Realizing an effective and flexible ITS evaluation strategy through modular and multi-scaled traffic simulation

Mathias Baur, Matthew Fullerton, Fritz Busch

Abstract—The next generation of transportation systems employing advanced ITS and vehicular inter-networking are difficult to evaluate because of their inherent complexity. Advanced control algorithms, sensor setups, multiple communication media, and the varying driver reactions require new planning and evaluation strategies. While simulation tools are often used to study the likely effects of such systems in advance, we do not fully know what to expect as the outcome of a given implementation, making it difficult to validate a simulation's output. Data collection from a variety of field-tests is beginning to provide this data, but simulation definition, implementation and validation tools are all in need of improvement in order to keep up. Here we propose a new simulation framework that can translate a single ITS use case to different simulation tools with multiple traffic networks and/or data sets. This allows simulation tools to be compared and combined as appropriate for the given use case in order to improve both the validity and scalability of the simulation, and to facilitate better cooperation between those working in this rapidly developing field.

Index Terms—ITS, traffic simulation, simulation framework, vtSim, multi-scaling, simulation evaluation, validation, communication modeling.

◆

COPYRIGHT NOTICE

1 INTRODUCTION

THE central problem facing new Intelligent Transportation Systems (ITS) stakeholders is an uncertainty of the likely effects of such systems on a large scale. Small-scale driver warning and assistance systems (with and without networking technologies) have enjoyed far more rapid deployment partly because they can be easily tested with a single equipped vehicle amongst unequipped traffic. When it comes to networking large numbers of vehicles with a specific traffic safety or efficiency goal in mind, a portion of the traffic network with its population of vehicles has to be considered. Hence the exploration of these systems is proceeding with both field tests (e.g. EuroFOT, COOPERS, simTD) and traffic simulations.

Transportation science considers field tests unable to examine truly large scale scenarios (i.e. anything above a few percent of vehicles equipped with a cooperative system under high traffic volumes) while policy makers (i.e. field test funders) deem simulation in this area to be largely unvalidated because there is no suitable field test results available for comparison. This is different from conventional traffic simulation of existing one-to-many ITS applications, where the communication method, exact

---

- *M. Baur, M. Fullerton, F. Busch are with the Department of Traffic Control and Engineering, Technische Universität München, Munich, Germany.*
E-mail:mathias.baur@vt.bv.tum.de,
matthew.fullerton@vt.bv.tum.de, fritz.busch@vt.bv.tum.de

locations of the distributed data (e.g. variable message signs) and effects on the traffic flow can be easily established.

To help overcome this conflict, we propose a new simulation framework, that can translate diverse ITS data and applications to different simulation tools and handle the resulting diverse types of output. This not only solves the practical issues of combining diverse data and tools, but also allows them to be compared and combined as appropriate for the given research question.

The motivation behind the framework is explained in Section 2. Previous and current work in the same area is briefly reviewed in Section 3 before laying out a set of requirements in Section 4. The framework is then described in detail in Section 5. A simple feasibility study using a skeleton implementation of the framework has been performed and the results are presented in Section 6. Finally, an outlook for future work is laid out in Section 7.

2 MOTIVATION

Modern ITS applications employ both new driver information and communication between vehicles and the infrastructure or other vehicles. For valid simulation-based testing, we require the ability to simulate each component of the system easily and accurately and also to be able to integrate smaller-scale real-world data. This comes not only from infrastructure/vehicle trials but also from tests with drivers to examine how the new systems' interventions are acted upon.

The individual components necessary to meet these goals have for the most part already been developed. What is lacking is a method whereby they can be combined in a flexible way to examine the proposed system at hand. As the scope of traffic simulation grows, such an approach could also facilitate the simulation of some elements in detail and others in approximation through the use of multiple simulation tools in parallel or in sequence (termed *multi-scaling* in this paper). Hence, tools for the combined evaluation of diverse results types must also be included.

Such a framework also holds some advantages of convenience and efficiency. It allows software once seen as 'bespoke' to be used again in new situations. Although multiple implementations of certain tools are desirable to aid validation, continuous reinventing of the wheel is clearly not. Furthermore, any attempt to improve the interoperability of different data and network definition formats would likely be welcomed by the traffic simulation community. By suitable abstraction of the integrated tools, the 'glue' of the framework can be published as open-source code for the simulation community while still allowing the use of closed-source, commercial tools. This allows a rich array of simulation tools to be compared and combined as appropriate for the given use case in order to improve both the validity and scalability of the simulation, and cooperation between those working in this rapidly developing field to be improved.

3 PREVIOUS AND ONGOING WORK

ITS applications have been one of the main driving forces behind the increased modularization of traffic simulation. Through the provision of modifiable software components (through scripting or recompilation) and application programming interfaces (API, e.g. COM on the Windows platform), engineers have been able to test various novel ideas (e.g. automated platoon management[3]). However, having to write new code for a single API severely limits general reusability.

In particular, much work has been carried out in the area of attaching communication simulations to existing traffic simulations. There are very mature tools available in both fields, and so a specialized coupling between the two is often seen as the most expedient way to test modern vehicle communication scenarios. One ongoing approach is the *iTETRIS* Project (*http://www.ict-itetris.eu/*), which aims to standardize the simulation of communication in traffic networks through the European Telecommunications Standards Institute (ETSI). The project, funded by the European Research Framework Programme 7 (FP7), aims to tightly integrate the traffic simulation SUMO[13] and the network simulation ns-3[1], in a replicable and standardized way. The project is based upon the ETSI standard for Intelligent Transportation Systems in terms of what aspects need to be simulated[10]. A more flexible approach has been provided by the V2X Simulation Runtime Infrastructure (VSimRTI)[15]. This uses existing interfaces to run parallel simulations with multiple traffic simulation tools and a single communications simulation[2], [16]. Although existing interfaces are used, they are wrapped in such a way that a developer can easily add another tool by writing a similar wrapper. The framework also reduces computation times by using a more spatially detailed simulation tool only where required (i.e. complex intersections with rotaries or traffic lights), while utilizing the same map and communication model throughout. The runtime infrastructure is used to synchronize the simulation tools. Another example of such multi-scaling is the traffic simulation tool AIMSUN (TSS, Barcelona) which allows micro, meso and macro elements to be connected and run in parallel. However, AIMSUN is a closed implementation, making the connection of other external elements at the various scales difficult. Rather than run parallel simulations, the goal of our work is to provide diverse data sets and communication models to different simulation engines, without trying to maintain further control over the simulations themselves.

4 REQUIREMENTS

As described above, there is a need for an extensible, reusable, flexible and efficient simulation framework for the testing of ITS applications. The framework must provide:

- Use of a common network definition format
- Use of a common scenario definition format
- Separation and abstraction of the application logic
- Necessary merging of the above data as appropriate for each tool employed
- Separation and abstraction of communication model from core simulation tool
- Use of a common data collection format
- Provision of tool-specific-to-generic translation layers (input and output)
- Output of a common results format with respect to the original input data

Concerning file formats, it is desirable to use XML-based format specifications, as they can be efficiently read, exchanged, and their format defined in a standardized way using XML schema (XSD).

As noted in [15], if the simulations are to be used in parallel and feed vehicles to and from one another, additional requirements on synchronization would be necessary. Particularly important is the requirement of temporal synchronization (i.e., that no vehicle can be in two simulations at once)[15]. However, for certain scenarios (such as an on-ramp with an in-flow and out-flow) this requirement could be relaxed allowing simulations to be run in sequence, with the framework tracking outputs and feeding them to the next more or less-detailed simulation. As one simulation's output is another's input, we require a common approach for input and output data. Such an approach can also ease comparisons necessary for calibration and validation of the simulation (Section 5.3.2). By taking this common approach to input and output data, calibration of the traffic simulation is also eased.

As mentioned in Section 3, the separation of the communication model from the traffic simulation has long been a popular choice. The communication in the scope of cooperative ITS systems is not something (yet) that is necessary for the majority of traffic simulations, hence it makes sense to include it only optionally. Most existing work has taken the approach of simulating the communication in the same detail as the vehicles (i.e. car to car packet movements), e.g. [9], [6], [14]. This makes the added complexity highly computationally expensive, making its optional inclusion even more important. This is coupled with a need at this stage of still-emerging technologies and standards to simulate many different communication media[10].

In terms of testing new ITS applications, new data is becoming available on the effect of information on the driver response. As a simple example, the 'green light optimal speed advisory'[4] ought to make a drastic change in the driver's normal braking and acceleration profile (or more directly, a change in speed). Hence, the framework has to preserve traditional, validated models of driving while being able to make changes to the driving parameters (such as desired speed) that give simulation results that are qualitatively similar to those observed in driver testing.

5 FRAMEWORK DESCRIPTION

The approach described here focuses on the evaluation of ITS applications regarding traffic efficiency and safety. The goal is the calibration and validation of diverse behavioral, traffic flow and communication models as well as highly efficient, flexible and valid simulation of their application. The framework couples modules in such a way so that no unnecessary overhead is generated while still preserving flexibility. In the following, it will be termed *vtSim* (**V**alidating environment for **T**raffic **Sim**ulation). The framework enables multi-scaling - the combination of appropriate tools at different modeling scales so that large-scale scenarios with a small number of small-scale changes can be examined (for example, the merging behavior at an on-ramp being highly relevant, but the traffic movements on the main carriageway being of less interest).

The software language for those components of the framework that need to be newly developed is Java due to its flexibility, understandability and frequency of use in the embedded technology market. A block diagram showing the framework's major components is shown in Figure 1, and the individual components are described in the paragraphs below.

5.1 Traffic, map and infrastructure data

The first step in preparing an application simulation is the provision of data for input traffic flows and for the purposes of driver model calibration and validation (see Section 5.3.2).

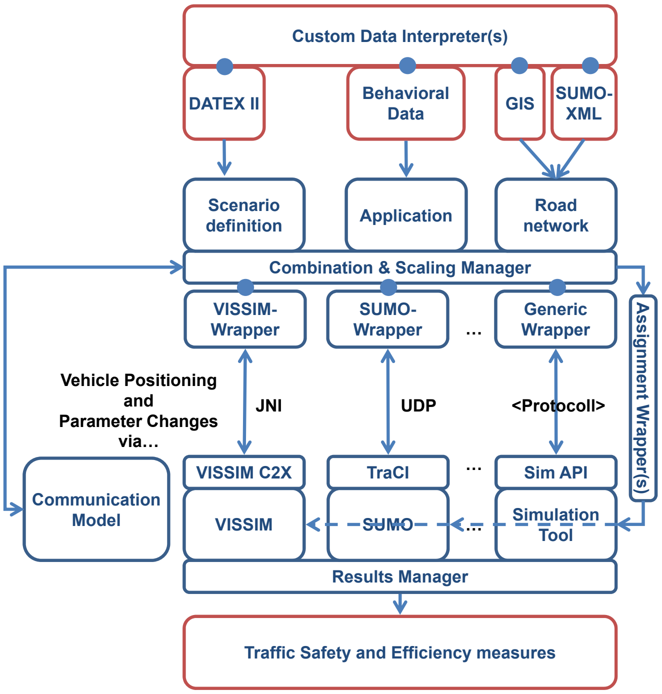


Fig. 1. The framework *vtSim* unites several core components containing simulation models and tools as well as functionality for wrapping and processing different simulation data and application/control algorithms.

Internally, data sets relevant to one simulation basis are established. These may contain multiple forms of traffic data, such as detector data or locations of lane changes (e.g. from overhead trajectory recording). Each of these different types is extended from a common base class to facilitate plotting and data exchange. A good basis format for this data is the DATEX II standard[5] for exchanging traffic data between road authorities, but additional data reading classes can be plugged in so long as they supply data in the common internal format. Data suitable for deriving routing and traffic inputs can also be exported in the SUMO DFROUTER format[5] (which are used internally by the framework, see below).

The same philosophy has been applied to road network and infrastructure data. Here, the internal format is a Java representation of nodes, edge and connections based on the SUMO-XML-Schema. Again, foreign formats are transformed to this common representation. One is the OpenStreetMap[11] database/file format. Hence, all important objects and information which need to be mapped from the real world into a simulation road network can be provided, such as:

- Roads and intersection represented as nodes and edges with exact positioning and identification and further information such as
  - Road type (e.g. highway or urban street)
  - Number of lanes
- Traffic light signals, mapped on roads, lanes and signal phases
- Priority rules
- Detectors and measurement infrastructure
- Traffic flow data

5.2 Combination and Scaling Manager

Ensuring that the right data is provided to the right modules is the responsibility of the combination and scaling manager. Based on the scenario definition, the correct tools are selected (according to, for example, the scale of detail of the simulation required). On the basis of the supplied data, routes are generated for the vehicles in the simulation (this can be accomplished, for example, by calling the SUMO DFROUTER tool). The complete input of vehicle inputs and routes for the simulation tools is then translated from the final generic data by specific assignment wrappers.

The manager is also responsible for assigning starting characteristics for the simulation such as vehicle characteristics or model selection. A distinction is made between this, model and parameter-set assignment process, and the detailed simulation-application interaction that is facilitated by other dedicated wrappers. The former is a one-time operation that leaves the simulation tool free to proceed. The latter is a constant interaction of information provision to the application and returned results of how individual driver-vehicle unit behaviors should be altered (see below).

The manager also configures what data should be recorded from the simulation(s) to facilitate the calibration, validation and multi-scaling processes. This includes the initial setup of the various wrappers so that interaction with the application can proceed.

5.3 ITS Application and altering of driving behavior

5.3.1 Application

The application is expected to be different for every project and therefore will be written by the simulation user. In order to make his or her life as straightforward as possible, an ITS application Java class is provided that can be extended to define the functionality of the application. Hence, it needs to be implemented against this well defined programming interface and does not need to take into account which simulation tool is actually used. The interface wrappers for each supported simulation tool will handle the appropriate connection between the vtSim API and the simulation tools' programming interface. Depending on the relevant tools' interface implementation, the connection is realized in a way that is as appropriate as possible with respect to functionality and runtime efficiency.

5.3.2 Driver Behavior Modification

The most crucial aspect of a valid traffic simulation study is the gain of relevant, reliable and realistic results regarding traffic efficiency and/or safety. For this purpose, the models for traffic flow as well as for the driving behavior must be calibrated to ensure that the interpretation of the measures of traffic flow in the whole system can be relied upon. As a first step for driving behavior modeling and calibration, there must be a decision for an appropriate basic model and, as a result, a decision for a simulation tool which uses this specific base model if there is any already available. Each model needs input data from reality in order to calibrate it appropriately. In terms of driving behavior, microscopic parameters of driving processes must be assigned. The most important ones being acceleration, desired speed, desired/minimum following distance and lane changing constraints.

These parameters can be set based on previous knowledge, data taken directly from an equipped-vehicle or driving simulator, or can be fitted based on an iterative adjustment of the parameters to match global traffic flow measures. In either case the simulation must be subject to a validation against traffic flow data from the real world (which has not been used for calibration of the model). Traffic flow indicators for such a validation are traffic flow per hour, traffic density per kilometer and aggregated travel times among others.

Most simulation tools allow a flexible enhancement or even a complete replacement of the behavioral model. The framework makes use of this possibility to affect the driving behavior, within the scope of special situations which are relevant with respect to the ITS application defined (see above). Respective behavioral data can be gained by driving simulator studies or real world tests of ITS applications. Broadly speaking, there are two ways with which to enhance a traffic simulator behavior model in order to fulfill the requirements of a dedicated simulation study:

- Various behavioral data need to be collected, aggregated and correctly applied to the behavioral model thus extending the original model of the used simulation tool via new statistical distributions based on microscopic data from vehicles in the real world or in a driving simulator. The model is then affected purely by different parameters which can be pre-assigned or switched during runtime.
- Using a validated basic behavior model for general road traffic and overwriting it occasionally in specific situations. The regarded situations have to be defined clearly (e.g. reaction to a system warning). The model can then be dynamically switched during runtime.

The framework offers the flexibility to define the appropriate approach based on the configuration selected by the user.

5.3.3 Communication model

The communication model governs the communication processes that have an impact on the driving behavior within the scope of the ITS application under test. The IDs of those vehicles that have received messages will be handed to the ITS application which can use it as a basis for the implementation of the situational behavior model for those vehicles. The communication model can request traffic flow information from the respective simulation tool, and use this as a basis for deciding what communication takes place and is successful. This information is transferred from the simulation tool over a wrapper that is particular to the simulation tool in use.

5.3.4 Simulation tool

A simulation tool executes the flow of vehicles via basic driver-vehicle models, a model of the simulated road network and conducts simulations based on parameters (usually selected at random from a specified distribution). Usually, a graphical interface can be used in order to setup and monitor the simulation. From the simulation tool access to vehicles' desired speeds, desired destinations and each vehicle's position is required. Macroscopically, we require measures such as link speed and intensity to provide data to the scenario decision processes. Which traffic simulation tool(s) is/are to be used depends on the requirements and aims of the simulation study. Different simulation tools, model implementations and paradigms are flexibly interchangeable or even combinable within the framework.

5.3.5 Scenario Definition

The scenario definition, written in XML, defines which scenarios should be examined with which combination of tools. The relevant network portions, data sets and ITS applications are selected along with important parameters for the ITS application at hand (e.g. rate of equipped vehicles).

5.3.6 Results Manager

After a simulation study is conducted with the use of these components, the output data of the simulation tool(s) are collected by the results manager. The results manager module provides automatic processing of simulation results in order to gather indicator values regarding traffic safety and efficiency. Any simulation tool and model produces results in a very specific way that makes it difficult or even impossible to compare and aggregate without a deeper insight and a careful handling of those data. Additionally the aggregated data provided are often not readily interpretable regarding traffic safety measures. To overcome this, simulation specific data can be translated to the *Surrogate Safety Assessment Module* (Siemens Traffic Solutions, TX) format. This module (distributed free of charge as a Java Application) permits the calculation of various measures that are proxy indicators for safety such as Time-To-Collision and Post-Encroachment-Time[8]. The results manager can also handle all data given as input for the simulations thanks to the generic storage format used by vtSim. This allows easy comparison of real-world and simulated data for the purposes of calibration/validation.

6 FEASIBILITY STUDY

The implemented vtSim framework core has been tested in the scope of a feasibility study. It comprised a series of simulation studies analyzing a cooperative weather warning scenario. For the study we have used the simulation tool PTV Vision VISSIM, version 5.20. The connection between the vtSim application interface and the VISSIM C2X interface library is handled by UDP sockets. In addition to these connection components, the necessary data input, map input, assignment management and results display interface have also been implemented and tested.

6.1 Application and Behavioral Impact

The application evaluated is a cooperative warning of standing water on the road ahead. All vehicles (traveling with a mean speed of 80km/h) crossing the affected area send an alert, and following vehicles in reception range are warned of the danger and instructed to slow down to a speed of 40km/h. In this example, drivers who receive the alert comply with the information within the normal simulated range of deviation.

6.2 Communication Model

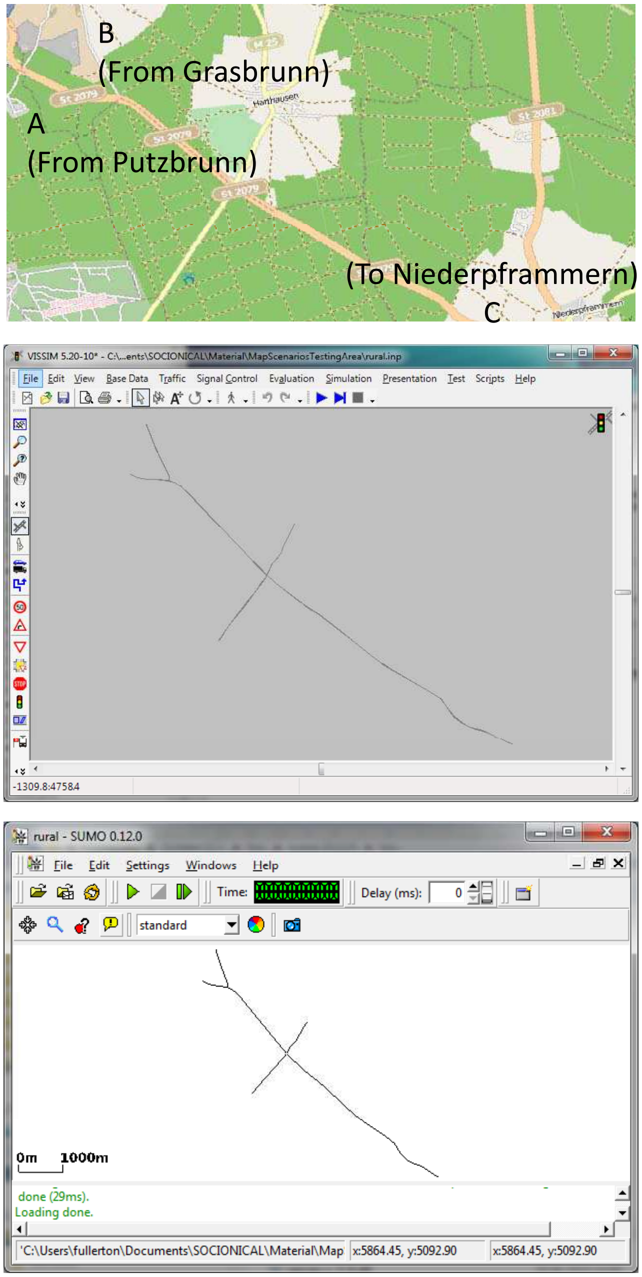
Considering that the communication can be added as an optional element in such a framework, it is very attractive to consider whether its simulation can be simplified without loss of accuracy of the traffic effects. Through hybrid simulation[17] a sophisticated communication *model* based on real data, instead of a full parallel simulation of the ongoing network communication, can be used[12]. The model of Killat et al. requires the position of the sending vehicle and all equipped vehicles and uses multiple probability distributions (according to surrounding vehicle density) based on ns-2 (*www.isi.edu/nsnam/ns/*) simulations to ascertain which vehicles successfully receive a message[12]. It is used here in a simpler form that ignores surrounding traffic density. The communication model provides back to the application a list of vehicles which have successfully received a message. This information is supplied by the same wrappers described above.

6.3 Demand and Map Input

The map for the simulation is taken from an edited portion of OpenStreetMap data between three villages in the greater Munich area (Figure 2). Traffic demand is entered directly using the Java user interface as fictitious traffic detectors. The demand (500veh/h entering from point A and 250veh/h exiting at point C, see Figure 2), is used internally to generate the traffic inputs and routes for the two simulations by the combination and scaling manager (Section 5.2). Hence, half the traffic will pass over the standing water area (on the final link towards point C) and half exits before this over a side road.

6.4 Results

The results show that traffic flow in both the no-intervention and application cases is extremely similar. With both simulation tools we see that traffic speed throughout the network is slowed as desired. The first simulation was conducted as a pure PTV Vision VISSIM 5.20 simulation without processing any external application. The relevant traffic flow profile is



done (29ms).
Loading done.

Fig. 2. Map transformation in vtSim. Top pane shows the feasibility study area (from *www.openstreetmap.org*). The lower two panes show the map transformed for VISSIM and SUMO respectively. The map is first transformed to the open source SUMO XML format before being imported into a Java representation of the XML-Schema whereby it can be used further and exported to VISSIM.

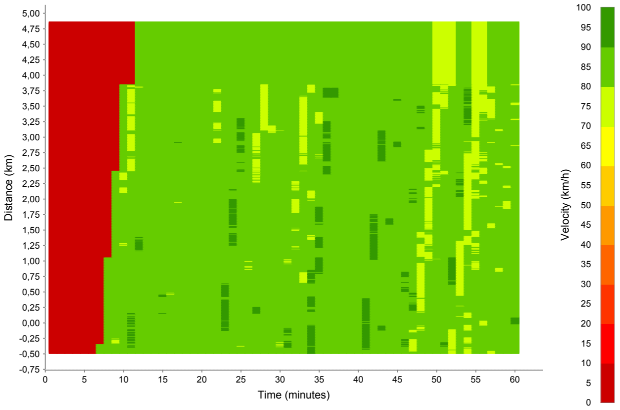
shown in the upper graph of Figure 3. As expected, no change in the speed profile can be seen. A second simulation study used the cooperative warning described above and implemented for VISSIM's C2X interface[12]. The warning is sent as soon as the vehicle enters the area of the dangerous weather condition. Here, the effect of the speed reduction warning is reflected in a decrease in average speed for the warning range. The speed reduction is seen at different locations owing to the different locations of vehicles at the time of reception. For the last simulation, another version of the cooperative weather warning was implemented, this time as a Java model that was connected with the vtSim application interface and the respective communication model abstraction. The simulation outcome is nearly the same as in the preceding simulation. Again, the speed reduction in consequence of the warning leads to a speed reduction for the surrounding traffic flow. The duration of the three simulations was equal (3600 simulation seconds). After every simulation run, the data were processed via the vtSim results manager. The simulation result plots were generated via the vtSim results manager and shown in Figure 3.

This result shows that the approach is technically feasible and suggests that the use of different techniques and models in the scope of the presented simulation framework is valid.

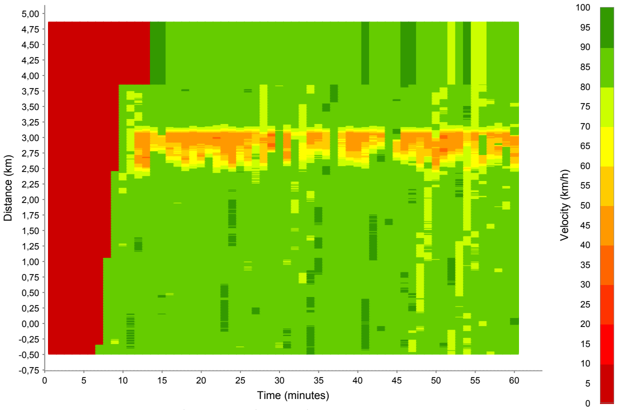
7 OUTLOOK

The need to examine potential vehicular internetworking systems requires flexible simulation strategies both because of the nature of the elements to be examined and the greater need for validation and comparison across tool sets. In addition, in such a rapidly developing field, it makes little sense for advances by one group of researchers to be unusable by others simply because of the tool choice made.

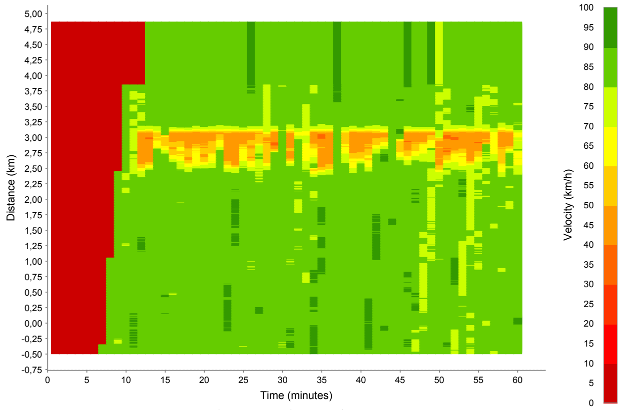
Here a flexible simulation framework, dubbed 'vtSim' has been proposed that addresses these issues. Several of the necessary new components of the framework that 'glue' existing tools together are already in use. The implementation has already made use of other open source software, in particular SUMO, and it is hoped it can continue to be extended to improve the diversity of data and tools used



1) VISSIM simulation without warning



2) VISSIM simulation with weather warning via VISSIM C2X API



3) VISSIM simulation with weather warning via vtSim

Fig. 3. Traffic speed profile for the area of a dangerous weather scenario (ca. at km 3): 1. without warning, 2. with weather warning in PTV Vision VISSIM via VISSIM C2X API, 3. with weather warning via vtSim using a connection to VISSIM C2X API. All figures were generated using the vtSim results manager.

from researchers elsewhere in order to foster greater collaboration and interoperability.

In order to accurately simulate the effects of ITS systems, the correct response of the driver must also be modeled, calibrated and validated[7], rather than assumed (Section 5.3.2). Although not discussed here, there are already widely-in-use systems for the gathering of human-response driving data, such as the driving simulation software SILAB (WIVW, Würzburg, Germany). In the future, we will explore whether automated integration of such data can also be applied to the application module, and what role such human data could also play in the results manager processing.

ACKNOWLEDGMENT

This work has been part-funded by the EU 7th Framework Programme research project “Complex socio-technical system in ambient intelligence” (SOCIONICAL) in the area of “Science of complex systems for socially intelligent ICT”.

REFERENCES

[1] A. Alvarez, R. Orea, S. Cabrero, X. G. Pañeda, R. García, and D. Melendi. Limitations of network emulation with single-machine and distributed ns-3. In *Proceedings of SIMUTools Conference, 2010*, March 2010.

[2] R. Barr, Z. Haas, and R. van Renesse. JiST: An efficient approach to simulation using virtual machines. *Software Practice and Experience*, 35(6):539–576, 2005.

[3] T. Dao, C. Clark, and J. Huissoon. Distributed platoon assignment and lane selection for traffic flow optimization. In *2008 IEEE Intelligent Vehicles Symposium*, pages 739–744, 2008.

[4] C. Dittrich, J. Lüssmann, F. Busch, R. Mänz, R. Stahlmann, and C. Menig. Greener roads by talking traffic lights - knowledge about queue length and next traffic light signalling. In *International Conference on Sustainable Automotive Technologies (ICSAT2010)*, 2010.

[5] EC DG for Transport and Energy. *DATEX II V2.0 User Guide*, 2009.

[6] S. Eichler, B. Ostermaier, C. Schroth, and T. Kosch. Simulation of car-to-car messaging: Analyzing the impact on road traffic. In *13th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems, 2005*, pages 507–510, 2005.

[7] M. Fullerton, A. Leonhardt, F. Busch, C. Maag, C. Mark, and H.-P. Kruger. Modeling Cooperative On-road Systems. In *International Workshop on Intelligent Transportation (WIT)*, 2010.

[8] D. Gettman and L. Head. Surrogate safety measures from traffic simulation models, final report. Technical Report FHWA-RD-03-050, Federal Highway Administration, 2003.

[9] C. Gorgorin, V. Gradinescu, R. Diaconescu, V. Cristea, and L. Ifode. An integrated vehicular and network simulator for vehicular ad-hoc networks. In *20th European Simulation and Modelling Conference*, 2006.

[10] J. Gozalvez, S. Turksma, L. Lan, O. Lazaro, F. Cartolano, E. Robert, D. Krajzewicz, R. Bauza, F. Filali, M. Röckl, J. Leguay, C. Michelacci, J. Vreeswijk, J. Maneros, A. Gonzalez, and M. Lenardi. iTETRIS: the framework for large-scale research on the impact of cooperative wireless vehicular communications systems in traffic efficiency. In *Information and Communications Technologies (ICT-MobileSubmit 2009)*, 2009.

[11] M. Haklay and P. Weber. OpenStreetMap: user-generated street maps. *IEEE Pervasive Computing*, 7(4):12–18, 2008.

[12] M. Killat, F. Schmidt-Eisenlohr, H. Hartenstein, C. Rössel, P. Vortisch, S. Assenmacher, and F. Busch. Enabling efficient and accurate large-scale simulation of VANETs for vehicular traffic management. In *Proceedings of the fourth ACM international workshop on Vehicular ad hoc networks*, pages 29–38. ACM New York, NY, USA, 2007.

[13] D. Krajzewicz, G. Hertkorn, C. Rössel, and P. Wagner. SUMO (Simulation of Urban MObility); an open-source traffic simulation. In *4th Middle East Symposium on Simulation and Modelling (MESM2002)*, pages 183–187, 2002.

[14] C. Lochert, A. Barthels, A. Cervantes, M. Mauve, and M. Caliskan. Multiple simulator interlinking environment for IVC. In *Proceedings of the 2nd ACM international workshop on Vehicular ad hoc networks*, page 88. ACM, 2005.

[15] D. Rieck, B. Schünemann, I. Radusch, and C. Meinel. Efficient traffic simulator coupling in a distributed v2x simulation environment. In *Proceedings of the third International ICST Conference on Simulation Tools and Techniques*. ICST, 2010.

[16] E. Schoch, M. Feiri, F. Kargl, and M. Weber. Simulation of ad hoc networks: ns-2 compared to JiST/SWANS. In *Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops*, page 36. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008.

[17] J. Shanthikumar and R. Sargent. A unifying view of hybrid simulation/analytic models and modeling. *Operations Research*, 31(6):1030–1052, 1983.