Modeling and Simulating ITS Applications with iTETRIS*

Jérôme Härri
EURECOM[†]
Sophia-Antipolis, France
haerri@eurecom.fr

Pasquale Cataldi
EURECOM[†]
Sophia-Antipolis, France
cataldi@eurecom.fr

Daniel Krajzewicz
German Aerospace Center
Berlin, Germany
Daniel.Krajzewicz@dlr.de

Robbin J. Blokpoel
Peek Traffic
Amersfoort , The Netherlands
blokpoel@peektraffic.nl

Yoann Lopez
Thales Communications
Paris, France
lopez@fr.thalesgroup.com

Jeremie Leguay
Thales Communications
Paris, France
leguay@fr.thalesgroup.com

ABSTRACT

This work presents the modeling methodology of the iTETRIS platform to integrate and simulate ITS applications. iTETRIS is a modular and open-source simulation platform composed of four key modules: the network simulator ns-3, the traffic simulator SUMO, an ITS (Intelligent Transportation System) application simulator, and a central federating module called iCS. Our contribution is twofold: First, we propose a methodology to model and simulate ITS applications with iTETRIS around three main mechanisms: (i) message management with generic open APIs based on subscription/result container mechanisms (ii) data management with the integration of an application facilities layer in the iCS, including a local dynamic map (LDM), (iii) application management with an ITS application simulator including one or more application logics. Second, we apply this methodology to implement the following four ITS applications: dynamic route planning, bus lane management, emergency vehicle, and contextual speed adaptation. We describe their integrations in iTETRIS, including a characterization of their interactions with the iCS, and illustrate the benefits of these ITS applications on traffic efficiency, gasoline consumption, or air pollutant emissions.

Categories and Subject Descriptors

I.6.5 [**Simulation and Modeling**]: Model Development—*Modeling methodologies*; H.4.3 [**Information Systems Applications**]: Communications Applications; C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design—*Wireless Communication*

General Terms

Design, Performance, Algorithms

Keywords

Intelligent Transportation Systems (ITS), Simulation Platform, ITS Application, iTETRIS, Performance Evaluation.

1. INTRODUCTION

Vehicular co-operative systems have been identified as strategic enablers to improve current transport systems by contributing to the global effort towards a safer, cleaner, and more efficient mobility of goods and people. Co-operative systems enable real-time exchanges of information among vehicles (V2V Communications) and between vehicles and road infrastructures (V2I Communications). They can help drivers and road operators to improve situation awareness, road traffic management strategies and road safety. Despite that co-operative ITS (Intelligent Transportation Systems) are currently being evaluated by Field Operational Test (FOTs) projects (e.g., EU DRIVE C2X, French SCOREF, German simTD), the inherent small-scale of such studies, as well as logistic and safety reasons, make evaluations via simulations a necessary step towards the deployment of ITS applications.

To assess the efficiency of co-operative technologies in realistic conditions, the FP7 iTETRIS project [2] has developed an advanced simulation tool, integrating both a road traffic and a wireless communication simulators (SUMO and ns-3 respectively). iTETRIS is yet not limited to the interlinking of two simulators, as other approaches such as TraNS [16] or Veins [6], but describes a specific modeling methodology and models requirements for the simulation of large scale ITS applications. First, the interconnected simulators should contain communication or traffic models required for the evaluation of ITS applications, which SUMO or ns-3 do not include. Second, ITS applications need to be integrated in fine into real vehicles but should also be a priori simulated. A novel modeling methodology is required for simulation platforms to integrate them transparently. Finally, ITS applications are based on a large number of ITS logics, such as Local Dynamic Maps (LDM), Traffic Light Control (TLC), Human-Machine Interface (HMI) or Human Reaction Models, which for modularity and extensibility reasons should be separately implemented from a common ITS application model. Accordingly, we see a federation of mul-

---

*Work presented at ACM MSWiM'11 Poster Session, October 31–November 4, 2011, Miami, Florida, USA.

[†] EURECOM industrial members: BMW Group, Cisco, Monaco Telecom, Orange, SAP, SFR, STEricsson, Swisscom, Symantec, Thales.

PM^2HW^2N'11, October 31, 2011, Miami, Florida, USA.

tiple modules, each of them providing a particular artifact or function to the ITS application (ex. vsimRTI [8]). The role and functionalities of the federating interface therefore becomes crucial. The interconnection of these modules also generates a large amount of information (commands, data) to be exchanged, especially considering large scale scenarios, and represents an important challenge to data and message management by the federating interface. The design of the federating interface consists therefore in a trade-off between modularity, data consistency and performance, and is fundamental for the modeling of ITS applications.

In this paper, we present the *iTETRIS Controlling System (iCS)*, the federating interface of iTETRIS and its modeling architecture and functionalities for the management of ITS applications. We propose a methodology for the modeling of large-scale ITS applications in iTETRIS consisting of the following three major aspects: (i) modular and extensible message management between modules via open APIs, (ii) cross-module data management with the implementation of application facilities, (iii) generic ITS application simulator consisting of an application proxy to the iCS and extensible ITS application logics. We then apply the described methodology to the evaluation of the following four ITS applications: dynamic route planning, bus lane management, emergency vehicle, and contextual speed adaptation. We illustrate how these applications are integrated in iTETRIS using the iCS, and also emphasize their benefits on traffic efficiency in terms of reduced travel time, fuel efficiency or reduced pollutant emissions. We finally would like to mention that iTETRIS has been thought as a community effort, and all contributions are available open-source in the community website[1] [2].

The rest of this paper is structured as follows: In Section 2, we describe the architecture of the interface iCS and the ITS application simulator. Section 3 illustrates the integration of four exemplary ITS applications and their respective performance evaluations. Finally, Section 5 summarizes this work.

2. MODELING ITS APPLICATIONS

Modeling ITS applications means the modeling of applications that will be integrated on-board of vehicles, fixed infrastructure nodes and traffic management centers (TMC). At the same time, these applications should also be evaluated based on simulations or emulations, and as such integrable into simulation platforms such as iTETRIS. These aspects require novel modeling methodologies on both an ITS application and an simulation platform sides to guarantee for instance:

Inter-operability An ITS application should be pluggable into an ITS simulation platform or on-board of a vehicle's electronics (GPS, communication unit).

Application interaction An ITS application needs to interact with various nodes not physically co-located, and at the same time, interact with different applications. The simulation platform should emulate data exchange in a transparent way to the ITS application.

Language agnostic The laguage employed by ITS applications might be different, simulation platforms should be able to integrate applications written in various languages (C++, java, python, etc..).

The iTETRIS platform addresses these aspect by employing a novel application modeling methodology. We first provide a brief introduction of the iTETRIS platform before describing the two main contributions of this work, the iTETRIS Control System (iCS) and the ITS Application simulator.

2.1 Brief Overview of iTETRIS

Figure 1 depicts the general architecture of iTETRIS with the iCS interface federating the traffic simulator SUMO, the network simulator ns-3 and multiple ITS applications. We describe in this section the key extensions of SUMO and ns-3 developped for iTETRIS.

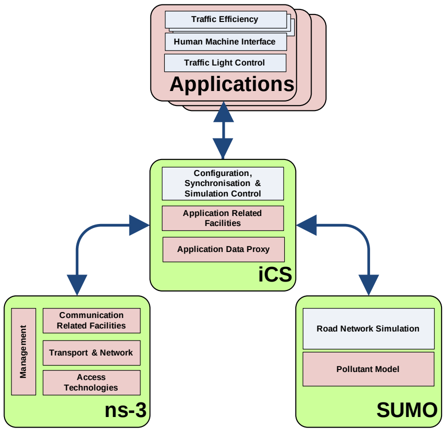

Figure 1: The iTETRIS simulation platform, with iTETRIS original functionalities in dark grey.

For simulating the behaviour of road traffic participants, the traffic simulation "SUMO" (Simulation of Urban MObility) has been chosen [5, 12]. SUMO is a microscopic, space-continuous and time-discrete traffic simulation, recently extended by the ability to simulate inter-modal routes. SUMO meets the requirements posed by iTETRIS, as it is open source under the GNU General Public License (GPL) since 2002, portable accross operating systems, and designed to be applicable for large road networks (200 000 traffic participants in real time[2]). Besides the traffic simulation itself, SUMO includes other applications for importing or generating the road networks and for converting different demand data into a format the simulation can handle. SUMO's functionalities and the specific extentions developed by iTETRIS are summarized in Table 1.

For simulating wired and wireless communication for ITS application, the Network Simulator 3 (ns-3) [3] has been chosen. ns-3 is the successor of the popular ns-2 simulator, but significantly differs from it by its new structural and modular implementation. Although ns-3 already contains a large number of protocols and access technologies, notably a IEEE 802.11p, it still lacks significant functionalities for the close-to-reality modeling and simulation of ITS application. The iTETRIS platform therefore added an ITS-specific communication stack, including mechanisms for Delay Tolerant

---

[1] *http://www.ict-itetris.eu/10-10-10-community/*

[2] *real time* means that the simulation of 1 second in SUMO corresponds to less than or equal to 1 true second

Functions	**Features**
Microscopic Model	Krauss Model
Macroscopic Model	O-D Matrix; weight-based shortest path
Extensibility	online APIs for route change, traffic lights, infrastructure retrieval, interactions with vehicles
Traffic Light Control	Embedded and through APIs
iTETRIS-specific Features	
Scenarios (Bologna)	Validated Urban (Pasubio-Costa); Suburban (Irnerio); Highway
Emission Model	HBEFA [10]
Noise Model	HARMONOISE [15]

Table 1: Summary of the available and extended functionalities of SUMO related to ITS Applications

Networks (DTN) protocols. The ETSI ITS-compliant [1] communication protocol stack implemented for iTETRIS is illustrated in Figure 2.

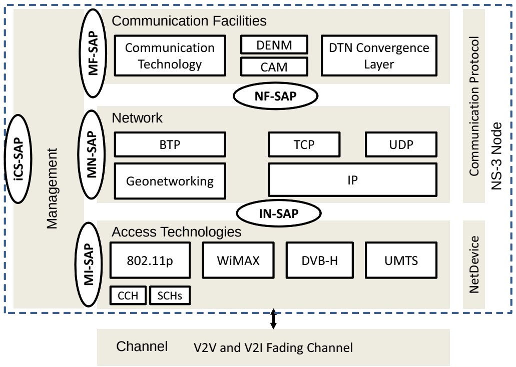

Figure 2: ITS protocol stack implemented in NS-3 for iTETRIS, with the mapping to ns-3 stack on the right. The iCS-SAP links ns-3 to the iCS.

2.2 iTETRIS Control System (iCS)

In order to handle the interactions of multiple ITS applications with ns-3 and SUMO, we describe here the architecture of the iCS. Compared to other V2X simulation platforms, where the interface is usually limited to a synchronization and data exchange between modules, the iCS not only provides larger functionalities but also adds specific models and functions to ns-3 and SUMO to support the evaluation of ITS applications (dark grey in Figure 1).

The iCS is an interface interconnecting various modules or simulators via *iCS-SAPs*. It is simulator agnostic and can be interfaced with different simulators by adapting the simulators to the iCS-SAP. Figure 3 depicts the architecture of the iCS showing more than synchronization fonctions:

Simulation Control As in all V2X simulation architectures, the iCS controls the simulation first by keeping all simulators synchronized and second by triggering actions in different modules. Moreover, in order to univocally identify the same stations on the different simulators, an identity management system is implemented.

Application Facilities The iCS contains the application-related facilities of the ITS ITS stack, and accordingly all their functionalities.

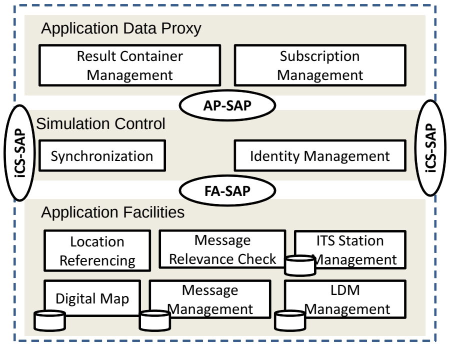

Figure 3: The iCS Architecture, where the iCS-SAP links it to external modules.

Applications data proxy The iCS finally manages the data exchange between different applications.

2.3 Scalability Design Aspects

Interconnecting multiple applications on large-scale scenarios (city-level) requires the exchange of a large amount of data between modules. In order to increase scalability and help the iCS keep data consistency, we implemented a data locality methodology, where an application payload is kept locally, and only a header and a reference to the application payload is actually transmitted. In particular:

- Application payloads are never sent to the traffic simulator. Communication is based on the transmitted header and dummy payload bits.
- Data required to be exchanged only within one application remains in the application module.
- Data required to be exchanged between applications is transmitted only to the iCS.

These design choices are however transparant to ITS applications and considering packet-level simulators such as ns-3, it does not impact communication confidence. The performance benefits from this methodology have been illustrated in [13].

2.4 Application Facilities

Application-related facilities are implemented in the iCS as a black box accessible by the synchronization manager using the *FA-SAP*. In this way updates in the communication layer functionalities and on the implementation logics will be hidden from the rest of the iCS. The iCS Facilities architecture is depicted in Figure 3, and we describe next the key blocks.

2.4.1 Digital Map

The iCS Facilities implement a digital map containing the road topology of the simulation area. At configuration time the SUMO configuration file containing the map description is parsed and the iTETRIS topological map is loaded. The data structure used for the iCS map closely resembles the SUMO one, but contains additional variables related to the topological entities (lanes, roads, etc, ...) that can be accessed/modified by iTETRIS applications (through custom request to the iCS). It is worth saying that the choice of implementing a topology map inside the iCS and not using the one provided by the road traffic simulator is motivated by two considerations. Firstly, the iCS is simulator agnostic, i.e. it is possible to use different road traffic simulators in place of SUMO (since the generic interfaces are defined, using a different road traffic simulator would only require to customise the generic interfaces and adapt the map loading parser). Secondly, using the topology map of the road traffic simulator would have been expensive in terms of number of packets exchanged by the two iTETRIS modules.

2.4.2 ITS Station Management

The Facilities maintain the list of stations with their latest information which can be accessed by queries from the iCS logic. In particular, stations are divided into mobile stations and fixed stations. Mobile stations can have several communication interfaces (WAVE, DVB-H, UMTS and WiMAX) defined at configuration time. The communication interfaces are active at the beginning of the simulation but can be controlled by the applications through iCS subscription commands. Fixed stations, instead have only one communication interface as fixed stations with several interfaces can be considered as several infrastructure stations linked to each other (as the iTETRIS platform does not simulates wired communication).

2.4.3 Messages Management

Communication message payloads among stations are handled by the Facilities block. In particular, the payload of facilities-related communication messages (CAM and DENM) are generated directly by the block considering information from the station information database, from the map and, if necessary, from applications. As the iTETRIS platform does not simulate real data packets, the generated payload is stored in a database for transmitted messages and a reference for this packet, together with its size and type, is passed to the iCS logic which will forward the sent request to the network simulator and collect the transmission result.

2.4.4 Local Dynamic Map Facilities

The facilities block implements the LDM functionalities. When the iCS collected the transmission results from the network simulator, the facilities messages results are passed back to the Facilities block which will look up the transmitted messages database to retrieve the payload and will insert the received packet inside a *received messages database.* This database contains all the facilities messages generated inside the simulation area that were received by stations. The entry will also contain the list of the receivers and the reception times. As we can observe, this implementation allows to significantly reduce the storage space with respect to implementing a different LDM per each station as the same message could be received by several stations.

2.5 Application Data Proxy

Exchanging data between applications, network and traffic simulators is provided by *Subscription / Result Containers* mechanisms. A Subscription is an API developed to retrieve and send data to other modules, or initiate iCS functionalities, while a result container is an object holding results from an ITS application. As illustrated in Figure 4, connecting an ITS application to the iCS requires the implementation of both objects.

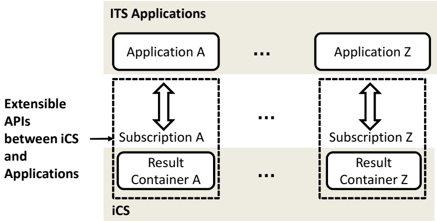

Figure 4: Interface modeling between the iCS and ITS Applications - Subscriptions and Results Containers.

2.5.1 Subscriptions

The iCS contains subscriptions related to basic functions (i.e. CAM transmissions, position from traffic simulator, etc..). To favor extensibility and modularity, we also developed a set of generic **push(MSG)** and **pull(MSG)** subscriptions for ITS applications to interact with the other iTETRIS modules. A **push** subscription sends a value or a command to a module, while a **pull** subscription retrieves values from a module. As an ITS application may interact and influence all iTETRIS modules (i.e. iCS, SUMO, ns-3 or even other applications), we specified a set of push/pull subscriptions for each type of interaction, rather than per type of message. For modularity reasons, each subscription accordingly employs a *generic Type-Length-Value (TLV)* data encoding format.

```
MSG :=  <NODE_ID><LENGTH><MSG_VALUE>
MSG_VALUE := (<TAG><LENGTH><FIELD_VALUE>)*
FIELD_VALUE := (<FIELD><VALUE>?)*
```

where $<TAG>$ represents a specific type of data, $*$ indicates one or multiple values, while ? mentions that the $<VALUE>$ is optional (it may only be found in **PUSH** subscriptions).

For example, if an ITS station requires to get its current position and speed from the iCS, it will use the following subscription to the iCS:

```
pull_ics(<Station_ID><4><TOPO><3><X><Y><SPEED>)
```

If the TMC needs to order SUMO to reroute an ITS station, the subscription will take the following form:

```
push_sumo(<TMC><2><REROUTE><Station_ID>)
```

2.5.2 Result Containers

As ITS applications produce various type of data, the facilities block (ex. LDM) cannot provide a structure to store them. For flexibility reasons, result containers are therefore introduced to provide a modular data proxy functionality between applications and with other modules. They are composed of a database storing data according to a flexible format. Each result container is attached to a station and a subscription.

2.6 ITS Application Simulator

The ITS application simulator is the fourth module of the iTETRIS platform as illustrated in Figure 1. Its architecture consists of two blocks: one or several ITS application logics and an application proxy (see Figure 5).

The application logics represent the actual intelligence or the cooperative strategies of an ITS application running on the simulated stations. They interact with the iCS via the

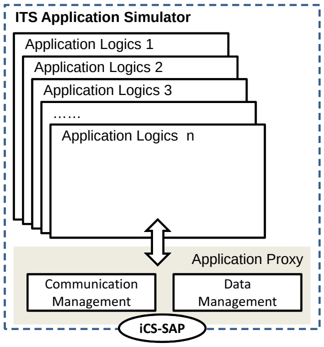

Figure 5: The ITS Application architeture, allowing interlinking of ITS applications and the iCS via the iCS SAP.

application data proxy using the subscription/result container paradigm as presented in Section 2.5.

The application proxy contains two blocks. Similar to TraCI, the Communication Management block is in charge of opening sockets connections, and provides APIs to interact with the iCS. The Data Management block temporarily stores application data to be exchanged between application logics belonging to the same ITS application. For the application logics, application data is sent over the air, although it actually remains locally stored in a database of the data management block. Please note that in the case of multiple applications, application data is exchange via a result container.

In conclusion, simulating an ITS application requires the implementation of the ITS logics into the ITS application simulator, an application-specific result container to store application-specific data at the iCS, and to use the available subscriptions to send or retrieve data between modules.

3. EVALUATING ITS APPLICATIONS

We illustrate in this section the integration capabilities of iTETRIS with the evaluation of four examplary ITS applications. We start by describing the reference scenarios employed for the evaluation of these applications.

3.1 Scenario

We used two traffic scenarios from the city of Bologna (see Figure 6) corresponding to average traffic situations summarized in Table 2. Communication parameters of the scenarios are described in Figure 3. The two scenarios, calibrated on real traffic, have been obtained from the Community of Bologna (COBO), converted to SUMO using tools from the SUMO package, and enhanced by additional data supported by COBO in the form of ArcView shapefiles. The traffic demand was mapped from the original networks onto the obtained SUMO networks using a network-matching script. The first three applications use Pasubio-Costa, whereas the fourth one is evaluated on Ringway. Without loss of generality, we only employed IEEE 802.11p access technology for the communication aspect, as we focused on the connectivity rather than infrastructure dimentioning. The rest of the scenarios specifications are omitted for space restrictions.

3.2 Speed Advise Applications

As illustrated in Figure 8, the speed advice application aims at preventing vehicles from stopping at a traffic light.

(a) Ringway. (b) Pasubio-Costa.

Figure 6: Two traffic scenarios in Bologna

Scenario	# Junctions	# Roads	# Vehicles
Ringway	1210	2216	2000
Pasubio Costa	309	271	950

Table 2: Traffic characteristics of the scenarios

This way CO_2 emissions can be reduced since the kinetic energy of the car will not be dissipated in the brakes and vehicles would need reduced energy to accelerate again. The success of the application is very dependent on the predictability of the traffic light controller. When the controller changes the plan after an advice has been given, then obviously there is a high chance that the advice will not be effective anymore. Therefore, the application was used in combination with a controller that fixes its control plan approximately 60 seconds into the future. This is equivalent to a fixed time controller or a network controller that plans ahead well. The first vehicle of a green phase will receive a speed advice at a distance D from the traffic light that will make it arrive at the stopline 2 seconds after the light turns green. Note that arriving exactly at the moment of green would require the driver to have a lot of confidence in the system since stopping in case of an error is not possible anymore. The following vehicles will be advised in such a manner that they will be 20 meters behind the previous at the moment the light turns green. This distance is required because they have to keep a safety distance between them. Platoon borders are determined by whether a vehicle still fits within a green phase and if the speed advice does not need to be above the legal limit in order to reach the stop line before the light turns red. When either of those conditions is not met, the vehicle will become the first vehicle of the next platoon.

Within the iTETRIS platform the application was built according to the flow diagram in Figure 7. At startup the application will subscribe to the iCS to receive all cooperative vehicles entering a certain zone. This zone corresponds to the advice point and the flow diagram of the operations of the application starts there. The message with the advice is a unicast message because every vehicle gets a personalized speed based on both the traffic light planning and the

Protocol	BTP, Geobroadcast, Geounicast
Technology	IEEE 802.11p
Transmit Power	30 dBm
Channel Model	WINNER B1
Techno/Application Penetration	50%

Table 3: Communication characteristics of the scenarios

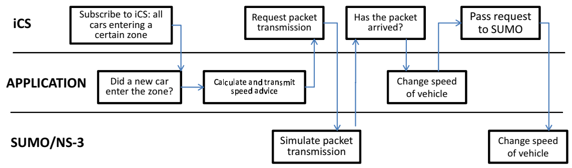

Figure 7: Flow diagram of the interaction between the three modules and the iCS for the Speed Advise Application.

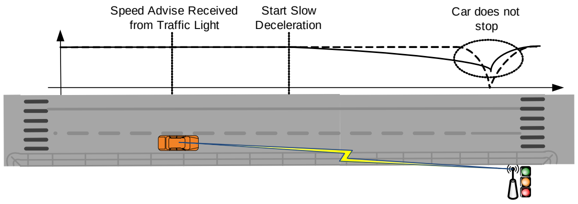

Figure 8: The Speed Advise Application, where traffic lights indicate optimized speed to avoid a full stop.

number of vehicles in front of him. The application results in terms of CO_2 reduction are presented in Table 4. We can conclude that increasing the speed limit and giving the advice earlier reduces the CO_2 emissions mostly.

3.3 Bus Lane Management

The city of Bologna is an old city with small narrow streets. Some of the streets in the city center are restricted to public transport. On the other hand big events such as football matches or concerts attract many people who use private transport. In such cases the traffic demand is increased. The idea of the "Bus Lane Management" strategy is to open the bus lanes for private vehicles when an unusual increase in the traffic demand is recognized (see Figure 9). When the traffic decreases to a normal state the bus lanes should again be restricted to public transport only. The aim of the application is to increase the capacity of the network and reduce the travel time and traffic jams.

The "Bus Lane Management" strategy includes two different actions. The first one is to recognize a higher traffic demand. This is done by evaluating the cooperative awareness messages (CAMs) broadcasted by equipped vehicles. These messages can be collected by Road Side Units (RSUs) which are located along the roads. Each RSU calculates the average speed of the equipped vehicles in their communication range. A low average speed indicates an increased traffic demand. Using this information the RSU decides whether the bus lanes should be opened for private vehicles.

The second action is to open bus lanes and to reroute equipped vehicles. When the RSUs recognize an increasing traffic demand they open the bus lanes and broadcast this information. Equipped vehicles receive the information about the possibility to use additional routes and determine whether this allows them to reach their destination faster.

Speed limit km/h	50	50	50	70	70
Distance m	none	500	1000	500	1000
CO2 reduction	0%	13.67%	24.84%	22.21%	28.31%

Table 4: Benefit of the Speed Advise application on carbon footprint as function of speed limit and notification distance

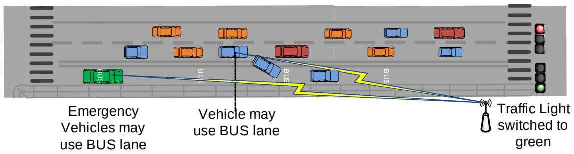

Figure 9: Emergency vehicle and bus lane manangement applications, where selected vehicles can use the bus lane and avoid traffic.

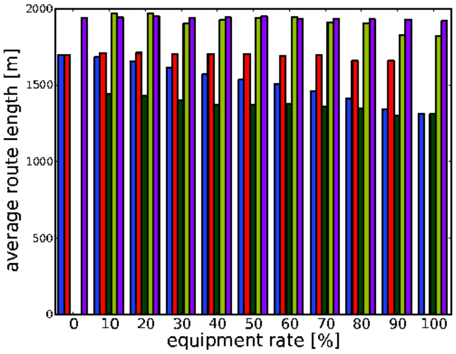

Figure 10: Average route length

Results depicted in Figure 10 show that vehicles gain shorter routes. Not only the route length was shorter but also the travel time could be reduced for all vehicles (see Figure 11). The benefit can only be achieved for penetration rates until 50%. It should be noted that not all equipped vehicles are rerouted. Whether a vehicle is rerouted or not strongly depends on the street network and the chosen route. A vehicle is only rerouted if another route seems to be faster according to the expected travel time.

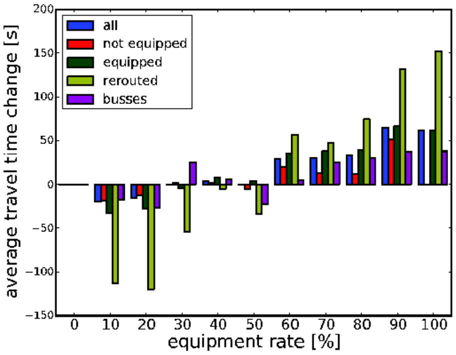

Figure 11: Average travel time

3.4 Emergency Vehicle

Emergency vehicles (e.g. ambulances, fire trucks and police cars) need to be at the incident location as fast as possible. In case of an emergency they have special rights. They are allowed to violate red lights, overtake other vehicles or

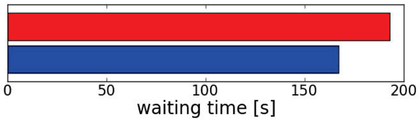

Figure 12: Travel time of emergency vehicle equipped (bottom) and non-equipped (top)

use bus lanes if these actions do not harm other traffic participants. A siren and a blue flashing light are the signals to other drivers that an emergency vehicle is approaching. But some drivers do not notice these signals. Therefore emergency vehicles have to stop at red light and check whether all other vehicles give priority to them. It is still dangerous for drivers of emergency vehicles to violate a red light, and emergency vehicles are statistically more likely to have an accident than other vehicles.

The following strategy aims at helping emergency vehicles to reach their destination faster by switching the traffic lights to green for them. Emergency vehicles are broadcasting CAMs with their current location and their expected route. RSUs are receiving the information of an approaching emergency vehicle and inform the traffic management center. Traffic lights on the route of the emergency vehicle can then be switched to green for the emergency vehicle and to red for all other vehicles by the traffic management center. When the emergency vehicle has passed the intersection the traffic light can continue its normal operation.

The simulation of a scenario with an emergency vehicle shows that the travel time of the emergency vehicle can be improved by this application. Figure 12 shows the travel time of an equipped emergency vehicle and an emergency vehicle without the application. In the scenario the emergency vehicle can reach its destination 40s earlier as the non-equipped. The speed gain is achieved because vehicles coming from the same direction as the emergency vehicle can use the green light, too, and free the way for the emergency vehicle.

3.5 Request-Based Personalized Navigation

Considering dynamic traffic conditions, obtaining personalized and updated navigation solution to reach a destination can be beneficial. A driver may therefore request a personalized navigation solution to the Traffic Management Center (TMC), which then computes the best route from the user current position to its destination, and replies with a route recommendation avoiding traffic. Would a new traffic condition would make the route, would the TMC update the driver with a new route. A two-way communication is therefore established between a vehicle and the TMC using a RSU covering the area where the user is located. The optimal route is computed based on the aggregated travel time on the routes, which is constantly gathered by vehicles running another distributed application called *travel time estimation* (not described here for the lack of space) running in parallel. If such information is not available, the TMC returns the distance-based shortest path according to the navigational map.

Within the iTETRIS platform, the application has been built following the flow diagram illustrated in Figure 13. The application subscribes to the iCS for the fraction of vehicles corresponding to the application penetration rate. If the current route becomes too long, a new route is computed by SUMO, which is then transmitted via unicast to the vehicle.

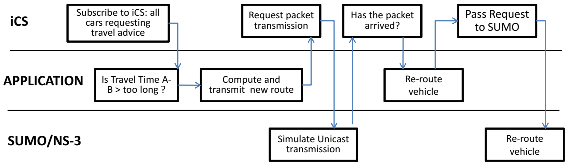

Figure 13: Flow diagram of the interaction between the three modules and the iCS for the Personalized Navigation.

Metric	**Reference (SUMO)**	**RBPN**	**Benefit**
Travel time (s)	82.82	31.95	65.5%
Traffic Density [vhl/km]	69.9	31.6	54.8%
CO2 (mg)	5.17e5	3.79e5	26.7%
NOx (mg)	9.51e02	7.22e02	24%
Fuel Consumption (ml)	206	151	26.7%

Table 5: Benefit of the RBPN application on non-routed or equipped vehicles.

As results illustrating the benefits on vehicles being rerouting has been provided in the previous section, we opted to take a different look and describe here the benefits to vehicles not being re-routed or not equipped (50% in this scenario) from vehicles being re-routed. We regrouped values for a selection of performance metrics in Table 5. As it can be seen, when all vehicles equipped are re-routed, a significant benefit can also be obtained by non-equipped vehicles. 65% of travel time can be obtained, a 26% and 24% reduction of critical gas emission (CO2 and NOx respectively) can be reached, while saving 26% in fuel consumption. These values are obviously large as we consider a large penetration rate and reflect the benefits of the application on non re-routed vehicles from the vehicles being re-routed. It yet shows that even non-equipped vehicles could benefit from future ITS applications and open the discussion on re-routing policies.

4. RELATED WORK

A large literature in simulation solutions may be found today about simulating mobility and networks. Due to the large number of simulators in this area, we will not cover them in detail in this paper and refer the readers to [9] for a detailed survey. We cover in this sections the major modeling methodologies found in literature for ITS. As mentioned in [9], modeling and simulating ITS applications may be found in the classes of *embedded* methodology, where various models (mobility, network, applications) are all embedded in a single simulator, or *federated* methodology, where various simulators are synchronized and federated for multi-model simulations. Some complete simulators such as nctuns [7] or MoVES [4] may be found in the former methodology, but modularity and extensibility reasons motivated the developments of multiple work in the latter and showed an increasing popularity in the ITS community. Due to the requirements of ITS applications of bi-directional interactions, interfaces synchronizing various network or mobility simulators have been widely proposed, as in [6,14,16] for the most popular ones. Their common aspects are APIs for data exchange and clock synchronization. However, when modeling ITS application, one major *application* block, contained either in network or in the mobility simulators in the cited work, would be required, making such federation a *trium vira*. Yet, adding a new application block also required an advanced data management and an increasing role of the interface, which is not considered by these related work. VSimRTI [8] represents one of the new proposals generalizing the role of an interface between multiple simulators, extending basic data exchange with data storage. However, this closed source approach limits the extension of the capabilities and data management of the interface. iTETRIS goes one step further by adding extensible functionalities in the interface, such as Local Dynamic Maps and other facilities features, and provide this as open source.

Beside an efficient inter-linking between simulators, the models available in each simulator also need to be adapted for ITS simulations. In the large traffic simulators cited in [9], mobility can be assumed to be correctly modeled. Yet, the iTETRIS project also required modeling of pollutant emission and other traffic related metrics to evaluate the benefit of ITS application, which is largely lacking in these simulators. iTETRIS therefore extended the traffic SUMO with an emission model and specific APIs to let ITS applications observe the impact on the emission footprint. Considering now the network simulator module, most of the simulators cited in [9] contain an implementation of the vehicular specific WLAN, IEEE 802.11p, but the general compliance with the other protocol stack as specified in IEEE 1609, ISO CALM or the ETSI TC ITS is lacking. For example, the facilities layer contain problably the intelligence of the ITS systems and is widely absent of all simululators. Also, DTNs also require specific modeling methodologies for C2C communication, which are usually found in separate simulators (see ONE [11] or Whitbeck et al. [17]). Accordingly, the iTETRIS platform represent a complete and open source ITS evaluation platform capable of interlinking and completing state-of-the-art simulators required for the close-to-reality modeling of ITS application and their evaluation.

5. SUMMARY AND FUTURE WORK

Field Operational Tests (FOT) represent a key step in the preparation of the mass deployment of co-operative technologies. As a cost-efficient tool for the preparation of FOT and to help sizing future ITS deployments, iTETRIS is the perfect candidate to perform large-scale technological and functional evaluations. Up coming FOT projects will assess co-operative technologies in real environments. However, the impact of this technology for road traffic management is difficult without a mass deployment. We strongly believe that iTETRIS will be of great support to prepare these field tests and allow to scale their outcomes.

6. ACKNOWLEDGMENTS

This work has been partly funded by the European Commission through FP7 ICT Project iTETRIS: An Integrated Wireless and Traffic Platform for Real-Time Road Traffic Management Solutions (No. FP7 224644). The authors wish to acknowledge the Commission for their support. We also wish to acknowledge our gratitude and appreciation to all the iTETRIS project partners for their contribution during the development of various ideas presented in this paper.

Jérôme Härri further acknowledges the support of the French DGCIS/OSEO for the FOT Project SCOREF.

7. ADDITIONAL AUTHORS

Additional authors: Christian Bonnet (EURECOM, email: `bonnet@eurecom.fr`) and Laura Bieker (DLR, email: `laura.bieker@dlr.de`)

8. REFERENCES

[1] ETSI Technical Committee on ITS, 2011, *http://www.etsi.org*

[2] The FP7 iTETRIS Project Portal, 2011, *http://www.ict-itetris.eu/index.htm*

[3] The Network Simulator 3 (ns-3), 2011, *http://www.nsnam.org/*

[4] L. Bononi, M. D. Felice, G. D'Angelo, M. Bracuto, and L. Donatiello. Moves: A framework for parallel and distributed simulation. *COMPUTER NETWORKS*, (52):155–179, 2008.

[5] DLR. Simulation of Urban Mobility SUMO, 2011, *http://sumo.sf.net/*

[6] F. Dressler and C. Sommer. VEINS - vehicles in network simulation, 2011, *http://veins.car2x.org/*

[7] EstiNet. NctuNS network simulator and emulator, *http://www.estinet.com/products.php*

[8] Frauhenhofer Focus. VsimRTI, *http://www.dcaiti.tu-berlin.de/research/simulation/*

[9] J. Härri, F. Filali, and C. Bonnet. Mobility models for vehicular ad hoc networks: a survey and taxonomy. *IEEE Communications Surveys & Tutorials*, 11(4):19–41, 2009.

[10] INFRAS. The Handbook Emission Factors for Road Transport (HBEFA), 2011, *http://www.hbefa.net*

[11] A. Keränen, J. Ott, and T. Kärkkäinen. The ONE Simulator for DTN Protocol Evaluation. In *Proc. SIMUTools*, 2009.

[12] D. Krajzewicz. *Traffic Simulation with SUMO - Simulation of Urban Mobility*, pages 269–294. Springer, 2010.

[13] V. Kumar, L. Lin, D. Krajzewicz, F. Hrizi, O. Martinez, J. Gozalvez, and R. Bauza. itetris: Adaptation of its technologies for large scale integrated simulation. In *Vehicular Technology Conference (VTC 2010-Spring), 2010 IEEE 71st*, pages 1 –5, may 2010.

[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*, pages 87–88, 2005.

[15] R. Nota, R. Barelds, and D. van Maercke. Harmonoise WP 3 Engineering method for road traffic and railway noise after validation and fine-tuning. Technical Report Deliverable 18, HARMONOISE, 2005.

[16] TraNS. Traffic and network simulation environment, *http://lca.epfl.ch/projects/trans.*

[17] J. Whitbeck, Y. Lopez, J. Leguay, V. Conan, and M. Dias de Amorim. Relieving the wireless infrastructure: When opportunistic networks meet guaranteed delays. In *12th IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM) (IEEE WoWMoM 2011)*, Lucca, Italy, 6 2011.