation strategy that improves the performance of the simulation without sacrificing accuracy.

All experiments described in this evaluation are based on the traffic network depicted in Figure 1. All simulations were done on Intel® Pentium® D 3.0Ghz processors with 1GB of ram and the OpenJDK 1.6 JVM. During the simulation, every road depicted in Figure 1 is modelled using either the detailed continuous model or the queue based model. The model can be changed during the simulation on a road per road basis based on the road density and the levels for  $\alpha$  and  $\beta$ .

##### 4.1 Influence of Runtime Adaptations on Simulation Outcome

The first experiment examines the influence of runtime adaptations on the simulation outcome. By fluctuating the vehicle injection rate we introduce evolving traffic densities on the simulated roads. We keep  $\alpha = \beta$ , to limit the parameter space. The performance penalty caused by the thrashing between both simulation models is of no importance to this experiment as we only examine the influence on the simulation outcome and not on simulation performance. The influence on simulation performance is examined in Section 4.2

Figure 5 shows the vehicle distribution according to their average speed for various values of  $\alpha$  and  $\beta$ . The distribution for  $\alpha = \beta = .4$  is already identical to that of  $\alpha = \beta = 1$ . Therefore, we narrow our parameter exploration to  $\alpha$  and  $\beta$  values around .2. This is depicted in Figure 6. Figure 6 illustrates how the vehicle density gradually shifts from the density corresponding to the queue and the detailed model.

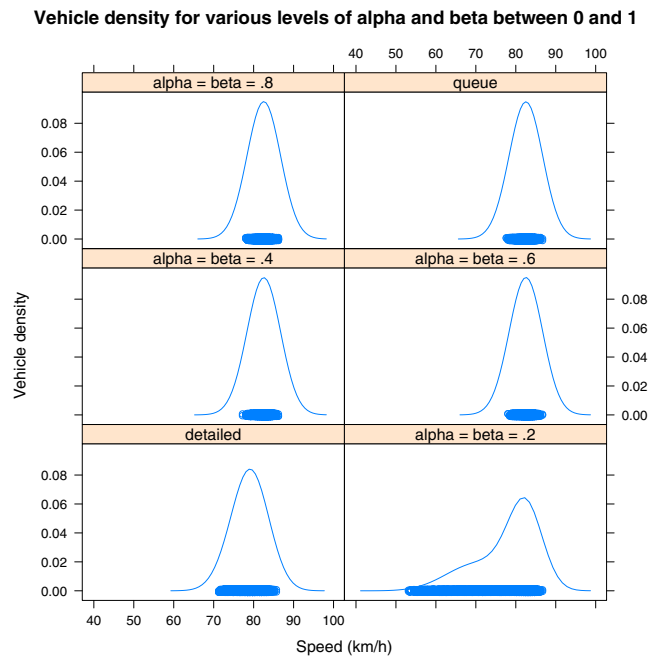


Figure 5: Distribution of the vehicles speed for various levels of  $\alpha$  and  $\beta$  between 0 and 1 with fluctuating vehicle injection rates

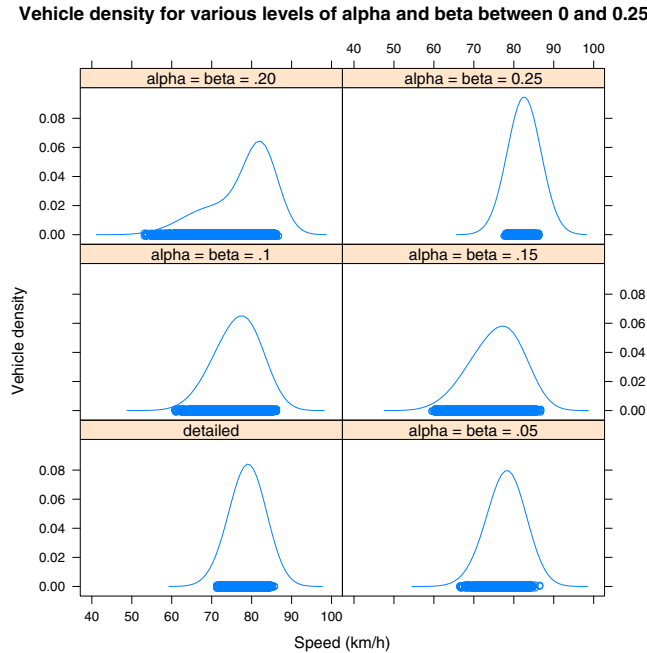


Figure 6: Distribution of the vehicles speed for various levels of  $\alpha$  and  $\beta$  between 0 and .25 with fluctuating vehicle injection rates

#### 4.2 Influence of Runtime Adaptations on Simulation Performance

Analysing the overhead introduced by the runtime evaluations and adaptations of the simulation models is hard. Not only will this overhead depend on the level of  $\alpha$  and  $\beta$ , it will also depend on the gap between them.

Figure 7 illustrates the influence of  $\alpha$  and  $\beta$  on the average cpu time required to run the simulation. The simulation will use the queue model more often as the value of  $\beta$  increases. The graph in figure 7 illustrates how the average cpu time decreases because of this increased usage of the queue model.

As the gap between  $\beta$  and  $\alpha$  increases,  $\alpha$  becomes smaller. Because of the increased gap, the simulation will spend less time switching between both simulation models. This should lead to a decrease of required cpu time. However, according to Figure 7, the exact opposite happens. The the influence of the decrease of  $\alpha$  and the resulting decreased usage of the detailed simulation model is more significant.

This last observation shows that, given a sensible adaptation strategy, the overhead of runtime adaptation is less significant than the speedup the adaptation is able to achieve.

#### 4.3 The Performance versus Accuracy Trade-Off

In this section we examine the trade-off between accuracy and performance. Runtime model changes have the potential of improving simulation performance. However, they do so by switching to a less detailed model, possibly resulting in a less accurate overall simulation outcome. The right balance between accuracy and performance will depend on the accuracy requirements of the post-simulation analysis models.

Combinations of  $\alpha$  and  $\beta$  can be characterised based on their accuracy and performance. Figure 8 plots the accuracy and performance of a number of interesting combinations of  $\alpha$  and  $\beta$  with a gap of 0.075. The vertical axis of Figure 8 shows the performance, while the horizontal axis indicates how accurate the simulation outcome is.

We define the accuracy of a simulation as being:

$$accuracy = 1 - \frac{\sqrt{(\bar{v} - \bar{v}_{sim})^2}}{\bar{v}}$$

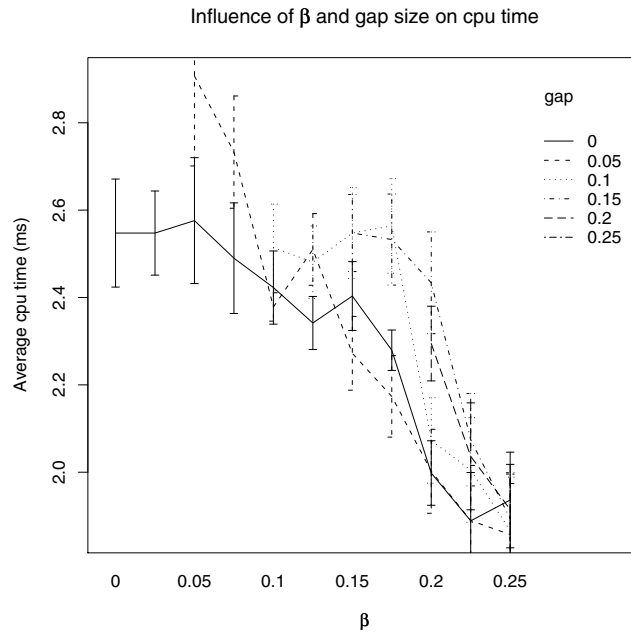


Figure 7: Average cpu time needed to complete the simulation. As  $\beta$  increases, the queue model is used more often, thus the average required cpu time decreases. As the gap becomes wider,  $\alpha$  becomes smaller. The error bars in the figure indicate the 95% confidence intervals.

Here *accuracy* is the accuracy of the average vehicle speed in a simulation,  $\bar{v}$  is the real-world average speed of vehicles and  $\bar{v}_{sim}$  is the average speed of vehicles in the simulation. Expressed in percentages, an accuracy of 100% means vehicles in the simulation have, on average, the same speed as vehicles in the real world. For the rest of this section we will assume the average speed measured using the detailed continuous simulation model is the best representation of the real-world average speed we are able to obtain using our simulation models. The horizontal axis of Figure 8 shows this relative accuracy of the  $\alpha \beta$  combinations.

The cpu time needed to complete the simulation using only the continuous detailed model will serve as a baseline when comparing the performance of the adaptive simulations. In Figure 8 this baseline is indicated by a horizontal line at 2.57ms. The vertical distance between the data point and the performance baseline indicates the performance benefit of using runtime adaptations based on that combination of  $\alpha$  and  $\beta$ . The data points in Figure 8 illustrate that accuracy comes at the cost of reduced performance.

Given a suitable adaptation strategy, switching between both models can result in a performance gain of at least 9.7% in cpu time required to complete the simulations. Figure 8 shows the accuracy and performance of several combinations of  $\alpha$  and  $\beta$  with a gap of 0.075. One combination ( $\alpha = 0.05$ ,  $\beta = 0.125$ ) achieves a performance gain of 9.7% while only being 0.1% less accurate than the detailed continuous model.

## 5 RELATED WORK

There is an extensive body of work around multiresolution simulation (Drewry et al. 2002, Jeschke and Uhrmacher 2008, Natrajan et al. 1997, Reynolds et al. 1997, Tan et al. 2001) and on multiresolution simulation in traffic specifically (Burghout, Koutsopoulos, and Andreasson 2005, Magne, Rabut, and Gabard 2000, Flötteröd and Nagel 2007). The limitations of macroscopic simulation models and the complexity of microscopic models have lead to a wide variety of hybrid approaches.

Multiresolution traffic simulations such as described in (Burghout et al. 2005, Magne et al. 2000, Flötteröd and Nagel 2007) combine microscopic simulations of traffic situations where modelling inter vehicle interactions such as overtaking or car following behavior are crucial to the accuracy of the simulation with computationally more easy macroscopic simulations of freeways. Thus combining accuracy and scalability by modelling at two different resolutions.



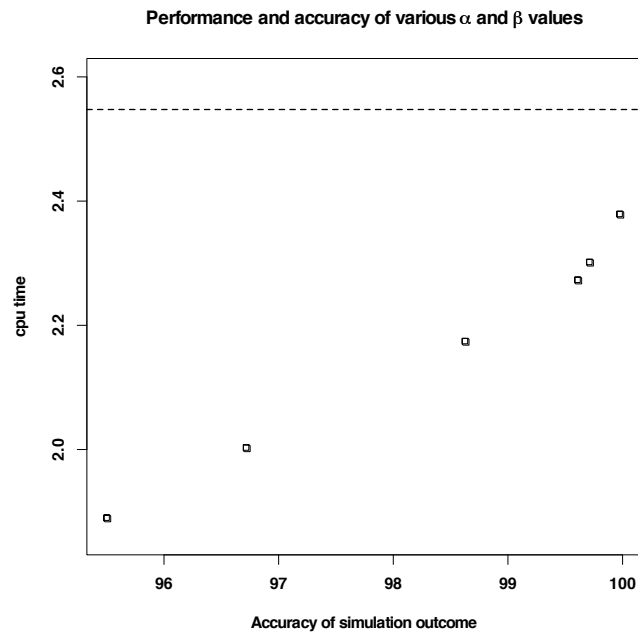


Figure 8: Accuracy and performance of several combinations of  $\alpha$  and  $\beta$  with gap of 0.05. Every data point represents one possible combination of  $\alpha$  and  $\beta$  values. This graph only shows combinations with a gap of 0.05. The horizontal dashed line indicates the cpu time needed to run the simulation using only the detailed continuous model. The top right data point represents  $\alpha = 0.05$  and  $\beta = 0.1$ . The vertical space between the data points and dashed line represents the performance gained by runtime switching. The horizontal space to the left of the data points represents the loss of accuracy due to the adaptive approach.

This approach has also been applied successfully in other domains such as military battle simulations (Tan, Ng, and Moradi 2001) and biological simulations (Jeschke and Uhrmacher 2008, Drewry, Reynolds, and Emanuel 2002).

The requirements of the MASE project do not allow such an approach because macroscopic simulation models are not fit to simulate activity based models. The aggregation of vehicles that occurs when the simulation switches to macroscopic model not only disregards the identity of the vehicle, making it hard to take into account the activity of the vehicle, but also severs the link between the vehicle and the output data.

This paper proposes another solution. Alternating between the two simulation models as described in section 3 does not cross the border of a simulation resolution. The granularity of the model does not change. As a consequence the process of aggregation and disaggregation, as described in for example (Tan, Ng, and Moradi 2001), together with related problems such as cascading effects (Reynolds, Natrajan, and Srinivasan 1997) are avoided here. Instead of aggregation and disaggregation, switching between simulation models at the same level only requires a translation. And even this translation can be avoided if both simulations can operate on the same representation.

In (Reynolds, Natrajan, and Srinivasan 1997) and (Natrajan, Reynolds, and Srinivasan 1997), the authors propose the use of Multiple Resolution Entities, or MRE's, to maintain the consistency between different levels of resolution. In our approach the representation of the vehicles serves a similar purpose: maintaining the consistency between vehicle information in all simulation models. Unlike MRE's which describe an entities status and behavior at multiple resolution levels, our vehicle representations only need to describe vehicles at one resolution level albeit independent of the simulation model.

The use of multiple models to describe one system at the same resolution has been applied in other fields such as chemistry (Alfonsi et al. 2005) where, depending on the state of the reagens, a choice is made between a stochastic or deterministic model. Stochastic chemical models are usually computationally more expensive than deterministic models. Stochastic models, however, are much more fit to describe chemical systems where a reaction depends on whether the necessary reagens encounter each other. If the necessary reagens are omnipresent in the environment a deterministic model can approximate the stochastic model.

Similarly to the stochastic model used in (Alfonsi et al. 2005), the queue based model described in section 2 disregards certain small scale interactions. The assumptions made in (Alfonsi et al. 2005) about the molecular interactions when the presence of all reagens is abundant are equivalent to the assumptions made in this paper on interactions between cars in a congested traffic state: spending computational power on determining whether the interactions occur is not necessary due to their likeliness it is assumed they occur.

## 6 CONCLUDING REMARKS AND FUTURE WORK

In this paper we have explored the applicability of multi-model simulations. Actions of drivers simulated by the micro simulation model are accurate enough to serve as input for the post-simulation analysis models. Intervals modelled by the queue models can be analysed and driver actions can be extrapolated from the simulation results. At no time however is there a need for aggregation. The relationship between simulation outcome and the activity based model that serves as its input stays preserved.

Experiments on the described approach illustrate the possible differences between multiple microscopic simulation models. Congestion occurs more frequently in the continuous detailed model due to a better modelling of the vehicle interactions leading to a decrease in vehicle speed. The experiments also demonstrate the ability of the simulation to alter between different simulation models at runtime. The overhead of these adaptations is shown to be of less significance than the speedup they realize.

It is important to notice that the approach described in this paper does not assume either the detailed or queue based model to be more accurate than the alternative. The results of both models depend on a large number of parameters and for neither of the models these parameters are validated or calibrated using real world data. The detailed continuous model has more inherent potential for accuracy. Its fine grained modelling of vehicle interactions and control over the vehicles driving behavior makes it more expressive. A well calibrated detailed simulation model could be used to calibrate the queue model using an approach similar to that of (Drewry, Reynolds, and Emanuel 2002).

The approach described should by no means be considered to be a generic approach. The assumptions made in this paper depend greatly on the post-simulation analysis models and domain specific characteristics. However, similarities can be drawn between the approach presented in this paper and adaptive or multi-model simulations in other domains (Alfonsi et al. 2005). A commonality between all these approaches is that they exploit specific domain characteristics to reduce computational complexity of the simulation by simplifying the simulated model without sacrificing the accuracy of the simulations outcome.

The decision making process used to select a suitable simulation model described in this paper only uses the current state of the simulated traffic network. Our aim within the MASE project is to use a biologically inspired routing algorithm described in (Weyns, Holvoet, and Helleboogh 2007). Because of the anticipatory nature of this algorithm, it can be used to predict future traffic densities. This would allow the decision making process to anticipate future traffic densities and to decide on simulation models accordingly.

## ACKNOWLEDGMENTS

The authors would like to thank Roland Ewald for his cooperation in the preparation of this paper. This research was funded by the IWT - SBO project 'MASE' (project no. 060823) and by the Interuniversity Attraction Poles Programme Belgian State, Belgian Science Policy, and by the Research Fund K.U.Leuven

## REFERENCES

- Alfonsi, A., E. Cancès, G. Turinici, B. D. Ventura, and W. Huisinga. 2005, sep. Adaptive simulation of hybrid stochastic and deterministic models for biochemical systems. *ESAIM: Proceedings* 14:1–13.
- Balmer, M., K. Meister, M. Rieser, K. Nagel, and K. Axhausen. 2008. Agent-based simulation of travel demand: Structure and computational performance of matsim-t. *VSP Working Paper* 08-07. Presented at "Innovations in Travel Modelling", Portland OR, 2008.
- Ben-Akivai, M., J. Bowman, and D. Gopinath. 1996. Travel demand model system for the information era. *Transportation* 23 (3): 241–266.
- Burghout, W., H. Koutsooulos, and I. Andreasson. 2005. Hybrid mesoscopic-microscopic traffic simulation. *Transportation Research Record* 1934:218–255.

- Cameron, G., and G. Duncan. 1996. PARAMICS—Parallel microscopic simulation of road traffic. *The Journal of Supercomputing* 10 (1): 25–53.
- Charypar, D., K. Axhausen, and K. Nagel. 2007. An event-driven parallel queue-based microsimulation for large scale traffic scenarios. *VSP Working Paper 07-03*. submitted to WCTR 2007.
- Drewry, D., J. Reynolds, P.F., and W. Emanuel. 2002, Dec.. An optimization-based multi-resolution simulation methodology. In *Proceedings of the 2002 Winter Simulation Conference*, ed. E. Yücesan, C.-H. Chen, J. L. Snowdon, and J. M. Charnes, Volume 1, 467–475. Los Alamitos, CA, USA: Institute of Electrical and Electronics Engineers, Inc.
- Flötteröd, G., and K. Nagel. 2007. High speed combined micro/macro simulation of traffic flow,. *VSP Working Paper 07-07*. accepted for presentation at ITSC 2007 ([www.itsc2007.org](http://www.itsc2007.org)).
- Gawron, C. 1998. An iterative algorithm to determine the dynamic user equilibrium in a traffic simulation model. *International Journal of Modern Physics C* 9 (3): 393–408.
- Hoek, G., B. Brunekreef, S. Goldbohm, P. Fischer, and P. van den Brandt. 2002. Association between mortality and indicators of traffic-related air pollution in the Netherlands: a cohort study. *The lancet* 360 (9341): 1203–1209.
- Jeschke, M., and A. M. Uhrmacher. 2008. Multi-resolution spatial simulation for molecular crowding. In *Proceedings of the 2008 Winter Simulation Conference*, ed. S. J. Mason, R. R. Hill, L. Mönch, O. Rose, T. Jefferson, and J. W. Fowler, 1384–1392. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.
- Künzli, N., R. Kaiser, S. Medina, M. Studnicka, O. Chanel, P. Filliger, M. Herry, F. Horak, V. Puybonnieux-Textier, P. Quenel et al. 2000. Public-health impact of outdoor and traffic-related air pollution: a European assessment. *The Lancet* 356 (9232): 795–801.
- Mabry, S. L., and J.-L. Gaudiot. 1994, Dec.. Distributed parallel object-oriented environment for traffic simulation (poets). In *Proceedings of the 1994 Winter Simulation Conference*, ed. J. D. Tew, S. Manivannan, D. A. Sadowski, and A. F. Seila, 1093–1100. San Diego, CA, USA: Society for Computer Simulation International.
- Magne, L., S. Rabut, and J. Gabard. 2000. Towards a hybrid macro-micro traffic flow simulation model. In *Proceedings of INFORMS Spring 2000 Conference, Salt Lake City, USA*.
- Natrajan, A., J. Reynolds, P.F., and S. Srinivasan. 1997, Jun. Mre: a flexible approach to multi-resolution modeling. In *Proceedings of the 1997 Workshop on Parallel and Distributed Simulation*, 156–163.
- Paruchuri, P., A. Pullalarevu, and K. Karlapalem. 2002. Multi agent simulation of unorganized traffic. In *Proceedings of the first international joint conference on Autonomous agents and multiagent systems: part 1*, 176–183. ACM New York, NY, USA.
- Reynolds, Jr., P. F., A. Natrajan, and S. Srinivasan. 1997. Consistency maintenance in multiresolution simulation. *ACM Trans. Model. Comput. Simul.* 7 (3): 368–392.
- Simon, P., and K. Nagel. 1998. Simple queueing model applied to the city of Portland. In *Transportation Research Board (TRB) annual meeting, Washington, DC (United States), Jan 1999*.
- Tan, G., W. Ng, and F. Moradi. 2001. Aggregation/disaggregation in hla multi-resolution distributed simulation. In *IEEE International 2001 Workshop on Distributed Simulation and Real-Time Applications*, 76–83.
- Weyns, D., T. Holvoet, and A. Helleboogh. Sept. 30 2007-Oct. 3 2007. Anticipatory vehicle routing using delegate multi-agent systems. *Intelligent Transportation Systems Conference, 2007. ITSC 2007. IEEE*:87–93.

## AUTHOR BIOGRAPHIES

**RUTGER CLAES** is a PhD student in DistriNet labs at the Department of Computer Science, Katholieke Universiteit Leuven. He graduated as a master in applied sciences and engineering at the Katholieke Universiteit Leuven. His current research interests include Multi-Agent systems, Multi-Agent based simulations and decentralized coordination mechanisms. His email address is <[rutger.claes@cs.kuleuven.be](mailto:rutger.claes@cs.kuleuven.be)>.

**TOM HOLVOET** is a professor in the Department of Computer Science, Katholieke Universiteit Leuven. His research interests include software engineering of decentralized and multi-agent systems, coordination, software architecture, and autonomic computing. He received his PhD in computer science from the Katholieke Universiteit Leuven in 1997. Contact him at <[tom.holvoet@cs.kuleuven.be](mailto:tom.holvoet@cs.kuleuven.be)>.