Proceedings of the 2014 ORSSA Annual Conference
pp. 1–8
www.orssa.org.za/wiki/uploads/Conf/2014ORSSAConferenceProceedings.pdf

ORSSA Proceedings
ISBN
978-1-86822-656-6

Design of a detailed microscopic traffic simulation modelling framework for signalised intersections

AP Burger[*], MD Einhorn[†] & JH van Vuuren[‡]

Abstract

There are numerous advantages to using simulation when investigating the effectiveness of novel traffic control strategies at signalised intersections. If the level of detail required for the investigation is not too demanding, a commercially available traffic simulation model may suffice. If, however, a high level of realism (such as the incorporation of explicit vehicle accelerations and decelerations, vehicle turning parameters and heterogeneous vehicle sizes) is required, it may be necessary to build a purpose-made traffic simulation model satisfying the specific requirements of the investigation. In this paper, a microscopic traffic simulation modelling framework is presented which may be employed as a stand-alone and customizable traffic simulation tool for testing the effectiveness of existing and novel traffic control algorithms, some of which require individual vehicle characteristics, such as vehicle speed and their positions on road segments as input data.

Key words: Microscopic, Simulation, Traffic, Model.

1 Introduction

Numerous strategies have been proposed in recent years for mitigating the debilitating effects of traffic congestion. One such approach, which is especially applicable to inner city commuting, is the attempted optimisation of traffic signal timings at signalised intersections. Improved and efficient signal timings have the ability to reduce driver delay times by effectively utilising intersection capacity and allowing for the formation and propagation of "green waves" (platoons of vehicles travelling unimpeded through several adjacent intersections displaying green signals). This reduces the stop-and-go driving patterns associated with congested traffic which drivers in Los Angeles, Mexico City, India, China, Singapore, and Johannesburg listed as their most serious commuter pain in the IBM 2011 Global Commuter Pain Survey [7].

[*]Department of Logistics, Stellenbosch University, Private Bag X1, Matieland, 7602, South Africa, email: apburger@sun.ac.za

[†]Corresponding author: Department of Logistics, Stellenbosch University, Private Bag X1, Matieland, 7602, South Africa, email: einhorn@sun.ac.za

[‡]**(Fellow of the Operations Research Society of South Africa)**, Department of Industrial Engineering, Stellenbosch University, Private Bag X1, Matieland, 7602, South Africa, email: vuuren@sun.ac.za

Before any novel traffic signal control strategies may, however, be implemented on public roads, their effectiveness and reliability should be tested extensively. Simulation modelling is a powerful tool which may be used in the design, implementation and evaluation of traffic signal control strategies. There are three distinct classes of traffic simulation models, *i.e. macroscopic, mesoscopic* and *microscopic models*. Macroscopic traffic simulation models are typically modelled from an aggregated point of view, based on a hydrodynamic analogy and regard traffic flows as a particular fluid process whose state is characterised by aggregate macroscopic variables such as density, volume and speed [3]. Mesoscopic traffic simulation models, on the other hand, have the ability to account for individual vehicles, but are still primarily concerned with traffic dynamics of the vehicles as a whole and do not explicitly consider the details of vehicle lane changing and vehicle following behaviour, nor changes in vehicle speeds [3, 8]. Finally, microscopic traffic simulation models explicitly account for individual vehicle motion characteristics (*i.e.* acceleration, deceleration and lane changes) and typically employ some form of *vehicle following model* [3]. In this paper, the design of a microscopic traffic simulation modelling framework is described. The framework is intended to be used for the investigation of novel self-organising traffic signal control algorithms which make use of live, real-time data associated with individual vehicles, such as vehicle speed and distance from an intersection, thus necessitating the accurate modelling of vehicle acceleration and deceleration, vehicle following distances, lane changes and turning profiles.

2 Simulation modelling paradigms

A simulation model is described by Banks *et al.* [2] as the imitation of a real-world process or system over time such that the behaviour of the system can be studied. If the model is a sufficiently realistic imitation of the real-world process, then data may be collected from this model as if it were collected directly from the real system under observation. Over time, simulation models have become extremely useful— almost indispensable, in fact — when analysing and verifying theoretical models which may be too difficult to analyse on a purely conceptual level [5]. When building a simulation model, there are four main distinguishable approaches that may be taken to replicate a real-world system. These four approaches are *system dynamics modelling, discrete event modelling, agent-based modelling* and *dynamic systems modelling* [1]. Because of space constraints, only agent-based modelling will be discussed further here as this is the method typically employed when building microscopic traffic simulation models. The reader is, however, referred to [6] for descriptions of the remaining three simulation approaches.

Agent-based modelling is most simply defined as a decentralised, individual-centric approach to model design [1]. With agent-based modelling the behaviour of the global system as a whole is not defined, but rather the behaviour of its constituent entities, or agents. These agents can be anything from people at a train station to companies in a specific business sector, or, as is the case in a microscopic traffic simulation model, vehicles on a road network. It is from the interactions among these agents that the global behaviour of the system emerges and may be studied [4].

3 A microscopic traffic simulation modelling framework

The agent-based traffic simulation framework described in this paper was developed in *AnyLogic*, a Java-based multi-method simulation software package. AnyLogic was chosen as the modelling platform as it supports agent-based modelling and contains a basic road traffic library [1]. The framework was designed to be used as a testing mechanism facilitating the investigation and comparison of the effectiveness of previously proposed or novel traffic control strategies. Building a model within this framework may be accomplished in two separate stages. The first stage involves the design of the road network itself as well as basic traffic signals at each intersection. The second stage involves populating the road network with vehicles and defining the logic responsible for the movement of these vehicles through the system (*e.g.* the desired speed of vehicles, when vehicles should accelerate or decelerate, what this rate of acceleration or deceleration should be, and the origins and destinations of vehicles). Once built, the model may be used for data collection and analysis purposes. The data thus collected may be used to evaluate metrics which act as performance measure indicators for traffic signal control strategies. Examples of the data that may be collected include the number of vehicles present along a given lane of a given road segment, the speeds of individual vehicles, the distances of individual vehicles from an intersection, or whether vehicles are queued or travelling at their desired speeds. Examples of performance measure indicators include the delay times experienced by vehicles in the system as well as the number of stops made by the vehicles.

3.1 Building the road network and traffic signals

Before building the road network, it is required that certain global parameters are defined which dictate the appearance and connectivity of the road network. These parameters are the scale of the road network, the connection tolerance and the lane widths. The scale defines the number of pixels per metre, thereby linking the unitless display of the modelling framework graphic with an actual unit of length. The connection tolerance (measured in pixels) is the maximum distance between two lane ends for which the two lanes are considered to be connected, *i.e.* if two lane ends are closer than the connection tolerance and form an obtuse angle, they are considered as connected, and a vehicle that exits one lane may continue travel on the other. The lane width (measured in metres) defines how wide each lane in the road network will be, and as a result, how many lanes each road segment will contain. For example, if a line has a width of 60 pixels and the scale of the road network is 10 pixels per metre, and the lane width is set to 3 metres, then the corresponding road segment will comprise two lanes. The default speed limit for the road network is also user-defined and is measured in metres per second.

The traffic signals positioned at each intersection are modelled as individual agents and potentially operate independently of one another in order to facilitate the use of self-organising traffic control strategies. The signal switching logic is controlled by means of a state chart which comprises various states and state transitions. The number of different states in the state chart is determined by the number of phases which comprise a complete signal cycle at the intersection. The transition from one state to another is determined by the type of signal control implemented. In the case of fixed pre-timed control, a time-out function is employed such that the transition from the current state to the next state is triggered once a user-specified amount of time has elapsed since the current state was entered. For more advanced, vehicle-actuated traffic signal control strategies, state transitions may be triggered when a specified boolean condition is true, or upon receipt of a specific message string.

3.2 Populating the road network

With the road network and traffic signals in place, the next step is to introduce vehicles into the simulation model. Vehicles enter the road network at designated entry points. These vehicle arrivals may be defined according to one of four user-specified methods. The vehicles may arrive at a user-specified rate, in which case arrivals are stochastic and follow a Poisson distribution with a mean equal to the chosen rate. This is equivalent to specifying exponentially distributed interarrival times between vehicles with a mean equal to the inverse of the chosen rate. Alternatively, the user may specify an interarrival time which would be identical for all arriving vehicles. The user may also choose to implement a stochastic rate schedule which defines how the arrival rate changes over time. Finally, the user may define a deterministic arrival schedule, in which case the arrivals of vehicles are generated according to the exact times defined in the arrival schedule.

When a vehicle is generated, several vehicle-specific parameters are defined instantaneously. These include the origin-destination pairing of the vehicle, the size of the vehicle, the vehicle's rates of acceleration and deceleration, and the vehicle's desired speed of travel. Vehicles are generated at each entry point to the road network, and upon generation the final destination of the vehicle is determined by Monte Carlo simulation. This origin-destination pairing of the vehicle dictates when and where a vehicle must change lanes, as well as whether it should turn left or right at an intersection, or carry on travelling straight. Monte Carlo simulation is used to determine the size of the vehicle generated. The user decides on the probabilities associated with the different sizes of vehicles which ultimately determines the number of small, medium and large vehicles present in the road network. A vehicle's size determines its rates of acceleration, deceleration and desired speed. Typically, the larger the vehicle, the slower its rates of acceleration and deceleration, and the lower its desired travel speed. These trends may, however, be overridden by the user.

Apart from the logic which determines how fast a vehicle travels, or at what rate it accelerates or decelerates, logic has also been implemented which determines when and where a vehicle must accelerate or decelerate. Associated with each vehicle are minimum and maximum allowable distances to the vehicle in front of it, which depend on the vehicle's speed, as well as minimum and maximum allowable speeds, which, in turn, depend on the distance to the vehicle in front of it. There is also a maximum speed allowed on curved roads (*e.g.* corners). Let v_i be the speed of vehicle i and let $s_{i,i-1}$ be the distance between vehicle i and vehicle $i-1$ in front of it. Now, if $s_{i,i-1}$ is less than the value of some function f of v_i which determines the minimum allowable distance between two vehicles or if v_i is greater than the value of some function g of $s_{i,i-1}$ which determines the maximum allowable speed of a following vehicle, then vehicle i will decelerate. On the other hand, if $s_{i,i-1}$ is greater than the value of some function f' of v_i which determines the maximum allowable distance between two vehicles or if v_i is less than the value of some function g' of $s_{i,i-1}$ which determines the minimum allowable speed of a following vehicle, then vehicle i will accelerate. The maximum speed on curved roads is determined according to a function h of the radius of the arc of the curve. The functions f, f', g, g' and h are all user-defined.

The logic responsible for a vehicle's interaction with traffic signals operates in much the same manner. When a red or a late amber signal is displayed, vehicles decelerate as if there were a stationary vehicle at the stop line of the intersection. In the case of permissive right-turning vehicles, while a green signal is displayed, the vehicle turning right will wait in the intersection until the intersection and a portion of road, which extends a user-specified distance on the opposite side of the intersection, are free of any on-coming traffic, at which point the vehicle will complete its turn. For the case in which a right-turning vehicle is still present in the intersection when the traffic signal changes from green to amber, the vehicle need only wait until the intersection is free of any oncoming traffic before completing its turn.

3.3 Data collection and assimilation

The traffic simulation modelling framework described above was designed to allow for testing traffic control algorithms which assume the use of radar detection technology mounted at the intersection. Such radar sensors typically achieve a detection range of up to 275 metres [9] and are capable of detecting and tracking the speeds and positions of individual vehicles along a road segment, thereby enabling them to determine the vehicles' estimated times of arrival at the intersection. It was therefore necessary to incorporate this logic into the model.

Three lists are associated with each lane adjoining an intersection:

CarList. This list contains all the vehicles present on a lane. As a vehicle enters the lane, be it at the lane's entry point or as a result of a lane change, it is added to this list. A vehicle is removed from the list when it reaches the end of the lane or when it changes onto an adjacent lane. This list provides the user with information on the number of vehicles present along a specific lane. It makes it easier for a user to access individual vehicles and their associated characteristics, such as speed.

Queue. This list contains all queued, motionless vehicles along a lane and is a subset of the previously mentioned list, *CarList*. A vehicle is added to this list as soon as its speed equals zero. It is removed from the list as soon as it begins accelerating from rest. The list provides the user with information on the queue length along a specific lane.

QPred. This list contains all vehicles both currently stopped and queued as well as those which have not yet stopped, but will become queued before the existing queue has been cleared, and is again a subset of the first list, *CarList*.

In order to predict which vehicles will become queued, it is necessary to predict where the back of the queue will be. The predicted vehicle queue and back-of-queue position are calculated continually by an algorithm. For every vehicle not in the predicted queue list *QPred*, the algorithm calculates the amount of time it will take the vehicle to reach the current back-of-queue position. The algorithm then compares this time to either the sum of the remaining red time and the time required to clear the current predicted queue of vehicles (if the traffic signal displayed is not green) or just the time required to clear the current predicted queue of vehicles (if the signal displayed is green). If this time is found to be shorter, then the vehicle is added to the predicted vehicle queue list *QPred* and the back-of-queue position is incremented by the length of the vehicle plus the minimum space gap between stationary vehicles. For the case in which the front vehicle along a lane is not yet queued (*i.e.* the predicted queued vehicle list *Queue* is empty) the vehicle is added to the predicted queued vehicle list under one of two conditions. If the traffic signal displayed is green and the vehicle cannot clear the intersection before this signal changes to amber and ultimately to red, then it is added to the list. Analogously, if the traffic signal displayed is not green and the vehicle will arrive at the intersection before the signal changes from red to green, then again, it is added to the list. Vehicles are removed from the predicted queued vehicle list when they depart from the associated lane, at which point in time the predicted back-of-queue position is decremented by the length of the vehicle together with the minimum space gap between stationary vehicles. An example of a typical intersection scenario may be seen in Figure 1.

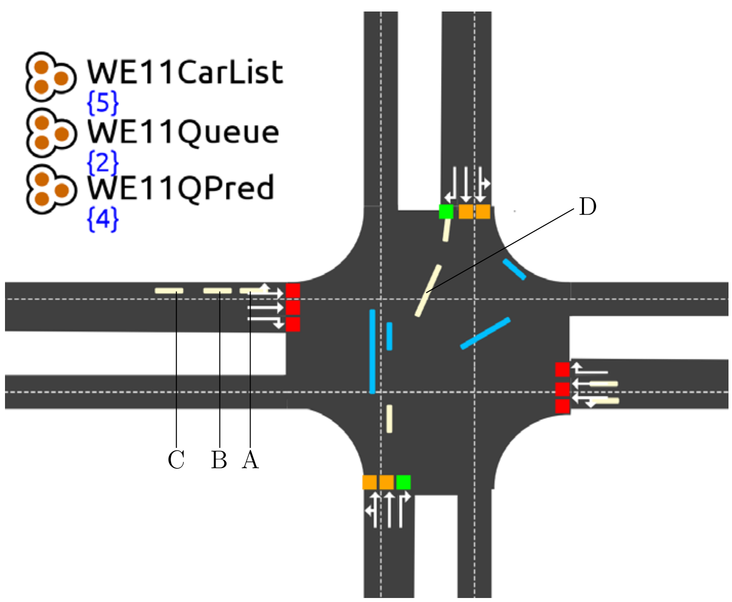
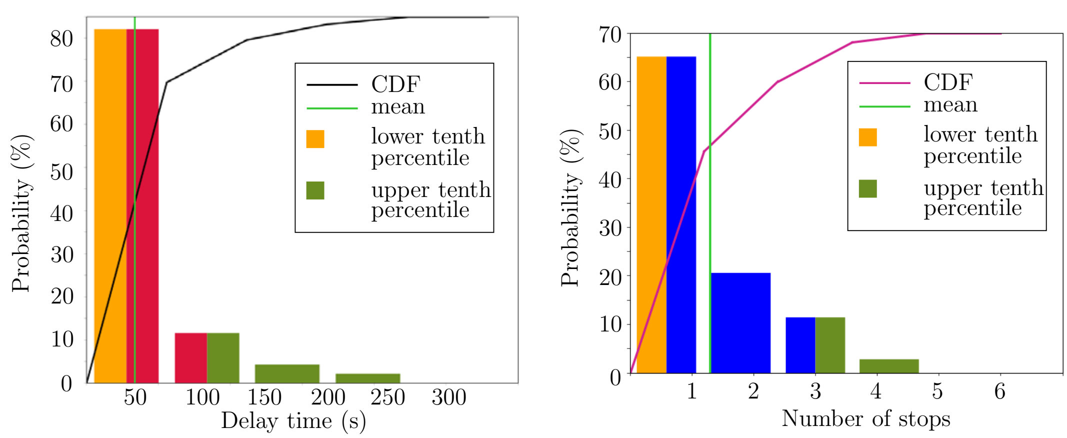


Figure 1: An example of an intersection scenario implemented in the modelling framework described in this paper. The list labelled *WE11CarList* is a list of all the vehicles present along lane 1 approaching the intersection (intersection 1) travelling in a West-to-East direction, and in this scenario has a size of five. The list labelled *WE11Queue* is a list of all the stationary vehicles along the lane, which comprises the two stationary vehicles in this scenario (labelled *A* and *B*). The vehicle labelled *C* has not yet come to a complete stop and therefore has not been added to the list. The third list, labelled *WE11QPred*, is a list of all the vehicles that are predicted to become queued and thus delayed. In the scenario depicted this list contains four vehicles. This means that the fifth vehicle along the lane will clear the intersection without becoming queued. The vehicles travelling in South-to-North and North-to-South directions are receiving an exclusive right-turn phase. The vehicle labelled *D* is currently waiting in the intersection while the three vehicles travelling in the opposite direction clear the intersection.

Due to the fact that the desired speed, as well as the origin-destination pairing of a vehicle, is known upon its generation and therefore, the total distance the vehicle is to travel, the delay time a vehicle experiences while travelling through the road network may be calculated by subtracting the time it would take the vehicle to move from its origin to its destination without being impeded by any traffic signals and resulting queues or slower moving vehicles from the actual time it spends travelling through the road network. The minimum time a vehicle can spend travelling through the road network is calculated by dividing the distance the vehicle has to travel from its origin to destination by its desired speed. The actual time spent by a vehicle travelling through the road network is captured by a timing mechanism which records the time the vehicle enters the road network as well as the time it leaves the road network.

The average delay time of all vehicles which pass through the road network is an important performance measure indicator as it provides the user with an idea of how different traffic signal control algorithms perform in respect of their ability to minimise driver delay under various prevailing traffic conditions. This feature also provides information on the maximum amount of time a driver was delayed, another important performance measure indicator to consider. A third performance measure indicator implemented is that of the number of stops a vehicle makes while travelling through the road network. An integer value is associated with each vehicle and is initialised as zero. Each time a vehicle comes to a complete stop, this value is incremented by one. The average number of stops made by vehicles can provide the user with an idea of the efficiency of traffic signal control algorithms in respect of their propensity to facilitate green waves, because the fewer vehicles that are required to stop as a result of red traffic signals, the lower their delay time is likely to be.

The framework allows for real-time analysis to take place as output is generated and the results of such an analysis can be displayed while the model is running. An example of this output is shown in Figure 2.



(a) Probability density function and cumulative distribution function of the delay time experienced by vehicles.

(b) Probability density function and cumulative distribution function of the number of stops made by vehicles.

Figure 2: An example of dynamic output generated by a model in the framework described in this paper while it runs.

4 Conclusion

The traffic simulation modelling framework described in this paper forms an integral part of an ongoing study into the feasibility and effectiveness of self-organising traffic signal control algorithms. It is intended that the framework, as well as all associated code, be made publicly available in the future. The model provides the user with an analytic tool which may be adjusted to suit the specific modelling requirements of the user.

Although the modelling framework described in this paper was built with the intention of investigating and comparing various traffic signal control strategies, it is not limited to such investigations alone. It may be used to investigate the effects of other factors on the system as a whole, such as the addition or removal of lanes from road sections, disruptions as a result of vehicle breakdowns, building of pedestrian crossings, or the installation of speed cameras, to name but a few.

It is the intention of the authors to implement novel self-organising traffic signal control strategies within the traffic simulation modelling framework described here so as to showcase the potential and benefits of the application of self-organisation to traffic control optimisation and congestion reduction as well as the benefits of using radar detection as opposed to conventional electromagnetic induction loop detectors. Another potential focus area for future work is the development of road network topologies of varying size and configuration so as to investigate under what conditions the various signal control strategies, as well as the types of detection equipment are most, and least effective.

References

[1] The AnyLogic Company, 2014, *AnyLogic help*, [Online], [Cited April 23rd, 2014], Available from http://www.anylogic.com/anylogic/help/

[2] Banks J, Carson JS, Nelson BL & Nicol DM, 2001, *Verification and validation of simulation models*, pp. 367–397 in Fabrycky WJ & Mize JH (Eds), *Discrete-event system simulation*, 3^{rd} Edition, Prentice-Hall, Upper Saddle River (NJ).

[3] Barceló J, 2010, *Models, traffic models, simulation and traffic simulation*, pp. 1–62 in Barceló J (Ed), *Fundamentals of traffic simulation*, Springer, New York (NY).

[4] Borshchev A & Filippov A, 2004, *From system dynamics and discrete event to practical agent based modeling: Reason, techniques, tools*, Proceedings of the 22nd International Conference of the System Dynamics Society, pp. 25–29.

[5] Craig DC, 1996, *Extensible hierarchical object-oriented logic simulation with an adaptable graphical user interface*, Doctoral Dissertation, Memorial University of Newfoundland, St. John's.

[6] Einhorn MD, 2012, *An evaluation of the efficiency of self-organising versus fixed traffic signalling paradigms*, MSc Thesis, Stellenbosch University, Stellenbosch.

[7] Gyimesi K, Vincent C & Lamba N, 2011, *Frustration rising: IBM 2011 commuter pain survey*, International Business Machines, [Online], [Cited April 23rd, 2014], Available from http://www.ibm.com/press/us/en/presskit/35314.wss

[8] Papacostas CS & Prevedouros PD, 2001, *Transportation software*, pp. 626–652 in Curless L (Ed), *Transportation engineering and planning*, Prentice-Hall, Upper Saddle River (NJ).

[9] Wavetronix, 2014, [Online], [Cited May 5th, 2014], Available from http://www.wavetronix.com/en/products/smartsensor/advance/features