

A Review of Traffic Simulation Software

G. KOTUSEVSKI AND K.A. HAWICK

Computer Science

Institute of Information & Mathematical Sciences

Massey University at Albany, Auckland, New Zealand

Email: {g.kotusevski, k.a.hawick}@massey.ac.nz

Computer simulation of traffic is a widely used method in research of traffic modelling, planning and development of traffic networks and systems. Vehicular traffic systems are of growing concern and interest globally and modelling arbitrarily complex traffic systems is a hard problem. In this article we review some of the traffic simulation software applications, their features and characteristics as well as the issues these applications face. Additionally, we introduce some algorithmic ideas, underpinning data structural approaches and quantifiable metrics that can be applied to simulated model systems.

Keywords: traffic simulation; vehicular traffic; software review; metrics; algorithm.

1 Introduction

Computer simulation is more and more popular discipline in the field of science in general. People tackle different scientific problems using computer power through simulating artificial experimenting environments. They use these simulated environments to test scientific models in order to prove or disprove their feasibility and correctness.

Today's high machine power gives people the ability to simulate environments at a rate much faster than any real environment, thus any experiment conducted in the simulated medium would provide results minutes, hours, days or sometimes even weeks, months and years ahead of what the same experiment would provide conducted in the real world.

One of the systems that is best studied using a computer simulation is a traffic network. It is more common to experiment with traffic networks in a computer simulated environment because experimenting with traffic in the real environment is not practical.

Computer Scientists have come up with different strategies to simulate traffic systems which fall under the following two categories:

1. Macroscopic simulation including: statistical dispersion models [1], freeway traffic model [2] etc.
2. Microscopic simulation including: cellular automata [3], multi-agent simulation [4, 5, 6, 7], particle system simulation [8] etc.

as well as strategies combining these two approaches, such as the ones discussed in [9, 10, 11].

In order to investigate the different traffic simulation strategies, we have reviewed and compared various traffic simulator software packages which implemented these strategies. In this paper we give a general review of the investigated software packages, their features, differences, while also discussing some ideas and algorithms that can be applied to simulated traffic systems in order to get more realistic results on the microscopic simulation level.

We reviewed the following well known and widely used traffic simulation packages:

1. SUMO - Simulation of Urban Mobility, version 0.10.3

“Simulation of Urban MObility” (SUMO) is an open source, highly portable, microscopic road traffic simulation package designed to handle large road networks. [12]

2. Quadstone Paramics Modeller, version 6.4.1

Quadstone Paramics is a modular suite of microscopic simulation tools providing a powerful, integrated platform for modelling a complete range of real world traffic and transportation problems. [13]

3. Treiber’s Microsimulation of Road Traffic [14]

Treiber’s Microsimulation is a personal software project created by that author and used in his research of traffic dynamics and traffic modelling.

4. Aimsun, version 6.0.4

Aimsun is a simulation package that integrates three types of transport models: static traffic assignment tools; a mesoscopic simulator; and a microsimulator. [15]

5. Trafficware SimTraffic, version 6 (build 614)

SimTraffic is a simulation application part of Trafficware’s Synchro Studio package. It servers as a traffic simulator for Trafficware’s Studio, which also includes traffic lights synchronization application. [16]

6. CORSIM TRAFVU, version 6.1

CORSIM TRAFVU servers as a traffic simulation viewer and is part of the TSIS CORSIM software package. It provides animation and static graphics of traffic networks, using the CORSIM input and output files created by a licensed user of TSIS CORSIM. [17]

We compared their features with respect to the following criteria:

1. Open Source and Free Use
2. Operating System Portability
3. Package Documentation and User Interface
4. Creating traffic networks and associated vehicle patterns
5. GUI Simulation and quality of graphic representation
6. Simulation output (data and files)
7. Ability to simulate very large traffic networks
8. Additional capabilities
9. CPU and Memory performance

These are explained in more details in the following sections.

2 Open Source and Free Use

Open source software gives users the right to use the software free of charge, a feature that is not very common in commercial software packages. However, open source projects are becoming more and more popular because they give their users the right to use, study and modify the program without any restriction.

Out of the six software packages that we considered, only two are free to use [12, 14], while the other 4 are paid [13, 15, 16, 17]. While the free packages could be investigated to their full potential, we could only study the demo versions of the four commercial packages. The demo versions had 30 days usage restrictions and/or feature restrictions.

Another feature of the free software packages (SUMO and Treiber's Microsimulation of road traffic) is that their source codes are freely available for download and use. The difference between the two packages is that SUMO is actually an open source project that is being developed by two different institutions, while Treiber's Microsimulation is a personal software project whose source code has been made available.

The source code is not available for study, modification or research for users of the other four commercial software packages (Paramics Modeller, Aimsun, SimTraffic and CORSIM).

One of the most popular features of open source projects is that they can be further modified by other programmers, as already mentioned previously. This feature supports the potential for parallelizing simulation models and packages to explore high end computer systems.

3 Operating System Portability

Operating System portability is another feature of a software package that is becoming more and more popular with the development of new and improved operating systems. This is also the biggest issue that most of the traffic simulation packages we reviewed face. We have found that only two packages [12, 14] have the ability to work under multiple operating systems, including the most commonly used today Microsoft Windows, Linux and Mac OS X. While the other four packages [13, 15, 16, 17], which are also the commercial ones were only able to work under the Microsoft Windows XP and Microsoft Windows Vista operating systems.

The inability to use software packages on Linux and Mac OS X operating systems, reduces and restricts their audience and user pool; more and more so everyday these two operating systems become more popular and used in the world.

4 Package Documentation and User Interface

In complex software systems such as traffic simulation packages, it is of utmost importance for users to have an easy to understand and use graphical user interface and user manual.

We were not able to locate a user manual for only one [15] of the reviewed software packages. However, all the other simulation systems had a well written and easily understandable user manual. The user manuals we compared all included their respective software's specifications, features, and usage explained, which can be very helpful for a new user of a software package.

Concerning the Graphical User Interface (GUI) of the simulation packages, we have found most of them to be really easy to understand and use, especially after following the user manual and the GUI explanations in it. Only one package [15] had a more challenging GUI to use, but that is mainly due to the fact of not having a manual to run the user through its operation.

It needs to be noted at this point that the GUI of the packages was only reviewed for its easy to follow set up for an early user of the software. Understandably, for an experienced user of a particular software application (no matter how complex its GUI), the software would be easy to use.

5 Creating traffic networks and associated vehicle patterns

While reviewing these traffic simulation applications, we have found different approaches as to means of creation of traffic networks and vehicle movement patterns.

We were able to find a graphical editor for traffic networks for four [12, 13, 15, 16] of the reviewed applications, while two of them did not have this feature. It is important to mention that the SUMO package did not include its own graphical editor for traffic network, but there is a SUMO Traffic Modeler software application available as an open source and free to use application. However, we were not able to study this modeler due to the constant errors it was bringing up. Additionally, the SimTraffic package purports to include a graphical editor in its full version, while this was not the case for its demo version we reviewed.

Concerning the approaches and ways to create the traffic networks, none were available in the demo versions of SimTraffic and CORSIM TRAFVU, as well as in Treiber's Microsimulation which only simulates predefined scenarios. However, we found the following techniques available under the mentioned software package:

1. SUMO simulation package:

- Manual (by hand) write up of a traffic network in an XML file;
- Import networks created in other traffic simulation applications;
- Using an automatic network generator creating three different types of networks:
 - Grid network
 - Spider network
 - Random network

2. Quadstone Paramics Modeller package:

- Using an automatic network generation wizard (that generated only a simple two zone network in the demo version) - apparently in the full version of the application a user could automatically create a traffic network using the wizard, and then further customize it using the graphical network editor available. However, we were not able to confirm this in the demo version.

3. Aimsun simulation package:

- Manually drawing of the traffic network using the available graphical network editor.

We also found various techniques to define vehicle movement patterns within the simulation applications, some of which were more user friendly than others. Difference exists between the packages because of the models they use, as previously mentioned micro or macro model. As an example the SUMO software package is very different in this sense from the other applications because it is the only application where each vehicle (agent) knows its own destination and list of edges it needs to pass until this destination is reached. This feature brings up issues of memory usage which will be addressed further in Sect. 12 on page 49. However, the other software applications included very similar techniques of vehicle movement patterns generation. The following list presents the approaches we acquired under the certain software package:

1. SUMO simulation package:

- Manual (by hand) vehicle route definition using four different types of definitions:
 - Flow definitions;
 - Flow and turning ratios;
 - Origin-Destination (OD) matrices;

- As well as, the usual for this application, manual edge by edge, vehicle by vehicle XML route definition.
 - Import vehicle routes created in other traffic simulation applications;
 - Using a random vehicle route generator.
2. Quadstone Paramics Modeller package:
 - Applying an OD matrix;
 3. Aimsun simulation package:
 - Applying OD matrices using zone supply and demand of traffic.
 - The package also included a vehicle route randomization.

We were not able to review any means of creating vehicle routes in the demo versions of SimTraffic and CORSIM TRAFVU, but they should be available in the full versions of the software packages. Additionally, Treiber’s Microsimulator includes some kind of statistical distribution of vehicles where the user can basically define the number of vehicles emitted per hour from a certain intersection as well as the ratio between cars and trucks in certain scenarios.

Comparing the results we acquired from investigating the different applications, we can conclude that most of the packages include graphical editor for traffic networks, and also most of the packages use a statistical method for distributing vehicular traffic through their traffic network. While application of OD matrices is the most popular approach to define vehicular routes in these applications, we found this feature to be very difficult to use in some and very easy to use in other software packages.

To further explain the issues we encountered with OD matrix application we take examples from Paramics Modeller and SUMO packages. In the Paramics Modeller package, OD matrix application is created to be more visual and user friendly, while in the SUMO package OD matrix application is more XML writing and using command line. OD Matrix application in Paramics Modeller is done by editing a simple table. This table, while being edited, has a feature to preview the zone we are editing in the viewing panel, coloring it with specific color and pointing an arrow. This way, a user can very clearly see what area he or she is defining the traffic supply and demand to and from. While on the other hand, the SUMO package OD matrix application was more writing intensive. A user has to create an XML file using a specific pattern, then using command line map the OD matrix to create a trip list which can then be used in the simulation to navigate the vehicles through the traffic network. We found the second approach to be more difficult to use than the one Paramics Modeller uses, mostly because users tend to understand better something they can see like in the Paramics case. While an XML can also be very easily readable and understandable it still lacks the visual effect Paramics Modeller has and having to write down each district id, and each source and destination id’s without making a mistake can also prove to be a difficult task by itself. Therefore, we found SUMO package’s approach to be much more difficult and less user friendly to use.

6 GUI Simulation and quality of graphic representation

One of the best ways for a non-technical user to see results of a traffic simulation is to actually view the simulation running in real time. Graphical representation of the traffic simulation is a good method to examine what exactly and at what periods in time something important happens. This method can also help determine if there is a positive or negative effect on traffic flow if something changes during the simulation or there is some kind disruption in the traffic pattern.

Different simulation packages included different graphical representation of the running simulation, one of the packages [12] we reviewed also included a non-graphical simulation that only

produces output files and statistics by which we could measure and determine what has happened within the traffic network. While all of the simulation applications included some kind of graphical representation of the simulation there still existed major differences between the quality of these representations. The key distinction consisted of the graphics being two or three dimensional. Three [12, 14, 17] of the reviewed packages had only two dimensional representation, while the other three packages had also an option for a three dimensional simulation.

Concerning the two dimensional preview of the simulation packages we found it to be a regular bird-view of a traffic network with fairly easy identifiable representation of the various vehicles. However, there were a few differences between the three dimensional representations of the software packages that included this feature ([13, 15, 16]).

We found that the Quadstone Paramics Modeller had the best graphics representation of all packages. It included libraries of textures to use for city buildings, vehicles, pedestrians etc. and thus making the simulation look much more realistic than the other packages, as pictured in Fig. 1. Paramics Modeller also had the feature for a user to add as many cameras as needed viewing the simulation from different points of reference. The user could actually have views of all points of interest within the traffic network open at the same time while the simulation was running and could very clearly examine the pattern of traffic flow through them.



Figure 1: Paramics Modeller traffic simulation running, showing three dimensional representation using the available graphics libraries to preview buildings around the traffic network, as well as traffic members such as cars, trucks, public transport and pedestrians [13]

The Aimsun package included a basic three dimensional preview of the simulation. It also had the feature to add as many cameras as needed to capture all the points of interest of the traffic network, as pictured in Fig. 2, but did not include as many different textures as the Paramics Modeller. Aimsun's simulation also dropped in frames per second rapidly when we used a three dimensional preview with multiple view points. Further details on CPU and Memory performance and the platform used to review these packages is given in Sect. 10 on page 46.

Trafficware's SimTraffic only included a two dimensional view, but it was mentioned in its user manual that the application supports three dimensional view. The explanation given in the manual

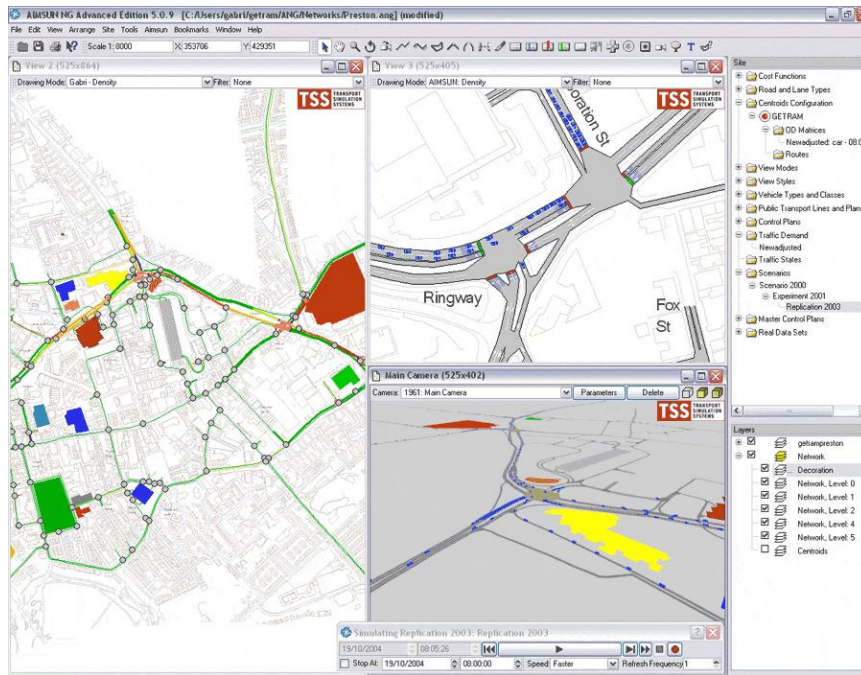


Figure 2: Aimsun traffic simulation package, illustrating how multiple views are used to capture different points of interest in a traffic network. Each view point with its own angle, representation and graphics quality settings [15]

clarified that the software has the option to save graphics data into an s3d file. This file is said to contain the information needed to display SimTraffic's three dimensional representation in a 3D graphics package. The file also includes vehicle and pedestrian locations at each time slice and geometric data for the simulated traffic network. However, at the present time there is no viewer that supports this file format available. Trafficware has agreed to provide the s3d file format specifications on request to software developers who wish to create a 3D viewer or a conversion utility to convert 3D graphics information into another file format.

7 Simulation output (data and files)

Although following a graphical representation of a traffic simulation can be one of the best methods of determining the traffic flow through a specific traffic network, statistical output always gives additional information that can escape the human eye when viewing a real time execution of a simulation. This is the reason why most of the reviewed software packages included some kind of output files or data. However, we were only able to investigate the output of the SUMO package to its full extent while the demo versions of the other applications did not include the statistical output, which was said to only be included in the full version of the products. The following list includes the approaches we have found under the different packages:

1. SUMO simulation package - SUMO included simulation output through generating output files. The output files contain the following:
 - Network state dump of a certain point of time in the simulation;
 - Lane/Edge dump of a certain point of time in the simulation;

- Detectors dump - detectors are used to locate a point of interest within this simulation package. Using this output file creates a dump of the specific area where the detector is located;
 - Net-wide vehicle emission states and travel times - an output file concerned with the emission states of the vehicles running through the simulated traffic network and their travel times;
 - Vehicle-oriented trip information - file generated to output the route information of the vehicles within the simulated traffic network;
 - Output coupled to traffic lights - output file generated to dump certain actions connected to the traffic lights and how their pattern affected the traffic flow.
2. Quadstone Paramics Modeller - this package included tools to statistically represent what is happening in the simulated traffic network. The measurements included in the software applications include queuing patterns, peaks and troughs in demand, speed, density, line of sight, journey time, etc. Additionally, all these tools could be used during run time by the user, so he or she could calibrate the model “on the run”. [13]
 3. Aimsun simulation package - Aimsun includes more than 20 different view styles for graphical representation of statistical information about the traffic and the events occurring in the ongoing simulation of the traffic network. However, we were not able to investigate them in the demo version of the software application.
 4. SimTraffic simulation package - Trafficware’s SimTraffic seemed to have different types of reports available as an output from a traffic network simulation, but we were again not able to investigate them because they were not included in the demo version of the application.
 5. CORSIM TRAFVU - this software package, as already mentioned in Sect. 1 on page 35, is only a graphical player for previewing the simulation generated by the CORSIM package, therefore it did not include any type of output files. However, it was noted that the full version of TSIS CORSIM package did include some ways of statistical representation and output files of the outcomes from a simulation of a certain traffic network.

We were really surprised to find out that there is no defined file or output standard that all simulation applications use. Looking from a user point of view, it would be very helpful for such a standard to exist because users of different packages can share their results and can compare their simulations using different packages with a consistent output of the same format. This is certainly an issue that producers of traffic simulation software need to work and agree on in the future.

8 Ability to simulate very large traffic networks

Traffic simulation software packages are mostly used to simulate and plan changes within a traffic network. Sometimes the traffic network that is being examined may be of a very large scale, like a downtown area of a large city or in some cases even a whole city, as pictured in Fig. 3. Therefore, it is important for a package to have the option of simulating very large traffic networks with thousands of roads and millions of vehicles running on them.

Most of the software applications we reviewed had the ability to cope with very large traffic networks. Using the demo versions of [13, 15, 16, 17] we were only able to study the examples that were included with the installed packages. All of the examples were fairly small networks, with a few more complex ones. As we expected, the larger the road network was the slower the traffic simulation ran. Additionally, we expected a simulation to run slower when using more complex graphic models, but instead we just had a lower frame rate while the simulation was running at the desired rate.

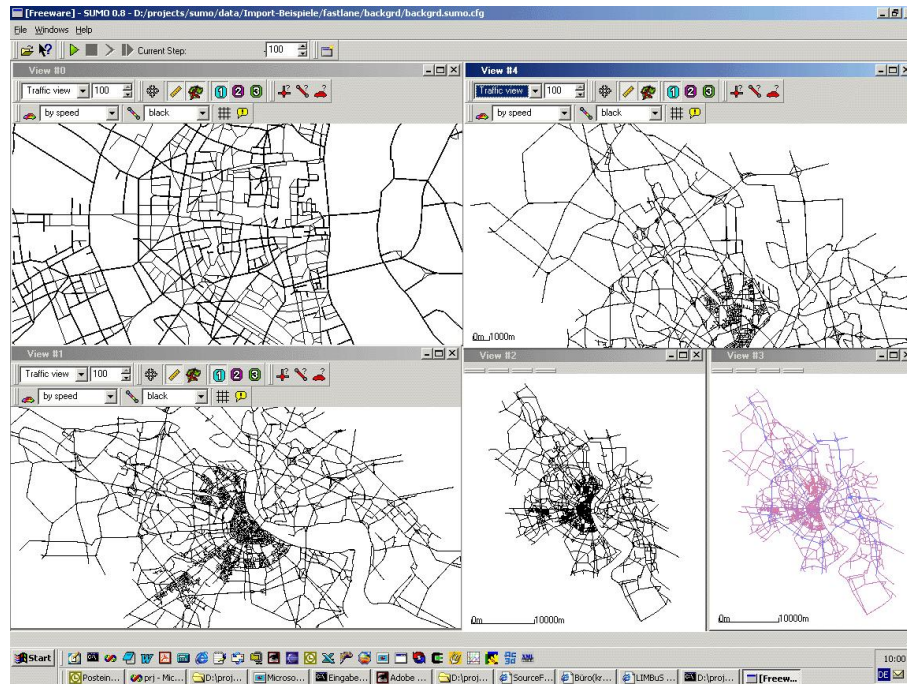


Figure 3: A traffic simulation of the city of Cologne conducted in the SUMO software package, illustrating a large traffic network simulation with several thousands of roads and intersections [12]

We also studied the SUMO [12] software package with a large traffic network and we received a similar result. The only important difference in this package was noted in its user manual, where we located information that the package could simulate traffic networks of up to 10,000 edges (roads).

9 Additional capabilities

Additional features give software packages more useful procedures and actions for the user to enjoy. In the packages we reviewed we have found a good amount of additional capabilities to be available. Starting with examples included with the installation of the software up to the means to record a video of a running simulation, these additional features are noticeable enough to provide better quality of the software use.

We have found the following additional capabilities that were worth mentioning in the simulation packages we reviewed:

- **Examples** included with installation - this feature was part of all packages with the exception to the Aimsun package. This came to our surprise because Aimsun proved to be the most difficult package to start working (because as already mentioned in Sect. 4 on page 37 there was no user manual present with the software), therefore we expected to at least have some kind of example traffic networks to simulate and gather results from. Additionally, examples were not included with the CORSIM TRAFVU package but they were available for download on the website of the software, one example is pictured in Fig. 4.
- Means to **convert** other **traffic simulator formats** - we found this feature in two [12, 15] of the reviewed packages, and they could convert traffic networks from most of the other software

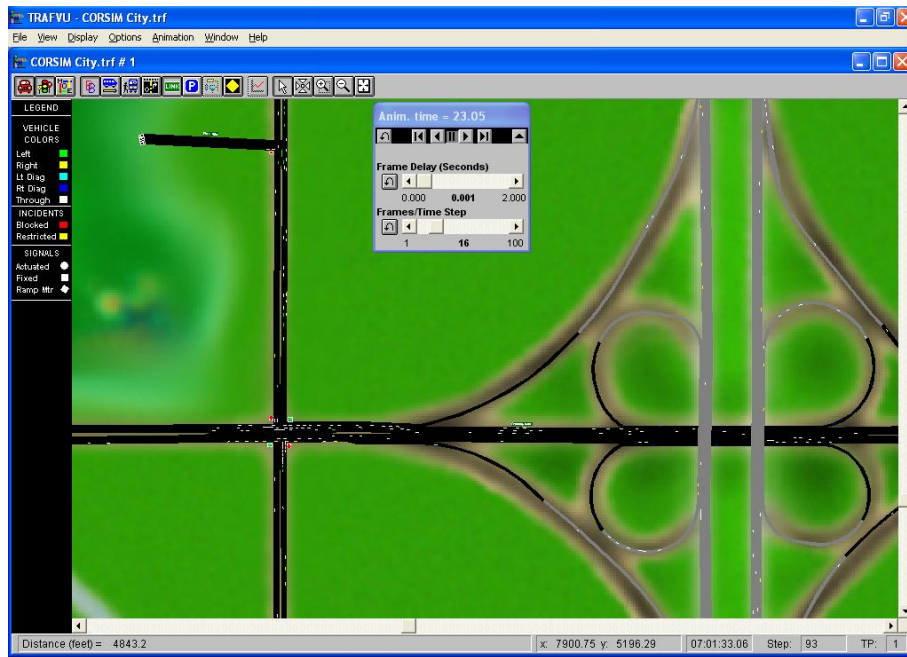


Figure 4: CORSIM TRAFVU software package - traffic network simulation of one of the example traffic networks available at the TSIS CORSIM website, the particular one simulating a spaghetti junction example [17]

packages. While the other four packages did not include this feature because of different reasons. We could predict that Treiber's Microsimulator would not have this feature because its scalability is not near any of the other packages. Also we knew that CORSIM TRAFVU would not have this feature due to the fact that this package is only a viewing environment used to preview and graphically represent the simulation already created with the CORSIM software package. But it was a surprise for us that the other two packages [13, 16] did not have this feature.

- **Different unit systems** and changing between them - maybe not as important to the average user, but still a viable feature for a traffic simulation package which was present in three of the reviewed packages but in different forms. For example the SUMO package only included means to convert the default m/s unit to km/h; Quadstone Paramics on the other hand includes three different systems in its package: UK, US and the Metric system, which were also part of the Aimsun package. However, the other three simulation packages did not have this feature where we suspect that for [16] and [17] this is due to the fact that we studied only their demo version, while for [14] this feature is not available due to the simplicity of this software project.
- Simulation of **different vehicle types** - all of the reviewed packages included this feature, some of them in more complex version while some in a rather simplistic version. To give an example, using the Paramics Modeller we could use a library of different vehicles, but we could also define our own vehicles while inputting the type, size, weight, color and other characteristics of the user-defined vehicle. The simple version of different vehicle types is used in Treiber's Microsimulation which only includes two types of vehicles: cars and trucks, a preview of them seen in Fig. 5.

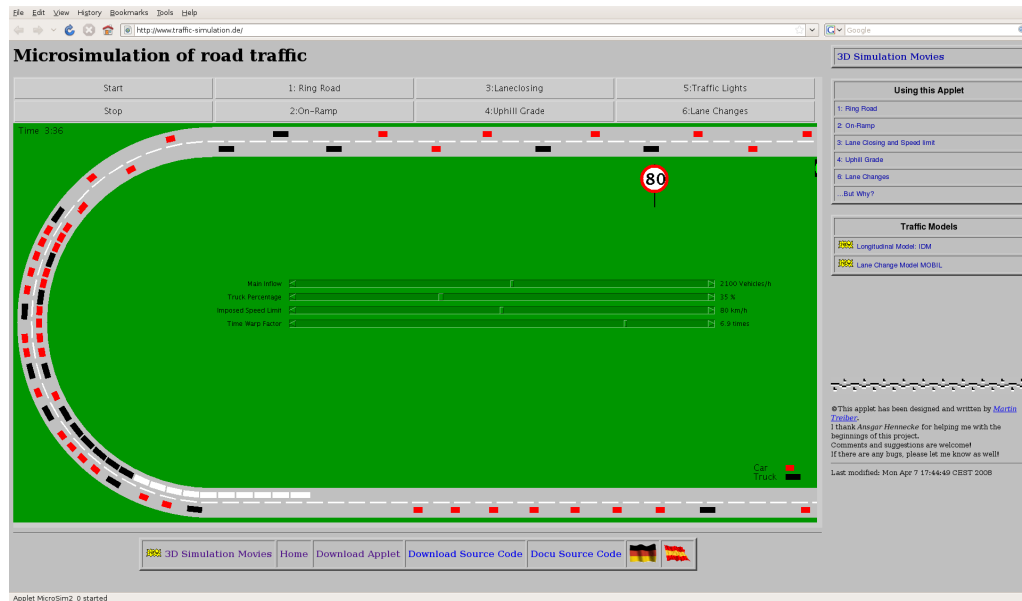


Figure 5: Treiber's Microsimulation of Road Traffic - illustrating the use of different types of vehicles used in the simulation: cars and trucks. The cars are represented with smaller rectangles in red color, while the trucks are larger rectangles in black color [14]

- Means to **configure traffic lights** - while all of the packages include some version of configurable traffic lights we found some to be difficult to structure. As an example for difficultly configurable traffic lights we mention the SUMO package because composing the traffic light pattern in SUMO is done manually (by hand) writing into an XML file (as most features of this package). Expectedly, we weren't able to study any of the means to configure traffic lights in the demo versions of the other software packages, while Treiber's Microsimulator included a traffic lights scenario in its predefined scenarios, but it was not possible to configure the pattern of the lights.
- Means of adding **public transport and public transport stops** - a key factor in city traffic, traffic jams and congestions is undoubtedly public city transport. As such, the feature to add public transport is of great importance for traffic simulation application. By adding buses, trains, trams etc. to the traffic network a user can see how they affect the flow through the network and what can be done to improve this flow if it is not at the desired level.

We found four [12, 15, 13, 17] of the reviewed applications to have this feature available. However, we were only able to study it in SUMO and Aimsun, while this feature was not part of the demo versions of Paramics Modeller and CORSIM TRAFVU. Additionally, we were not able to find presence of this feature in the SimTraffic package, but we suspect this is due to the reviewed version being only a demo. Furthermore, Treiber's Microsimulation did not include a public transport scenario to be studied.

- Means of **vehicle rerouting or closing a street** - a specific feature that can see a lot of use in simulating traffic during road works. Using vehicle rerouting a user can simulate and determine the flow of traffic in specific conditions that require rerouting of vehicles by closing a street.

Three of the demo versions we reviewed [13, 16, 17] did mention to include this feature in the full version of the software package, but we were not able to investigate it. However, we

were able to study it in SUMO and Aimsun. The SUMO package included vehicle rerouting using the (usual for this application) manual configuration of rerouting in an XML file; while the Aimsun package included this feature using specific scenario settings.

- Means to **restrict road access depending on vehicle type/class** - very similar to the rerouting feature, but dependent on the vehicle type. Again a feature that can be used to plan future road additions to a traffic network and determine the impact it would have on the traffic flow if different types of vehicles were restricted on not using the new road.

Comparing the different applications for this feature we received exactly the same result as in the vehicle rerouting feature. Three [13, 16, 17] of the demo applications user manuals again noted that this feature was included in their full versions, while the SUMO and Aimsun package had the usual XML configuration and specific scenario configuration of road access for different vehicle types respectively.

- Support for **right and left hand driving** - this feature is very important for broadening the potential user pool of one traffic simulation application. Support for both right and left hand driving ensures the package to be usable by a user from any country in the world, while support for only one side restricts this pool to only part of the countries of the world. That being mentioned, we found two [13, 15] of the reviewed packages to include this feature even in their demo versions, while we were not able to locate this feature within the SimTraffic and CORSIM packages. Additionally, we found this feature to be one of the very few that were not available in the SUMO package, while Treiber's Microsimulation did not include this feature because all the scenarios included in this application contained only one way streets.
- Means to **record a video** of the running simulation - a very useful feature for computer simulation applications when the user needs to have a video as an output of a particular traffic network simulation. However, we only found this feature in two of the reviewed software packages [13, 16]. While in Paramics Modeller we were able to record a video and encode it using any of the installed codecs on the computer, SimTraffic's video recording was only mentioned as part of the application's full version. The other four packages do not include means to record a video of the running simulation.
- Means to **print and screenshot** the current view of the simulation - similar to recording a video feature, but instead just printing or screenshotting a particular frame of the running simulation. This feature was however included in more packages than the video recording. We found it available in four [13, 15, 16, 17] of the reviewed packages (even in their demo versions); while this feature was not available in the SUMO and Treiber's Microsimulation packages.
- Simulation of **pedestrians** - this feature is certainly very important in simulating city traffic, especially in areas where there is high density of pedestrians that move either on the sidewalks or cross the streets. We found this feature available in three [13, 15, 16] of the reviewed simulation packages. We could not study the demo version of CORSIM if this feature is available, while SUMO and Treiber's Microsimulator packages did not include this feature in their applications at all.

10 CPU and Memory performance

We reviewed these simulation packages using a laptop with Intel Core 2 Duo Extreme processors running at 2.8GHz with 6MB of cache memory and 2GB of RAM memory.

Tab. 1 provides information about the CPU and Memory performance of the software applications while they were actively simulating a traffic network.

Table 1: CPU and Memory performance

	SUMO	Paramics Modeller	Treiber's Microsimulation of road traffic	Aimsun	SimTraffic	CORSIM TRAFVU
CPU usage	Between 5-17%, depending on the number of vehicles currently running on the traffic network	Constant 50%	N/A	Between 25-40%, depending on the number of vehicles and the scenario currently simulated	Constant 50%	Constant 50%
Memory usage	Between 12-16 MB, depending on the traffic network	Between 40-140 MB, depending on the traffic network and the graphic models used	N/A	Between 30-40 MB, depending on the traffic network	Around 35 MB, does not depend much on the traffic network simulated	Between 28-32 MB, depending on the traffic network

Our tests confirmed our expectations. We expected the simulations with the best graphics representation to use more memory than the other simulations that used only 2D or poorer 3D graphics representations. Concerning the CPU usage, we expected most of the simulators to have 50% CPU usage, we found out that one half of the reviewed simulators do use one processor to its 100% potential.

11 Metrics

The study and research of traffic engineers greatly helps in determining traffic parameters and measurements that are viable to be used for examining the outcome of a traffic simulation. Traffic system researchers introduce variety of traffic parameters and quantifiable metrics commonly used in traffic analysis and traffic modelling today. The following list is introduced by Taylor et al. in their Understanding Traffic Systems [18]; we have outlined the measurements, which form the basics for traffic analysis, and at the same time are of interest to us in determining and evaluating the different navigation algorithms we study in order to achieve the desired traffic flow, further explained in Sect. 13 on page 50.

- Traffic Flow or Volume - Number of vehicles per unit time
- Speed - Time per unit length
- Concentration or Density - Vehicles per unit length

Additionally, Mannering et al. in their Principles of Highway Engineering and Traffic Analysis [19], provide mathematical equations and explanations of traffic parameters such as

- Time Headway for a particular vehicle i - Time elapsed between the arrivals of vehicles i and $i - 1$

- Average Time Headway - the reciprocal of traffic flow or volume
- Spacing - Roadway length between successive vehicles
- Average Spacing - the reciprocal of traffic density
- Average Traffic Speed - Defined in two different ways:
 - Time-mean speed - the arithmetic mean of the vehicle speeds observed at some designated point along the roadway
 - Space-mean speed - determined on the basis of the time necessary for a vehicle to travel some known length of roadway

all of which are used to explain the relationship between the three basic measurements in more detail.

Taylor et al. also mention the relationship between the three basic parameters that are used to describe a traffic stream. They present this relationship in the following equation, which is also derived from the equations given in the work of Mannering et al. and Haberman in his *Mathematical Models* [20].

$$q = k\bar{v}_s \quad (1)$$

Eq. 1 gives us the traffic flow q as a product of the density k and the speed \bar{v}_s (space-mean speed). Additionally, Mannering et al. give a categorization of the metrics in order to associate them with the appropriate level of investigation. They refer to average time headway and average spacing as microscopic measurements, while metrics such as traffic flow, average speed and traffic density are said to be macroscopic measurements. [19] They also indicate that the microscopic measures can be related to the macroscopic measures as given in their study on traffic measurements, and also supported by the work of Taylor et al. and Haberman.

However, Eq. 1 generally applies only to uninterrupted traffic, such as traffic flows on main highways and freeways, but it is noted by Mannering et al. that it is also applicable to interrupted traffic flow, but that there are additional complexities involved. They go further to explain the key concepts and measures used in the analysis of interrupted traffic flow such as at a signalized intersection. Following is a list of measures, as given by Mannering et al. [19], which we will use to prepare and examine our traffic navigation algorithms.

- Saturation Flow Rate - the maximum hourly volume that can pass through an intersection, from a given lane or group of lanes, if that lane (or lanes) were allocated constant green light over the course of an hour
- Lost Time - the portion of the cycle length that is not being completely utilized. In other terms, because the traffic light alternates the right-of-way between conflicting movements, traffic flows are continuously started and stopped. Every time this happens there is a lag due to drivers reacting to the change of traffic light signal. This lag at the beginning and the end of green and yellow signal interval results in a portion of that interval to not be completely utilized. This lag is known as lost time
- The Effective Green Time - the time during which a traffic movement is effectively utilizing the intersection
- The Effective Red Time - the time during which a traffic movement is not effectively utilizing the intersection
- Capacity - the maximum hourly volume that can pass through an intersection from a lane or group of lanes under prevailing roadway, traffic, and control conditions

The measure of capacity is essentially one of the most important for us because it provides means to calibrate traffic lights intervals in such a way that we will achieve a desired traffic flow in all directions, thus giving us a way to minimize traffic congestion, queues and waiting times.

$$c = s \frac{g}{C} \quad (2)$$

Eq. 2 gives capacity c (usually in vehicles per hour) as the product of saturation flow rate s (vehicles per hour) and the ratio between effective green time g and the cycle time C (both of which usually given in seconds). Therefore, adjusting the traffic light interval we can influence the capacity of a certain intersection and bring it to the desired level so it can keep up with the traffic inflow passing through the intersection.

Furthermore, just like intersections, roads also have a value which we refer to as a road capacity value. We consider this to value to coincide with the point of maximum traffic flow of a road. We have considered this value in our research and we will cover this issue in more detail in Sect. 13 on page 50.

12 Discussion

In this section we present relevant issues we have discovered within the applications we reviewed as well as how they are significant and various ideas on how to proceed in order to overcome them.

One of the greatest issues in traffic simulation packages is deciding on the model that will be used in the simulation. Whether the model used should be microscopic or macroscopic.

In microscopic models, we have every simulated vehicle acting as an independent agent, while in macroscopic models, we have vehicles that are statistically distributed by some kind of emitter object at a certain predefined or dynamic rate. Each model has its own advantages and disadvantages. From the reviewed traffic simulation applications we have found that microsimulation models for example can have issues with the use of computer memory, while macrosimulation models did not provide any means to examine impacts a single vehicle had on the traffic network and the flow of other vehicles through the network.

This brings us to the issue of how detailed a traffic network needs to be as well as the scalability of the traffic network. Under scalability we understand how many millions or tens of millions of roads and vehicles can a system cope with. This is something that is particularly important for microscopic modelled simulators because a system has limited amount of memory as well as restriction due to the power of the CPU of the system. We have found that this issue is present in the SUMO simulation package, as already mentioned in Sect. 5 on page 38. This is the reason why this particular package was noted to be able to cope with networks of only up to 10,000 edges/roads, while in the other packages (using macrosimulation models) we could simulate a very large network without any performance issues.

However, it is also important to note that even microscopic simulators with a limited system in terms of memory and CPU power can cope with extremely large networks, but these simulations would not run in real-time. For example, we can simulate a city network using a microsimulator on a very slow computer with a very low amount of memory, but this would take a large amount of time and thus any result we get from the simulated experiment would be long overdue and therefore useless.

One very interesting approach to the problem of different models of simulation was found in the Aimsun package. Aimsun had integrated three models of simulation in the package, including micro, macro and meso simulations models. Additionally, the user is in control of which of these models would be used for a particular simulation. This feature gives a user the power to decide what is important in the simulation, and at what level the traffic flow will be examined.

Furthermore, we noticed that these traffic simulators only cope with **static environments**. Vehicles in these static environments had already been inserted knowledge of the road network,

their destination point and almost always a list of all the roads they need to pass in order to reach this destination.

However, if we want to simulate traffic the way it is happening in the real world, we must implement a simulation that will comprise a dynamic environment (traffic network). There are many factors that can affect and change the environment; they can be external factors (eg. closing down of a particular road), and internal (eg. the vehicles travelling in the traffic network by themselves also affect the network). All the various factors affect mostly the congestion of traffic within the traffic network, and congestion is something we strive to bring to a minimum point through our work.

13 Areas for further work

In this section we present the areas we will concentrate on in our future work. We have considered the issues we have discussed in the previous section and have set up a set of goals that we want to achieve through our future research.

As already mentioned in Sect. 12, the real world is a dynamic and always changing environment. This is why we are working on implementing a traffic simulation application that includes vehicles which are navigated by “smart” agents through a dynamic traffic environment.

Therefore, we are currently implementing a map data structure that will be searched in real time during simulation execution. In other words, each of our “smart” agent vehicles will be navigated through the traffic map by an algorithm that will dynamically search for the best path through the traffic map to the destination point.

Our Map data structure has two major components: Map Nodes and Roads, where roads are essentially the edges between our map nodes but with many additional properties and characteristics. One of these properties is the cost of taking this road on the path to reaching the destination point by our agents. Every change in the traffic network will influence the cost of a particular road, thus making it easy for our agents to evaluate and choose the one with the minimal cost.

At the present time we have assumed two types of changes that can occur in our traffic road map while the simulation is running:

1. Other vehicles congest a particular section of the road network;
2. The user deletes or adds a road.

It is easy to determine and adapt to a change in the network when a user adds or deletes a particular road within the traffic map. Our path finding algorithm currently re-searches the traffic map every time it comes across a new map node. In this way, we can account for any addition or deletion of a road (edge) from the traffic map and adapt our path to the destination node by taking only the present roads in the traffic network.

However, it is more challenging to implement an evaluation technique that will account for the other vehicles moving within the traffic network. They are constantly changing their positions and we cannot predict the next road to their destination. Coming back to the issue we introduced in Sect. 11 on page 47, we considered implementing a road capacity variable, that essentially represents the road’s optimal density. Fig. 6 shows how the optimal density gives the maximum traffic flow on a particular road, while the relationship between the two traffic parameters was explained in Sect. 11 on page 47.

Using this road capacity coefficient for each road in our traffic network, we can evaluate whether or not the road is heavily or lightly congested with traffic. An illustration of this phenomena is given in Fig. 7. Taking the congestion of the road in consideration, our algorithm can evaluate what the best route would be in order to reach their destination.

Considering the thin line that separates light and heavy traffic, the optimal density and maximum traffic flow, we need to define specific rules that our agents will follow in order to achieve

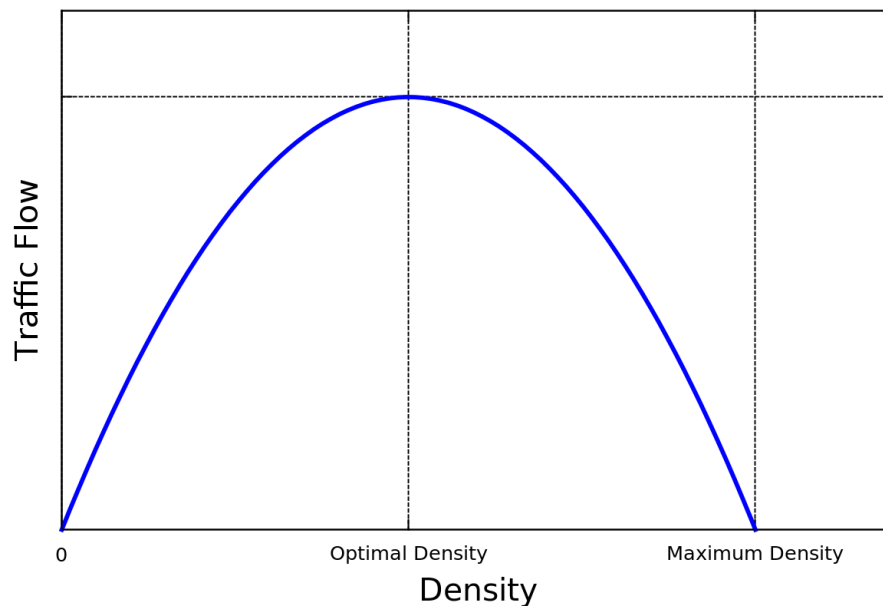


Figure 6: Flow-Density Diagram of Road Traffic, illustrating the relationship between density and traffic flow including a critical point (the optimal density point) at which the a road's maximum traffic flow is achieved

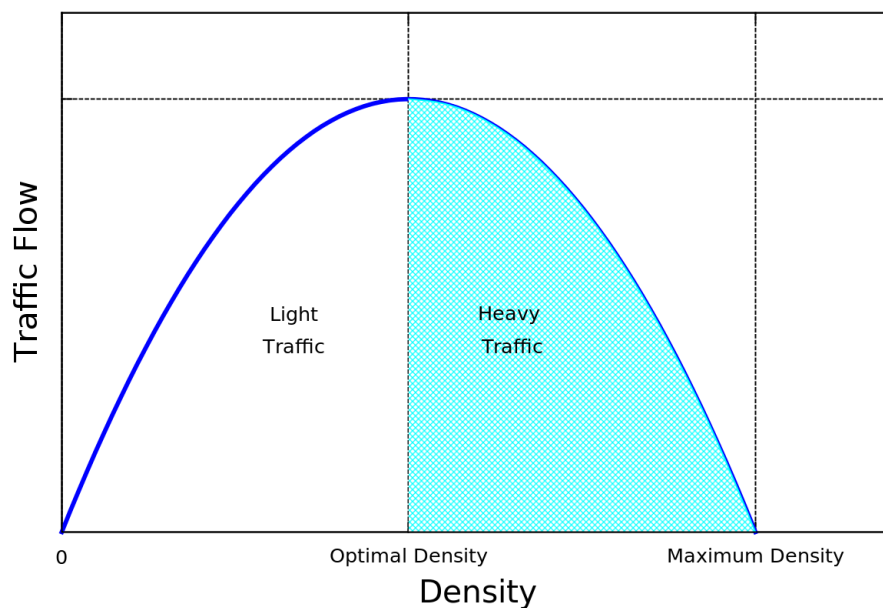


Figure 7: Flow-Density Diagram of Road Traffic, illustrating how the optimal density point separates the two areas of light and heavy traffic on a road

excellent traffic flow. If we define rules for our autonomous agents to go through less congested roads, ie. take roads whose current density is less than their optimal, we can increase the density in these roads. By doing so, we will maximize the traffic flow of the road, while still avoiding traffic congestion. If however, our agents cannot find a road with light traffic density, this will show us that the particular section of our traffic network where this even occurred cannot cope with the amount of vehicles that pass through it. Finally, this will be an indicator that the particular section of the traffic network needs to undergo changes in order to be able to keep up with the amount of vehicular traffic that demands to use it.

We have taken into consideration many additional issues, however we found these to be easily solved in the traffic simulation packages we reviewed. The following paragraphs will give an overview of these issues, how they relate to each other, why they are important to us, and some strategies we are planning to use to solve them.

One issue we have looked into is roads with multiple lanes. Which is relatively easy to implement, but the challenge is deciding on the structure and implementation of these lanes. We considered two options:

1. Implement number of left lanes and number of right lanes as a property under the `Road` class;
2. Implement a separate `Lane` class.

We decided to take upon the first approach, but this brought in an unexpected issue - lane changing. This brought us to thinking on approaches we could take to design the way vehicles change lanes, and even why vehicles need to change lanes? Two answers that immediately pop up are overtaking, and changing lanes to follow right/left turns. One of them is more relevant to motorway while the other to urban traffic, which is yet another issue that will be addressed further in this section.

Finally, we implemented a mixture of the two options we considered for the lanes. We implemented a separate class `Lane`, but this class serves the purpose of a container for information about the vehicles that are currently traveling on that lane. While at the same time we still have the number of left and number of right lanes as properties under the `Road` class, with an addition of two separate lists that contain collections of left and right lanes that belong to this road. Using these lists of lanes each road contains, and the list of vehicles each lane contains, we can build a relationship between the vehicles and the roads and we can say what vehicles are positioned on what roads and in what lane at any given time during the simulation.

We have also looked into an issue of one way roads and streets, but our implementation of left and right lanes on a road solves this problem by itself. In other words, instead of specifically pointing out that a particular road is used for only one way traffic, we can easily configure the road to have 0 (zero) left or right lanes, thus making it usable for driving in only one direction.

Coming back to the issue on motorway and urban traffic patterns, the basic difference between the two is that motorway traffic is uninterrupted traffic, while urban traffic is interrupted. These interruptions are due to traffic lights, stop signs, give way signs etc. Therefore, we need to account for these differences in our research. In order to have a realistic traffic flow in both motorway and urban traffic network, we have considered different mathematical and statistical explanations of the phenomena. In other words, we have adapted a traffic simulation model that includes both uninterrupted and signalized traffic flow. A more mathematical approach and explanation on this issue, and how traffic lights and signs affect the traffic flow was given in Sect. 11 on page 47.

14 Summary and Conclusions

Most of the traffic simulation packages mentioned in this paper have one use, and that is to help engineers simulate a traffic network they are planning to build or planning to add modifications to. Each of the software applications reviewed, when using their full versions, will accomplish

this task for the user and will provide relevant information, statistics, and preview of the traffic system, whether existant or still in planning. However, our interests in the field of computer traffic simulation lie in finding a navigation algorithm or various navigation algorithms that will provide the best flow of traffic in an already existant traffic network or types of networks. Additionally, the navigation algorithms we are currently studying dynamically adapt to the changes within the traffic network, thus providing us with means to investigate what happens in a specific traffic network when changes occur “on the fly”.

A basic model (eg. statistically plausible) gives us a framework in which to insert various smart driver agents and see how they cope under different scenarios. “Smart” agents can be studied using different path finding algorithms or different cost and heuristic functions in order to check which one copes better in what conditions (in low populated areas, in high populated areas, during national hollidays, events such as a parade in downtown, road restrictions, etc.) A basic A* path finding algorithm can be used to study the various cases, implemented with different cost and heuristic functions, we can have: minimal time needed, minimal distance, minimal number of turns, minimally congested roads, any other plausible hybrid function, etc. Gerdelan has done extensive research on agent navigation combining A* and D* path finding with Fuzzy Logic. [21, 22]

We plan to implement a traffic simulation application that includes means to construct a traffic network, simulate a predefined scenario of random vehicles as well as means to define own scenarios, and most importantly send out “smart” agent vehicles and examine their behaviour in this environment. We also plan to investigate scenarios where all of the vehicles are controlled by our “smart” agents and examine the outcomes of these types of scenarios investigated under various road networks. Additionally, we want to have different path finding algorithms and cost functions so that we can examine their impact on the path followed by our “smart” agents, and how this path would differ if another algorithm or cost function was used.

References

- [1] Gantz, D.T., Mekemson, J.R.: Flow profile comparison of a microscopic car-following model and a macroscopic platoon dispersion model for traffic simulation. In: WSC' 90: Proceedings of the 22nd conference on Winter simulation, Piscataway, NJ, USA, IEEE Press (1990) 770–774
- [2] Kühne, R.D., Rödiger, M.B.: Macroscopic simulation model for freeway traffic with jams and stop-start waves. In: WSC '91: Proceedings of the 23rd conference on Winter simulation, Washington, DC, USA, IEEE Computer Society (1991) 762–770
- [3] da Silva, B.C., Junges, R., de Oliveira, D., Bazzan, A.L.C.: Itsumo: an intelligent transportation system for urban mobility. In: AAMAS '06: Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems, New York, NY, USA, ACM (2006) 1471–1472
- [4] Ferber, J.: Multi-Agent Systems - An Introduction to Distributed Artificial Intelligence. Addison-Wesley (1999)
- [5] Paruchuri, P., Pullalarevu, A.R., Karlapalem, K.: Multi agent simulation of unorganized traffic. In: AAMAS '02: Proceedings of the first international joint conference on Autonomous agents and multiagent systems, New York, NY, USA, ACM (2002) 176–183
- [6] Balmer, M., Cetin, N., Nagel, K., Raney, B.: Towards truly agent-based traffic and mobility simulations. In: AAMAS '04: Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems, Washington, DC, USA, IEEE Computer Society (2004) 60–67

- [7] Radecký, M., Gajdoš, P.: Intelligent agents for traffic simulation. In: SpringSim '08: Proceedings of the 2008 Spring simulation multicongress, San Diego, CA, USA, The Society for Computer Simulation, International (2008) 109–115
- [8] Schaefer, L.A., Mackulak, G.T., Cochran, J., Cherilla, J.L.: Application of a general particle system model to movement of pedestrians and vehicles. In: WSC '98: Proceedings of the 30th conference on Winter simulation, Los Alamitos, CA, USA, IEEE Computer Society Press (1998) 1155–1160
- [9] Wittmann, J., Göbel, J., Möller, D., Schroer, B.: Refinement of the virtual intermodal transportation system (vits) and adoption for metropolitan area traffic simulation. In: SCSC: Proceedings of the 2007 summer computer simulation conference, San Diego, CA, USA, Society for Computer Simulation International (2007) 1–5
- [10] Jain, S., McLean, C.R.: A concept prototype for integrated gaming and simulation for incident management. In: WSC '06: Proceedings of the 38th conference on Winter simulation, Winter Simulation Conference (2006) 493–500
- [11] Olstam, J.J., Lundgren, J., Adlers, M., Matstoms, P.: A framework for simulation of surrounding vehicles in driving simulators. *ACM Trans. Model. Comput. Simul.* **18** (2008) 1–24
- [12] Centre for Applied Informatics (ZAIK) and the Institute of Transport Research at the German Aerospace Centre: Simulation of urban mobility - sumo website. Available at <http://sumo.sourceforge.net/> (2009)
- [13] Quadstone Paramics: Quadstone paramics website. Available at <http://www.paramics-online.com/> (2009)
- [14] Treiber, M.: Microsimulation of road traffic applet. Available at <http://www.traffic-simulation.de/> (2009)
- [15] TSS - Traffic Simulation Systems: Aimsun website. Available at <http://www.aimsun.com/site/> (2009)
- [16] Trafficware: Trafficware website. Available at <http://www.trafficware.com/> (2009)
- [17] McTrans Moving Technology: Tsis: Traffic software integrated system website. Available at <http://mctrans.ce.ufl.edu/featured/tsis/> (2009)
- [18] Taylor, M.A.P., Young, W., Bonsall, P.W.: Understanding Traffic Systems. Avebury Technical (1996)
- [19] Mannering, F.L., Washburn, S.S., Kilareski, W.P.: Principles of Highway Engineering and Traffic Analysis. Fourth edn. John Wiley & Sons, Inc. (2009)
- [20] Haberman, R.: Mathematical Models: Mechanical Vibrations, Population Dynamics and Traffic Flow. Prentice-Hall, Inc. (1977)
- [21] Gerdelan, A.: A solution for streamlining intelligent agent-based traffic into 3d simulations and games. Technical Report CSTN-072, Massey University (2009)
- [22] Gerdelan, A.: Driving intelligence: A new architecture and novel hybrid algorithm for next-generation urban traffic simulation. Technical Report CSTN-079, Massey University (2009)