A Driver Behavior Learning Framework for Enhancing Traffic Simulation

Ramona Maria Paven, Mihai Pachia, Dan Pescaru
Faculty of Automation and Computers
Politehnica University of Timisoara
V. Parvan no. 2, 300223 Timisoara, Romania
e-mails: ramona.maria.paven@gmail.com, dan.pescaru@upt.ro

Abstract— **Traffic simulation provides an essential support for developing intelligent transportation systems. It allows affordable validation of such systems using a large variety of scenarios that involves massive data input. However, realistic traffic models are hard to be implemented especially for microscopic traffic simulation. One of the hardest problems in this context is to model the behavior of drivers, due the complexity of human nature. The work presented in this paper proposes a framework for learning driver behavior based on a Hidden Markov Model technique. Moreover, we propose also a practical method to inject this behavior in a traffic model used by the SUMO traffic simulator. To demonstrate the effectiveness of this method we present a case study involving real traffic collected from Timisoara city area.**

Keywords— traffic model, traffic simulation, driving behavior.

I. INTRODUCTION

Traffic simulation and prediction is involved in development of all modern intelligent traffic systems. They represent the optimal way to test and validate such systems in safer conditions, without interfering with, or disturbing the real traffic. However, most traffic models used in modern simulators do not include accurate driver behavior models. In some situations, this can create a significant gap between simulation and traffic reality.

To describe the behavior of people in traffic is not a trivial problem. A realistic model depends on several aspects, not very well explored since now. A first category is related with personality of the driver, which is hard to be captured and modeled at large scale. Other aspects are related with social nature of the traffic [1] and imply coordinate decision and behavior adaptation. Besides of that, persons are also influenced by cultural and socio demographic realities of the geographical area considered. Finally, road configuration and some traffic conditions could also influence driver actions.

Several studies were conducted in the last decades in order to find various parameters that influence driving behavior. As an example, a qualitative method was proposed in [2] to investigate drivers' speed choice. The analysis concludes that speed choices are related to the physical design of the road, interaction with other road users, personal attitudes and reasons related with respect for the law. Driving with passengers in the car, particularly children, influence drivers to choose a lower speed, except for some young drivers who reported speeding when driving friends. Several drivers expressed difficulties in maintaining a speed below the speed limit. The conclusion was that the consequence of speeding, such as increased accident risk, was given little consideration by the study participants.

In context of developing realistic traffic simulation, one of the first aspects regarding driver behavior was the lane-changing action. The SITRAS system presented in [3] describes two lane-changing models: the forced model, and the cooperative lane-changing model. The goal was to produce realistic flow-speed relationships during traffic congested conditions. After that, some other models was proposed, but all of them considers only specific actions and do not propose a general model for the behavior.

The approach proposed here also does not aim to create a complete mathematical model to describe the behavior. Instead of that the model is learned from real traffic recorded from investigated area using a Hidden Markov Model technique.

The rest of the paper is structured as following. Section II gives an overview of some popular traffic simulation systems and draws some conclusions useful in our attempt to extend the models by introducing the traffic behavior. Section III explains the framework proposed by us to support learning of the traffic behavior. Next section presents a way of using the learnt traffic behavior in order to enhance the simulation model in Sumo. Enhanced traffic model is validated in Section V by using a case study for a real segment of road. Last section concludes the work and presents some promising future improvements.

II. TRAFFIC SIMULATION

A. *Traffic Simulation Models*

Traffic simulation is very important in context of the demand of increasing traffic safety and managing growing traffic flow. It allows evaluation and validation of various solutions developed to deal with problems like traffic congestions or traffic patterns discovery [4]. In general, the traffic simulator systems support three categories of traffic models: macroscopic, mesoscopic and microscopic.

Microscopic simulation describes each individual vehicle movement in terms of position, speed, acceleration and action. The most used microscopic models are derived from the car following model [5][6] and the cellular automata model [7].

Macroscopic simulation models analyze traffic flow at a macroscopic level, which has been inspired from the hydrodynamic theory. Instead of describing individual vehicle behavior, the model concentrates on statistical variables that summarize the traffic flown in the simulated area. In this respect the traffic is characterized by global variables as the flow rate, the flow density and the flow average velocity [8].

Mesoscopic simulation models combine the previous two models in various ways. They could decrease the simulation resource consumption by calculating traffic states only when something happens in the network [9].

The work presented in this paper aim to improve microscopic simulation models with real traffic characteristics close linked with non-uniform driving behavior developed by traffic participants.

B. *Traffic Simulators*

Traffic simulators are essential tools in developing intelligent traffic systems. They implement various traffic models and generate realistic vehicles movement data to be used as an input for these models. They manage road models and various scenario parameters as maximum vehicular speed, rates of vehicle arrivals, and rate of vehicle departures. The output has fine granularity and details the location of each vehicle at every time sample for the entire simulation time. Examples of popular simulators are SUMO [10], MOVE++ [11], CityMob [12], FreeSim [13], and Netstream [14].

We choose to extend in our work the traffic model of Simulation of Urban MObility [15]. SUMO is an open source microscopic road traffic simulation package designed to handle complex road networks [16]. It can import various types of maps, as for example complex OpenStreetMap [17], it can handle multiple lanes and traffic signals and it implements various traffic rules. Traffic configuration is based on individual routes for vehicles. A special designed module called Traffic Control Interface (TraCI) allows interaction with external tools and control systems [18].

The most used SUMO traffic demand model uses poison distribution to generate vehicles that are injected into the network simulation. All model information is written in XML format in standard configuration files.

III. A Framework for Learning Driving Behavior

Driving behaviors are complex and they can be consistent within a specific range of population when facing a particular trajectory. In this paper, common driver behavior is captured from several recorded sequences of vehicle movement using Hidden Markov Model [14].

Using only road sensors vehicle movements can be recorded, but it is practically impossible to obtain drivers' disposition while being on a certain trajectory. To derive unobservable driving attitudes using HMMs, vehicles movement data are treated as the observable states.

The proposed methodology has four components: driving behavior learning, driving behavior decoding, driving behavior distribution adjustment, and population generation.

Driving behavior learning component is based on a prior training period performed by drivers with different behaviors on individual road segments. The model aims to associate each sequence of vehicle movement on an individual road segment to a cognitive model. Five different cognitive models of human behavior are considered.

Driving behavior decoding is acquired using Hidden Markov Models (HMMs) to characterize and detect driving maneuvers throughout various trajectories and place them in five different cognitive models of human behavior.

The *distribution adjustment* component aims to find and adjust the frequency of the five cognitive models in order to obtain a distribution closer to the real model.

This distribution will be used in the generation of the virtual population of drivers by the *population generation* component.

The first step is to build individual models for each trajectory segment type. For that we choose the k-means algorithm, known also as the Lloyd algorithm [19], in order to optimally estimate model parameters for each model. We aim to obtain five clusters for each trajectory segment type, corresponding to five driving behavior types.

The second step is to describe driving behavior as a succession of basic actions each defined by a set of observable parameters and particular states of the driver-vehicle environment. The observable states of the driver-vehicle environment are the effects of the driver's internal state. The driver behavior recognition model prototype proposed here aims to subclass driving behavior in five different classes: B1, B2, B3, B4 and B5, corresponding to the five clusters.

The frequency of a given behavior during the simulation may be relevant. Adjusting the frequency we can design the population behavior distribution that is more appropriate to a given problem. We aim to distribute the obtained driving behavior classes in order to obtain a virtual population of drivers having a behavior closer to real traffic.

A. *Driving behavior learning procedure*

First, we train our model on the three different segment types considered: plain segment, sag curve, and crest curve. The training consists in a number of repetitions on this trajectory segments. It has to be performed by drivers having different driving behaviors.

Second, we apply Lloyd algorithm to cluster the training data into clusters for each segment type. The Lloyd algorithm is an unsupervised learning algorithm for clustering analysis. To map the Lloyd algorithm to driver behavior, the data point definition needs to be adapted.

The initial means number is the number of clusters generated by the algorithm. In our approach we define five means, corresponding to the five clusters we want to obtain. Each cluster corresponds to a different driving behavior type.

We define a data point $p=(p_1, p_2, \dots p_n)$ as a sequence of actions taken by the driver on a particular trajectory segment. In our approach the coordinates p_i corresponds to the driver actions. These actions are characterized by a steering angle $p_i.st$, and a velocity value $p_i.v$.

The distance between two data points, $p=(p_1, p_2, \dots p_n)$ and $q=(q_1, q_2, \dots q_n)$, computed as the Euclidean distance between two points in an n-dimensional space is adapted as follows:

$$d(p,q)=\left[\left(\frac{p_1.v}{p_1.st_a}-\frac{q_1.v}{q_1.st_a}\right)^2+\left(\frac{p_2.v}{p_2.st_a}-\frac{q_2.v}{q_2.st_a}\right)^2+\cdots + \left(\frac{p_i.v}{p_i.st_a}-\frac{q_i.v}{q_i.st_a}\right)^2+\cdots+\left(\frac{p_n.v}{p_n.st_a}-\frac{q_n.v}{q_n.st_a}\right)^2\right]^{1/2} \quad (1)$$

The obtained clusters corresponding to different driving behaviors for each segment type are displayed in Table 1.

TABLE I. HIDDEN STATES TYPES

Behavior types	**Segment types**		
	Plain segment	***Sag curve***	***Crest curve***
B1	β_1	μ_1	τ_1
B2	β_2	μ_2	τ_2
B3	β_3	μ_3	τ_3
B4	β_4	μ_4	τ_4
B5	β_5	μ_5	τ_5

B. Driving behavior decoding

The driving recognition system, based on HMM, is comprised of two stages.

The first, an offline stage, is the training phase. We define a different HMM structure for each segment type. It would be irrational to have the same HMM structure, as the number of identifiable actions for each of the three segment types are different.

The HMM representations for the sag curved segment type is presented in Figure 1. The other HMM structures corresponding to crest curve segment and to plain segment differ as number of states recorded to capture the behavior.

The second is the recognition phase, as we are interested in recognizing the model, which most probably gave rise to the observed sequence. We train the model using more complex trajectory sequences made of the three segment types considered: plain segment, sag curve and crest curve segment. The aim is to partition each of the training trajectory sequences into states.

We consider a HMM with state space $S = \{s_1, s_2, \dots, s_n\}$, where the states take values from the driving behavior types identified by the Lloyd's algorithm.

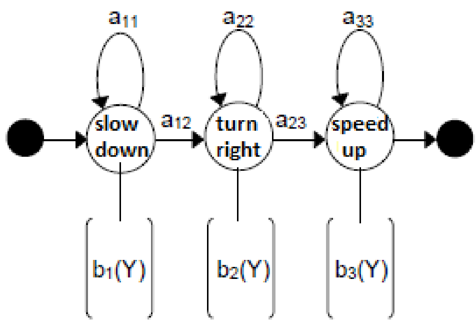

Fig. 1. HMM structure for sag curve segment.

An initial set of parameter values is obtained using the Viterbi Training algorithm [13] over the training data. We adjust the initial probabilities and the transition probabilities definitions. The initial probabilities are π_i – the probability of a certain $y.v$ being in state S_i and ψ_i – the probability of a certain $y.st$ being in state S_i.

The transition probability $a_{i,j}$ is the probability of transitioning from state s_i to state s_j. Then we use the Viterbi algorithm [8] to find the most likely sequence of hidden states related to a given set of observations.

For a sequence $Y=\{(y_1.v, y_1.st), (y_2.v, y_2.st), \dots, (y_T.v, y_T.st)\}$ of observed outputs during vehicle movement, the most likely state sequence $X=\{x_1, x_2 \dots, x_T\}$ that produces the observations is defined by the following recurrence relations:

$$V_{1,K}=P(y_1.v \mid k)\cdot \pi_k + P(y_1.st \mid k)\cdot \psi_k \quad (2)$$

$$V_{t,k}= P(y_t.v \mid k)\cdot \max_{x\in S} (a_{x,k}\cdot V_{t-1,x}) + P(y_t.st \mid k)\cdot \max_{x\in S} (a_{x,k}\cdot V_{t-1,x}). \quad (3)$$

In these relations $V_{t,k}$ is the probability of the most probable state sequence responsible for the first t observations that have k as its final state.

The most likely sequence of hidden states is found by saving back pointers that remember the x state used in the (3) equation.

C. Driving Behavior Distribution Adjustment

After obtaining the most likely sequence of hidden states for the training trajectories, we aim to extract the distribution of the driving behavior types.

The distribution is calculated separately for each trajectory, considering the corresponding behavior type for each hidden segment type. Each behavior type will be assigned a distribution value.

Moreover, our objective is to analyze the frequency of each behavior type and adjust this frequency in order to be more appropriate to the real system. Based on this appropriate frequency, we obtain a new distribution named real distribution.

D. Population generation

Our main purpose is to create a virtual population of drivers where the global behavior of the population resembles a set of driving behavior types. The occurred frequency of a given behavior helps in designing a behavior distribution in the population that is more appropriate to a given issue.

This will be implemented by taking a set of drivers from the simulated traffic system in order to assess the assigned distribution. The selection of the drivers that will be modified in the population is carried out by the evaluation of driving behavior distribution, and a driver can remain in the new population if a certain threshold of the real distribution is reached, else the driver behavior is modified.

IV. Generating DB Enhanced Traffic in SUMO

The proposed software architecture consists of the SUMO simulator, a driving behavior component and a broker between the simulation engine and the driver behavior component, as presented in Figure 3.

The SUMO [7] simulator is microscopic, explicitly defining each vehicle by at least an identifier, the departure time, and the vehicle's route through the network. The mobility model in SUMO is based on the car following model proposed by Krauss [17]. The simulator allows departure/arrival properties to be defined, for example the acceleration and deceleration properties, maximum speed, the minimum longitudinal and lateral clearances can be specified.

To couple the road traffic to the behavior component we use TraCI [9], which acts as a broker between the two components. It connects the two in real time thus enabling the control of mobility attributes of each simulated vehicle. TraCI uses TCP based client/server architecture to provide access to the SUMO traffic simulator. Thereby, the traffic simulator acts as server and the behavior component acts as a client. Once the TCP connection is established, the behavior component controls the traffic simulator via the data exchange protocol, which enables movement changes for each simulated vehicle, making it is possible to instantaneously adjust the movement of individual vehicles.

The behavior component perceives the simulation through the broker and implements the proposed framework for learning driving behavior, making the necessary updates into the simulation.

First, in the simulation environment are trained various vehicles having different behaviors on different segment types. The Driving Behavior Component keeps track of these vehicles by sending periodically a query to the traffic simulator regarding the velocity and position of the vehicle and receives a response that is added to the collected amount of data. When data collection is concluded, a clustering task is performed on this data, and different behavior types are identified for each segment type.

Second, the driving behavior component manages a set of vehicles on complex trajectories from the simulation environment. Using the behavior types identified, it decodes the information received in order to find the distribution of the behavior types identified earlier.

In the next phase, the driving behavior component proceeds to generate a virtual population of drivers that respects the distribution value of the different behavior types. In order to respect this distribution value, only a portion of the population of drivers shall be modified during simulation. The Driving Behavior Component sends periodically for these drivers, every simulation step a command to the traffic simulator that contains the actual simulation time plus one simulation step. The simulator performs the next simulation step and the resulting vehicle positions are sent back to the Driving Behavior Component to be processed.

This approach can affect a bit the performance of the simulation, since the Driving Behavior Component must wait the simulation environment responses before requesting to continue to the next step, but represents a flexible solution that allows SUMO to be used without major improvements.

The population aims to reflect the proportion of drivers, which have different driving behavior types, and hence do not respect the car following model. The population generation is actually a simulation phase where this proportion of drivers that do not respect car following model will be modified through the broker and the assigned updated values will be committed into simulation.

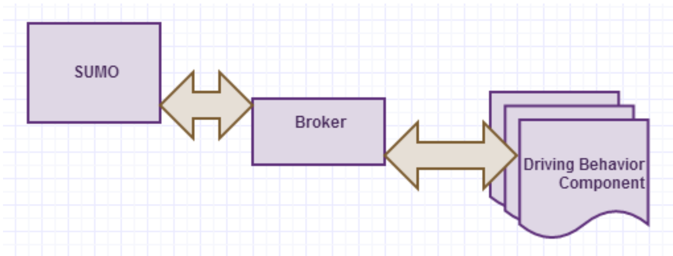

Fig. 2. Architecture overview.

V. Case Study for Model Validation

First, we obtain our hidden state types as a result of a number of training repetitions on each segment type as plain segment, crest curve and sag curve. The segments used for training are presented in Figure 3.

Second, we train our model on a chosen trajectory. The trajectory represents a street portion from Timisoara, located at 21.26 degrees latitude and 45.72 longitudes. The traffic flow is regulated and uninterrupted on this trajectory.

In order to use this area in SUMO the corresponding OpenStreetMap file has to be converted from a map to a SUMO network file. The conversion extracts the information related to the simulation from the OpenStreetMap file and puts it out in the SUMO network file. The network file is imported in SUMO as presented in Figure 5.

We perform a number of training repetitions on this trajectory where we obtain a sequence of states for each repetition. In the Table II, we have the states corresponding to the training performed.

a.

b.

c.

Fig. 3. Example of segment types: a. sag curve, b. crest curve, c. plain segment [25]

Fig. 4. View of the entire case study region [25]

Next we calculate the distribution for each behavior type. The obtained distribution is 20,5 percents for B1 behavior type, 23 percents for B2 behavior type, 9,2 percents for B3 behavior type, 35,3 percents for B4 behavior type, and 12 percents for B5 behavior type. The most frequent behavior obtained is B4, corresponding to a stressed driver behavior.

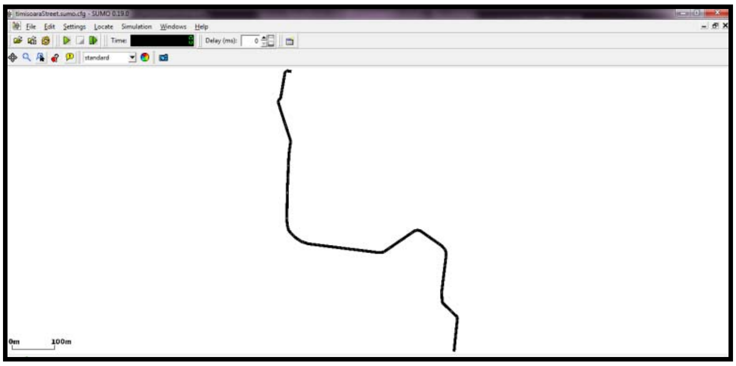

Fig. 5. Map representation in SUMO.

Based on the observed frequencies, we can now create a virtual population of drivers having an increased distribution of the B4 behavior type.

TABLE II. STATES SEQUENCE FOR THE TRAINING PHASE

No	States
1	$\beta_1\ \tau_1\ \beta_2\ \mu_3\ \beta_1\ \tau_4\ \beta_4\ \tau_2\beta_5\ \mu_5\beta_1\ \mu_2\beta_4\tau_5\beta_1\ \mu_3\beta_1$
2	$\beta_4\ \tau_2\ \beta_4\ \mu_5\ \beta_1\ \tau_1\ \beta_2\ \tau_4\beta_3\ \mu_4\beta_4\ \mu_2\beta_2\tau_1\beta_4\ \mu_5\beta_1$
3	$\beta_4\ \tau_4\ \beta_5\ \mu_2\ \beta_2\ \tau_2\ \beta_2\ \tau_2\beta_1\ \mu_4\beta_4\ \mu_2\beta_4\tau_1\beta_2\ \mu_3\beta_4$
4	$\beta_2\ \tau_4\ \beta_3\ \mu_4\ \beta_5\ \tau_4\ \beta_3\ \tau_4\beta_1\ \mu_4\beta_5\ \mu_4\beta_4\tau_5\beta_1\ \mu_4\beta_4$

We compare the proposed framework with the SUMO simulation and with the real time traffic behavior for the proposed trajectory. We monitor the real traffic for thirty minutes, counting the number of vehicles that enter the trajectory and the number of vehicles that leave the trajectory each minute. The chosen trajectory is not a crowded road, excepting the peak time. Therefore, the experiment is registered the flow on a relevant peak time interval.

We used the same input traffic as the one registered from the chosen trajectory during the selected time interval to perform a SUMO simulation. Next, we repeated the experiment considering the same registered traffic as input, but performing an enhanced SUMO simulation adding the behavior computed using proposed framework.

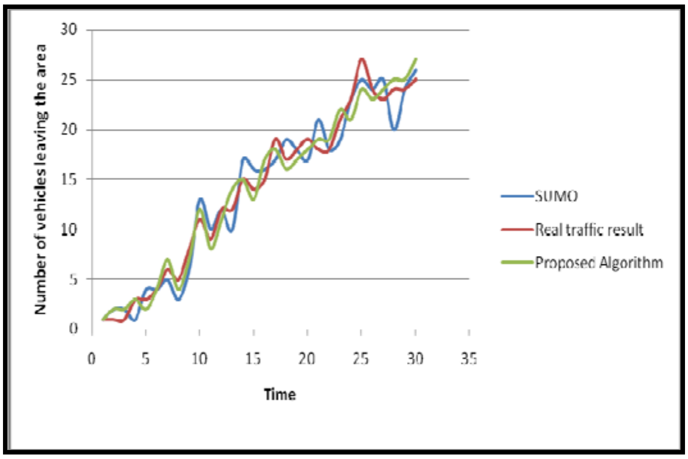

Fig. 6. Experimental results.

The results of the two experiments are presented in the Figure 6. They demonstrate the accuracy of simulation achieved when drivers' behavior is considered. On the chosen trajectory, the mean squared error of output traffic for the proposed solution is 1,53. This mean squared error is lower than the value of 2,55 obtained for the classic SUMO simulation on the same trajectory. Moreover, we can assume even a larger difference in case of a longer trajectory and for heavy traffic conditions. Overall, the results indicate that the proposed framework was capable of simulating the flow in a more realistic manner, closer to the real traffic than in case of using the exiting SUMO simulation model.

VI. CONCLUSIONS

This paper proposes a new method for enhancing traffic simulation models by adding driver behavior. The goal is to provide accurate simulation environments for developing complex intelligent transportation systems.

The method is based on two steps. First step involves capturing the driver behavior from several recorded sequences of vehicle movement using Hidden Markov Model. This information is then used by the second step to inject this behavior into an existing traffic model.

To validate the work we present a case study involving a real segment of a road from Timisoara city area. The experimental results demonstrate the improvement of the simulation model comparing with the real traffic recorded from considered area.

REFERENCES

[1] A. L. Bazzan, J. Wahle, and F. Klügl. "Agents in traffic modelling—from reactive to social behaviour." KI-99: Advances in Artificial Intelligence. Springer Berlin Heidelberg, 1999, pp. 303-306.

[2] G. Berge and T. Vaa, "Feeling of speed: A qualitative study of reasons for speed choices", Institute of Transport Economics, Norwegian centre of tTransport Research, Oslo, Norway, ISBN: 82-480-0359-0, Rep. nr. 660/2003.

[3] P. Hidas, "Modelling lane changing and merging in microscopic traffic simulation", Transportation Research Part C: Emerging Technologies, Vol. 10, Iss. 5–6, December 2002, pp. 351-371.

[4] M. Treiber, A. Hennecke, and D. Helbing, "Congested traffic states in empirical observations and microscopic simulations", Physical Review E, vol. 62, no. 2, pp. 1805–1824, 2000.

[5] J.M. Del Castillo, "A car following model based on the Lighthill and Whitham theory", in the Proceedings of ISTTT'96, Lyon, France, 1996, pp. 103–118.

[6] D. Helbing, "Traffic and related self-driven many-particle systems", Reviews of Modern Physics, vol. 73, no. 4, pp. 1067–1141, 2001.

[7] H. Fuks, N. Boccara, "Generalized deterministic traffic rules", IJMPC Vol. 9(1), 1998, pp. 1–12.

[8] D. C. Gazis, "Traffic theory", Kluwer, 2002.

[9] W. Burghout, H. N. Koutsopoulos and I. Andreasson, A discrete-event mesoscopic traffic simulation model for hybrid traffic simulation, Proc. IEEE Intelligent Transportation Systems Conference, Toronto, 2006.

[10] D. Krajzewicz, "Traffic Simulation with SUMO – Simulation of Urban Mobility", *Fundamentals of Traffic Simulation*, vol. 145, Springer New York, pp. 269-293, 2010.

[11] MOVE (MObility model generator for VEhicular networks): Rapid Generation of Realistic Simulation for VANET, 2007.Available at: http://lens1.csie.ncku.edu.tw/MOVE/index.htm, 2007.

[12] F. J. Martinez, J. C. Cano, C. T. Calafate, P. Manzoni: "Citymob: a mobility model pattern generator for VANETs". In IEEE Vehicular Networks and Applications Workshop (Vehi-Mobi, held with ICC), Beijing, China, May 2008.

[13] J. Miller, E. Horowitz , "FreeSim - a free real-time freeway traffic simulator", IEEE Intelligent Transportation Systems Conference, ITSC 2007, pp.18-23, September 2007.

[14] E. Teramoto, M. Baba, H. Mori, Y.Asano, H. Morita, "NETSTREAM: traffic simulator for evaluating traffic information systems," IEEE Conference on Intelligent Transportation System ITSC '97, pp. 484-489, 9-12 November 1997.

[15] M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz, "SUMO - Simulation of Urban MObility: An Overview," in SIMUL 2011, The Third International Conference on Advances in System Simulation, 2011.

[16] D. Krajzewicz, J. Erdmann, M. Behrisch, and L. Bieker. "Recent Development and Applications of SUMO - Simulation of Urban MObility." International Journal On Advances in Systems and Measurements, 5 (3&4):128-138, December 2012.

[17] OpenStreetMap Project, http://www.openstreetmap.org/, 2014.

[18] A. Wegener, M. Piórkowski, M. Raya, H. Hellbrück, S. Fischer, and J.-P. Hubaux. "TraCI: an interface for coupling road traffic and network simulators". In Proceedings of the 11th communications and networking simulation symposium (CNS '08),2008.

[19] S.P. Lloyd, "Least squares quantization in PCM", IEEE Transactions on Information Theory, Vol. 28(2), 1982, pp. 129–137.

[20] N. Kuge, T. Yamamura, O. Shimoyama, A. Liu, "A driver behavior recognition method based on a driver model framework", SAE Transactions, Vol. 109(6), 2000, pp. 469-476.

[21] A.J. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm", IEEE Transactions on Information Theory, Vol. 13(2), April 1967, pp. 260–269.

[22] J. Lember, A. Koloydenko, "The adjusted Viterbi training for hidden Markov models"

[23] R. R. Lawrence, "A tutorial on Hidden Markov Models and selected applications in speech recognition", 1989.

[24] S. Krauß. Microscopic Modeling of Traffic Flow:Investigation of Collision Free Vehicle Dynamics. PhD thesis, Universitõt zu Köln, April 1998.

[25] Google Maps, maps.google.com, accessed Feb. 2014.