

# A method for real-time adaptation of weather conditions within a traffic simulation\*

Gábor Pótári<sup>a</sup>, Zoltán Vincellér<sup>a</sup>, Zoltán Ács<sup>b</sup>

<sup>a</sup>Inter-University Centre for Telecommunications and Informatics, Debrecen, Hungary  
Eötvös Loránd University  
potari@caesar.elte.hu, vzoli@inf.elte.hu

<sup>b</sup>Eötvös Loránd University  
acszolta@inf.elte.hu

## Abstract

The increasing amount of trajectory like information enables us to simulate real traffic situation more and more accurately. With the proper use of this large dataset we can create more realistic simulations of local traffic problems; moreover we can run such simulations which can take the environmental changes into account. A lot of commercial and open-source software exists to realize and visualize these mentioned needs. Most of these software packages are specialized to solve a special group of problems, so each has its special field of use. The examined software packages don't take environmental conditions into consideration within a simulation, such as a slippery road in a snowy night. The calibration of a traffic simulation model to different weather conditions is considered to be a difficult task; for example, it is not trivial to determine the impact of each factor on the system. The article below ([1]) discusses the problem of slippery roads caused by snow. Slipperiness can be given an n-dimensional function, which sufficiently approximates the real measurement values. The purpose of my paper is to implement the above-mentioned methodology in a licensed and in an open-source environment. To examine that this method is capable to simulate other weather conditions can be a subject of further researches.

*Keywords:* slippery roads, microscopic simulators, calibration

*MSC:* 60K30, 68U20

---

\*The authors thank the partial support of the project TÁMOP-4.2.2.C-11/1/KONV-2012-0001 supported by the European Union, co-financed by the European Social Fund.

## 1. Introduction

We can easily notice, that weather conditions have several effects on the daily traffic. For example, when it is snowy, then the road surface becomes slippery, and thus vehicle drivers have to take attention on leaving higher following distance rate and lowering their desired speed in order to avoid accidents. We have examined some vehicular traffic simulators to reproduce the effect, which is caused by some weather condition. As result, we selected two microscopic simulators for further calibrations. These two were Vissim and Sumo. While Vissim is a commercial modelling tool, Sumo is an open source, microscopic and continuous road traffic simulation package designed to handle large road networks. In this paper, we will shortly take the advantages and disadvantages of both tools into consideration. After this short overview, the Wiedemann 74 car-following model will be presented used by both applications. Finally, we will tell some details about one of our calibration methods. It is a brute force calibration procedure, which has been implemented in C++ Qt framework, and its efficiency has been compared with the one in [1].

## 2. Microscopic simulators

A microscopic simulator offers several functionalities for us, some of these are the follows: determining roads capacity, analyzing the effect of a new network's element, calculating different traffic specific values like the vehicle emissions or the amount of vehicle in a specific point and so on. Another important field of use of these tools is when companies or organizations, which are responsible for traffic related problems or urban planning, use them for validation and verification of some special models used by them. Depending on how detailed are we examine a certain topology, we can distinguish submicroscopic, microscopic, mesoscopic, and macroscopic simulators. Here we deal only with microscopic and macroscopic models. While the macroscopic models handles the vehicular traffic as a flow in some network, the microscopic models examine each vehicle individually as an independent object. So, in the latter case the simulation resource requirement is growing with the number of simulated vehicles. This is the reason why we use microscopic models for a city sized examination and macroscopic models for freeways and larger areas.

### 2.1. Vissim

Recall, that the name of the first selected traffic model is Vissim which is a microscopic traffic simulation model developed by a German company called PTV Group [6]. We could say that Vissim is the most common and widely used commercial traffic microscopic simulator. Besides the standard traffic network elements, it contains many additional modules and with these modules we can solve different tasks like measuring vehicles emissions, programming traffic lights, routing with dynamic

assignment and so on. Vissim has also another module called COM which can be called a Vissim network by an external programming environment. With it, we are able to implement new methods around the traffic model. Besides its advantages, Vissim has some disadvantages like its resource cost, we cannot change every mathematical models what it contains only a few one, the creation of the network needs a lot of time, and there is a limit of the network size what it can simulate, what is around or even less than the quarter of the eleventh district, and finally Vissim is available only on Windows.

## 2.2. Sumo

The name of the second selected traffic simulator package is Sumo, which is an open source microscopic and continuous road traffic simulator implemented in C++. The network creation is quite simple, we can import an arbitrary network from [openstreetmap.org](http://openstreetmap.org), or Tiger/Line database into Sumo using its netconvert module. With Sumo, we are also able to measure up to country-sized areas because of its robustness due to its portability, its platform independent. The main difference between the two simulators is that Vissim has a graphical interface to create the traffic lights in a specific junction, but in Sumo we have to create these settings directly in the network configuration files, which are usually in XML format.

## 3. Data collection

The Figure 1 shows the selected topology. Recall, goal of this paper is to determine the effect of snowy weather to Bartók Béla way. The topology has been imported from OpenStreetMap. Few modifications have to be done with the map, like add missing traffic lights (so-called tlogic object in the XML).

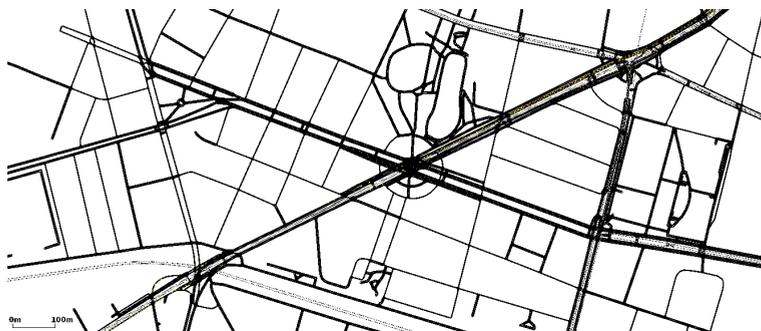


Figure 1: An example for a selected topology in Sumo

The target values that we have to measure in each simulation is the saturation flow rate. Saturation flow is defined as the number of uniform vehicles discharging at an intersection in a dense flow of traffic per hour of effective green time for

a given lane, from this point we abbreviate vehicle per hour green per lane with vphgpl. Saturation flow rate is an indication the potential capacity of a junction. To determine saturation flow rate, saturation headway ( $h_s$ ) has to be calculated as bellow from [5].

$$h_s = \frac{\sum_{j=n}^l h_j}{l + 1 - n},$$

where  $l$  is the last queued vehicle position,  $h_j$  is the headway of  $j$ th queued vehicle in seconds and  $n$  is the position of queued vehicle from where saturation flow region started.

Saturation flow affects only the first few vehicle, where the start-up delay between them is not equal. After it, saturation flow rate ( $s$ ) can be determined by the following equation from [5].

$$s = \frac{3600}{h_s}.$$

Two different intersections have been selected for data measurements as pictures show. The data used in this paper were observed from video recording, each recording were 1 hour long.

According to [3] and [4] the impact of snow on traffic flow is about 20-30%. The analysis of the video recordings was performed manually. The saturation flow rate has been evaluated from these recordings.



Figure 2: Observed intersection (I1)



Figure 3: Observed intersection (I2)

468 vehicles/hour were counted in the first intersection (I1) and 411 at the second intersection (I2). These values provides the input values for OD matrix. Traffic light settings and site plans have been adjusted based on data from BKK. By the way, BKK is an organization in Hungary, which is responsible for traffic tasks in Budapest. At I1 45, while in I2 30 seconds of green time have been observed.

## 4. Wiedemann car-following model

Microscopic traffic models use car-following models to avoid vehicle collisions. Papers claim that Wiedemann is the most suitable car-following model for urban traffic simulations. Paper [7] contains the detailed equation list that it contains. The model has four different statements (Figure 4), depending on the distance and speed difference between a given and the front vehicle: following, emergency, approaching, free flow. In Vissim we can change it before each simulation run from COM interface, and in Sumo we have to parse the network's XML, and change the car-following parameters which is a more time-consuming method.

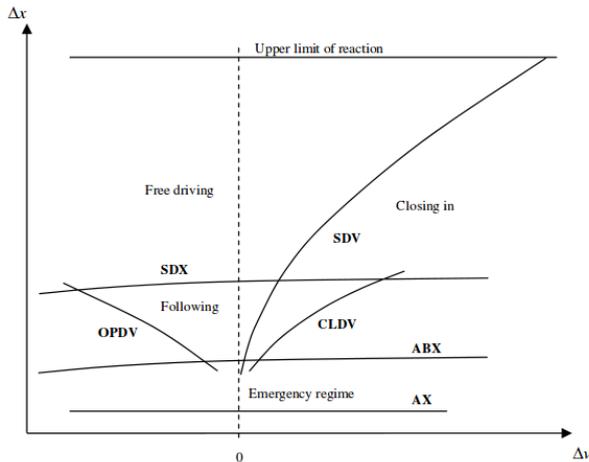


Figure 4: Wiedemann 74 car-following model statements [7]

## 5. Calibration method in Sumo

Now, let's move on to the calibration method. What if the simulation has to be run under different weather condition? We want to reproduce vehicle driver's behavior in different weather conditions with an appropriate settings and we want to find the most suitable. For example drivers in snowy weathers increase the distance from the front vehicle, drive with lower speed and so on. According to [1], to simulate

snowy roads enough to change three parameters. These parameters are the vehicles acceleration, desired speed and the clearance distance, which is the parameter  $bx$  from the Wiedemann model. We have to compare thus ran simulations result with the real observed data. Value of the acceleration can be between 0.1 and 1.0 with 0.05 increments, it means 19 different levels. Desired speed can be the range of 15 and 55 with 5 km/h increments so its 9 different values, and the following distance can be between 0.25 and 6 with increments of 0.25 it means 24 values.

## 5.1. Calibration in Vissim

As mentioned Vissim network objects are easily reachable using com interface. Several useful data from each vehicle can be retrieve, such informations like their speeds, positions, lanes, even the vehicles position in the queue at a traffic light, which is a necessary data for determining saturation flow rate.

Each simulation run takes 30 seconds what is equal to 1800 simulation seconds. We have 4104 different parameter settings, therefore in Vissim the calibration needs about 34 hours.

## 5.2. Calibration in Sumo

The calibration method has been implemented in Qt framework. The task is to determine in above-mentioned intersections the saturation flow rate. To get the necessary data, several additional files have to be process: FCD data from each vehicle, traffic light state changes (so-called SaveTLSStates object), and induction loop measurements (instantInductionLoop object). First, the investigated traffic lights starting time-step of each phase has to be collected. Thereafter, positions of the vehicles in the queues can be retrieve in the given time-steps from the FCD data output. Finally, start-up delay and saturation flow rate can be measured using the placed induction loop detectors at the stop lines as Figure 5 shows.

Each run took 70 seconds to simulate 1800 seconds. Processing the generated outputs took 19 seconds, thus the calibration procedure in Sumo takes approximately 90 hours.

## 6. Conclusion

Summarizing, the referred brute-force calibration method is slower in the open-source sumo, than the commercial used Vissim, although in Sumo we are able to simulate the effect of different weather factor on larger areas. This slowness mainly comes from the output processing. If there would be an output type which can be used to determine each vehicle in a queue, time-cost different between them would be smaller.

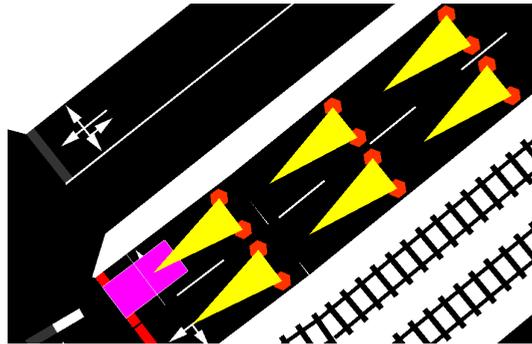


Figure 5: Loop detector measurements in Sumo (purple rectangle)

## References

- [1] Asamer, J., Calibrating VISSIM To Adverse Weather Conditions (2011)
- [2] FOWLER, B., BARTELS, R., Constraint-based curve manipulation, IEEE Comp. Graph. and Appl., Vol. 13 (1993), 43–49.
- [3] Asamer, J. and vanZuylen, H. J. (2011) Saturation Flow Under Adverse Weather Conditions. Proceedings of the 90th Annual TRB Meeting, Washington,.
- [4] Hranac, R., Sterzin, E., Krechmer, D., Rakha, H., Farzaneh, M., Arafah, M. (2006) Empirical Studies on Traffic Flow in Inclement Weather. Federal Highway Administration Report No. FHWA -HOP-07-073, Washington, D.C.
- [5] C. J. Bester, W.L. Meyers (2007) Saturation flow rates
- [6] User Manual for VISSIM 5.10, PTV – Planung Transport Verkehr AG, 2008
- [7] Analysis of the Wiedemann Car Following Model over Different Speeds using