A Modular Traffic Simulation Model for Traffic Planning in Small Urban Scenarios

Dipl.-Ing. Herbert Paulis, Eveline Prochaska BSc

FH Campus Wien, Verein zur Förderung des Fachhochschul-, Entwicklungs- und Forschungszentrums im Süden Wiens, Department Technik, Angewandte Elektronik, Favoritenstraße 226, A-1100 Wien, AUSTRIA

ABSTRACT:
A micro-simulation for small urban traffic models is presented here which serves as a basis for traffic planning in small towns and rural areas. The simulation features a flexible modular road model. This allows for realistic modelling of existing traffic situations. Also crossroads can be simulated realistically with regards to right of way and traffic lights. A graphical visualisation predominantly focuses on a mathematically and physically realistic pattern display.

1 INTRODUCTION

A micro-simulation system for traffic models is developed to serve as a basis for efficient traffic planning in small urban and rural environments. To allow for various experiments with new and analysis of existing traffic situations, a simulation concept with highly adjustable parameters is developed. Own research has shown that existing models for traffic simulation either follow theoretical road situations or represent a fixed model of a specific existing road situation only. Our goal is to realise a tool which is freely configurable to adapt to any existing road model and also easy to operate. As such it shall allow for small urban or rural domains to examine any planned temporary or permanent changes of traffic situations with regards to their impact on traffic in advance.

A defined objective of this project was to create a useful and working basis for further student projects. So the actual version of the simulator has the basic functionalities implemented as described in the next chapter. Students of the various technical degree programmes shall now provide concepts and implementations for the functional extensions and improvements during undergraduate projects and bachelor and master theses.

2 REALISATION

For the realisation a preliminary assessment of various programming language concepts has been performed, primarily focusing on C++11, ADA, and Java. Besides object-orientation properties [8] and the possibility of using efficient libraries, good code performance was emphasised as well as clarity, efficiency, and powerful data structure availability. The decision was against ADA because of a lack of available support libraries. Java was ruled out mostly because of performance considerations and also a lack of simple and efficient hardware connections which might be considered for further interfacing extensions. So the simulation is finally realised in C++11 [3][7][8], extensively using the STL [6] as well as the Boost libraries [4].

The simulation system shall stand out by the following characteristics:

(1) Modular configuration of the simulated road situation to be able to model any existing traffic situation as realistically as possible. This can be modelled freely and shall allow interaction between various traffic generated in stochastic time periods. This modularity generates a set of complex requirements to the data models used in the simulation [3] as optimisation for modularity and optimisation for simulation are competing characteristics.

(2) Realistic simulation of crossroad traffic with regard to right of way situations and traffic lights. Existing crossroad simulations (like Bungartz [1] or Treiber [2]) handle traffic at crossroads only theoretically or extremely simplified (e.g. Four-Phase-Model in [1]). Here the situation is modelled as realistically as possible using Treiber's Intelligent-Driver-Modell [2], adapted to the reality of daily traffic situations using his defined "politeness factor".

(3) Realistic simulation behaviour in connection with the driving behaviour of road users. The data structures used in the model shall ensure that every road user acts as an individual object with independent behaviour parameters (within the limits of constraints). Using the example of passenger cars this means that every vehicle has information concerning its own route and individually pursues this route travelling through the road model, adopting its travel speed according to the actual traffic situation.

For practical usage the concept of a graphical visualisation is pursued. This shall not have gaming aspects but the major focus is centred on a mathematically and physically realistic display pattern. A simple graphic display prototype is realised with Qt. This allows easy creation of platform independent versions of the simulation, as Qt is available for Windows, Mac OS X, and UNIX platforms with almost no source code modifications necessary for the application. Actual versions used are Qt 5.3.1 [5] and Qt Creator 3.2.0.

The simulation system also provides an editor which allows for easy construction of an individual road scenario by providing a set of standard road blocks, cf. **Figure 1**. These can be combined to a scenario (a "map") which is then used to run the simulation on. This works in such a way that the constructed road scenario provides for a set of entry/exit points. According to preset statistical parameters new vehicles will be generated randomly at the entry points, each selecting also a random exit point. For the generated entry/exit points a path through the road scenario is determined with Dijkstra's shortest path algorithm (using library functions from Boost [4]) which acts as the route for the generated vehicle.

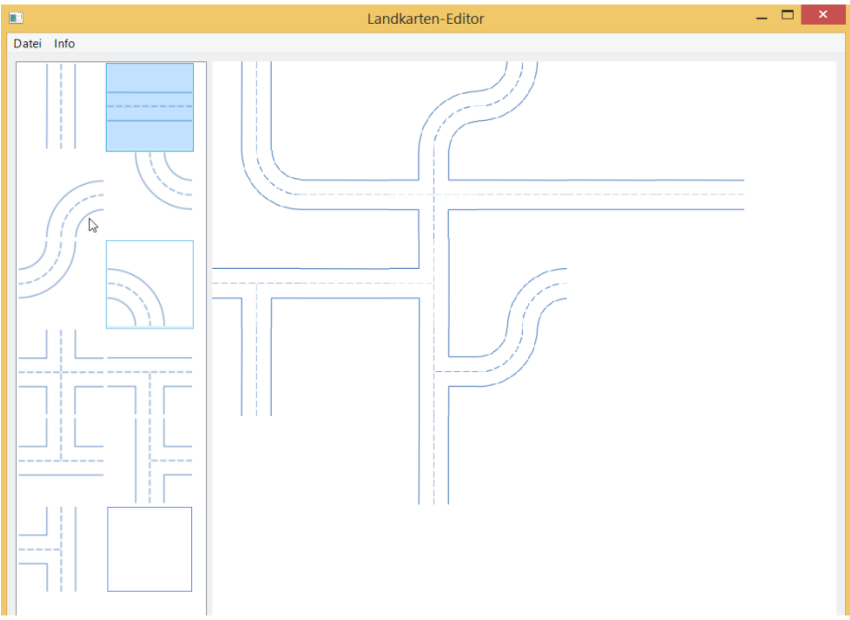

Figure 1. Layout Editor Screenshot

A first simplified prototype has been realised in form of a discrete micro-simulation with stochastic elements using two superposed cellular automata models. This allows among other things to especially observe the following elements:

- Modelling the impact of single vehicles respectively their driving behaviour on traffic
- Simulation of the influence of speed limitations, no-passing limitations for trucks, etc.
- Interpretation of human driving behaviour according to the IDM model by Treiber [2]
- Taking into account the interaction between various forms of traffic like passenger cars, trucks, busses, etc.
- Definition of arterial and ancillary roads. This is realised by different values for the weight of the edges representing the roads in the graph underlying the cellular automata model.

The simulation also includes a complex realisation of traffic lights which are operated after a realistic model via a traffic light operation chart and a central controller instance. This allows for extra turning lanes and complex crossroad traffic light models, also allowing traffic holdup analysis and traffic light synchronisation. Also situations occurring at crossroads without traffic lights are simulated realistically regarding right of way situations. An example with a screenshot of a simulation session can be seen in **Figure 2**.

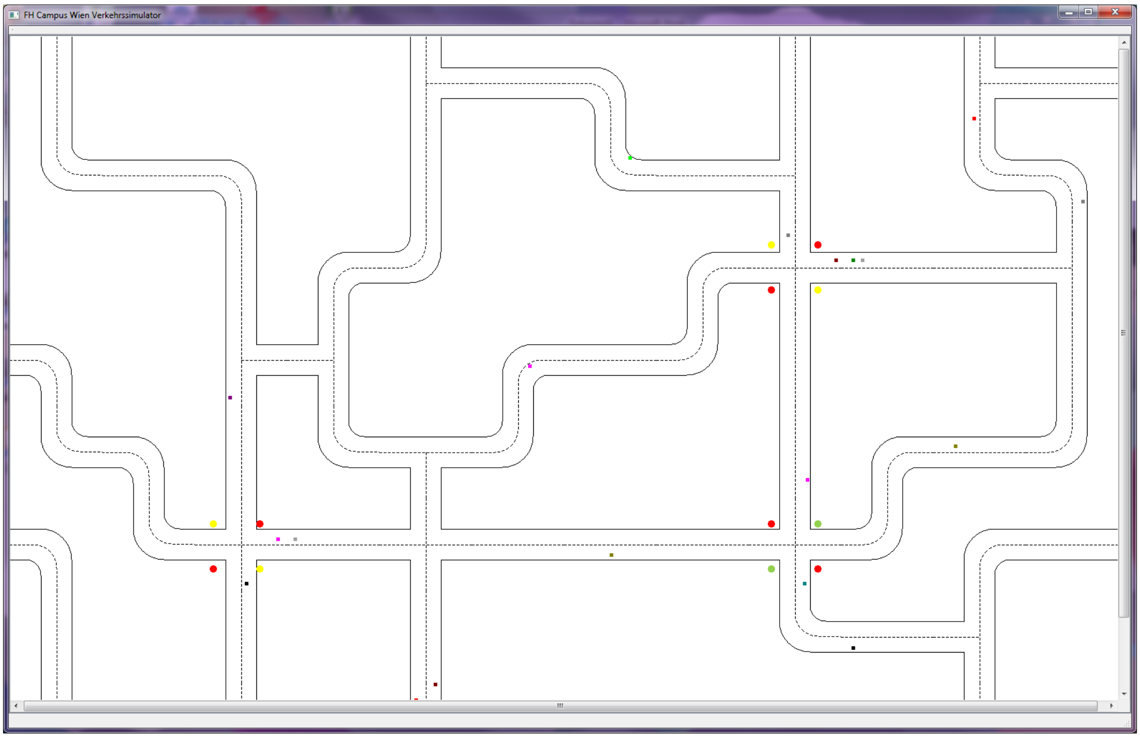

Figure 2. Screenshot of a Simulation Session

For debugging purposes all events are logged to allow tracking of errors and special situations like traffic jams and deadlocks, cf. **Figure 3**. While the simulation tries to detect possible deadlocks and to resolve them in a way similar to actual situations (one randomly selected vehicle waives its right of way), this problem is not yet fully resolved.

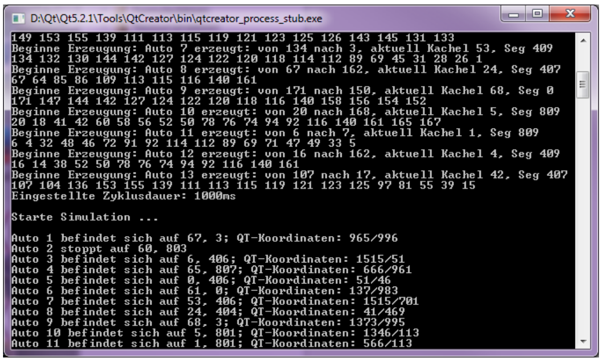

Figure 3. Debugging Window

3 PLANNED ENHANCEMENTS

The planned enhancements will largely be realized in the form of student projects. Students of Information and Communication Technology degree programmes work during various practical projects and their Bachelor theses on extending the functionality and the features as listed:

- Multi-lane roads (all road models are realised as single lanes in either direction only in the prototype, while the data structures already consider multiple lanes)
- Traffic road signs controlling traffic flow (speed limits, driving directions, …)

- An editor for traffic light definitions allowing also synchronised traffic lights to examine how this afflicts flow of traffic in given situations
- Various traffic including outside traffic participants (bus, trucks, underground, tram with and without separate rail tracks, cyclists, pedestrians)
- Environmental aspects like CO_2 foot print and noise emission. Using a microscopic simulation model approach allows to deal with questions like fuel consumption depending on an individual driving style regarding accelerating and breaking characteristics [9].

4 CONCLUSION

This contribution introduced the development of a modular traffic simulation concept. Modularity requirements were considered already at the stages of planning and modelling so that the basic realisation can be used as a platform for improvements and new functionalities. One key aspect was to focus on clear and comprehensible data structures. Considerable efficiency and functional safety was reached by using the C++ STL (Standard Template Library) which provides a lot of well designed and tested data structures and functional algorithms. Container classes with well-defined and guaranteed performance characteristics are provided, something which is of significant importance when dealing with simulation issues.

LITERATURE

[1] Hans-Joachim Bungartz et al., Modellbildung und Simulation, Springer Berlin Heidelberg, 2009
[2] Martin Treiber, Arne Kesting, Verkehrsdynamik und –simulation, Springer Berlin Heidelberg, 2010
[3] Bjarne Stroustrup, C++11 Programming Language; Addison-Wesley, 2013
[4] Boost C++ Libraries, www.boost.org (Stand 20.01.2015)
[5] www.qt-project.org (Stand 20.01.2015)
[6] Standard Template Library Programmer's Guide, http://www.sgi.com/tech/stl/ (Stand 20.01.2015)
[7] Jürgen Wolf, C++ Das umfassende Handbuch, Galileo Press, Bonn 2014, 3. Auflage
[8] Joyce Farrell, OOP using C++, Course Technology, USA, 2009, fourth edition
[9] Martin Treiber, Arne Kesting, Traffic Flow Dynamics, Springer, Berlin, 2013