A Semantic Framework for Integrated Modeling and Simulation of Transportation Systems

Qunzhi Zhou[*], Viktor Prasanna[*†], Hwan Chang[*], Yun Wang[*]

[*]*Ming Hsieh Department of Electrical Engineering, University of Southern California,*
[†]*Department of Computer Science, University of Southern California,*
(qunzhizh, prasanna, hwanchan, yunwang)@usc.edu

Abstract: A number of modeling and simulation tools exist for studying transportation systems. The value of these tools would be significantly increased if they can be used in an integrated manner to investigate scenarios involving domain aspects modeled by different tools. This paper describes the design of an integrated traffic modeling and simulation framework using Semantic Web and workflow technology. The core component of the system is a domain ontology model which captures domain concepts and forms the common vocabulary for data and application integration. The proposed framework is designed to be modular and extensible, and to accommodate advanced visualization and human interaction together with data access and management capabilities. It hides the disparity of data formats, models and tools from domain experts and provides a single logical space for simulation design. We describe the technologies to achieve these goals and discuss the methodologies through representative use cases in studying transportation systems and control algorithms.

1. INTRODUCTION

Globalization, international trade and economic development have led to an increase in the volume of traffic flows by all transport modes. Traffic at ports, rail yards, warehouse distribution nodes and metropolitan areas contributes to congestion, traffic accidents and environmental pollution and puts tremendous pressure to deal with such issues. A prerequisite for proposing any solution that would make the existing transportation systems more efficient - especially in complex environments such as in a metropolitan area - is a careful and accurate understanding and analysis of traffic flow. Economic forecasts are unanimous in predicting the need for additional transportation capacity or control strategies in order to keep pace with rising demand. Adding capacity or introducing technologies by following the traditional way of building new infrastructures is no longer as feasible as it used to be especially in metropolitan areas where the evaluation and implementation cost are obstacles hard to overcome.

The state-of-the-art method for studying transportation systems is simulation modeling which affords the opportunity to evaluate control and design strategies without committing expensive and time-consuming resources to implement alternate strategies in the field. Major efforts have been invested in developing such simulation systems during the past a few decades (see Section 2). Most of them are based on a mathematical approach which consists of reproducing vehicle streams in specific areas at a certain granularity (Arnaud *et al*, 2006).

A holistic evaluation of the impact of different transportation policies requires a comprehensive simulation environment which models all aspects of traffic systems. This by itself is a very challenging problem due to the presence of multiple modes of transportation which interact and affect each other in a complex manner and cannot be captured by a single existing simulation model (Ioannou *et al*, 2007). To facilitate the introduction of new technologies and methods to improve transportation efficiency and keep pace with rising demand, a framework for integrated traffic modeling and simulation which supports integration of a multitude of simulation tools is needed.

Advances in information technologies, data management and computation tools, together with recent research in systems optimization and control, makes possible new approaches to solve the problem. In this paper, we propose *a semantic framework for integrated traffic modeling and simulation*. Heterogeneous simulators, data sets and computational models can be easily integrated to the simulation environment and used to model and simulate complex transportation systems. Some of the key challenges we identified are:

Model integration: Multiple simulation models capture different aspects of a complex system. One key goal of our work is to enable the evaluation of complex simulation scenarios where each scenario represents a certain realization of the system or a certain operational strategy. An evaluation of the scenario is a forecast of the system response and an optimization chooses the ideal designs or operations. Analysis at the system level necessitates that the data in the many models and the results of their simulation be merged in meaningful ways.

Model management: A traffic system could have alternate models depending on the simulators and modeling approaches used. An alternate model may also represent a variation of the system with a different set of values of parameters. The integration framework should provide services to manage legacy models including search, view and retrieval for reuse.

Application integration: Researchers/engineers may use different simulation tools to model traffic systems, and use a variety of computation tools to develop control algorithms and data analysis models etc. The integration framework should be designed with modularity and extensibility in mind. Domain experts would be able to insert heterogeneous tools into the simulation environment with little effort and without affecting other modules. Therefore the system will be an evolving framework allowing additions, improvements and changes on a modular basis while maintaining the integrated nature.

User focus: It is also important to note that the ultimate purpose of the integration framework is to assist domain scientists (e.g., control theorists) who may be not computer experts in the rapid prototyping and execution of integrated simulations. Our experience has shown that "soft" factors such as the design of user interfaces, novel visualization tools, intuitive workflow wizards, etc are as important from the end users' perspective as the underlying technological solutions.

The problems addressed by our framework touch upon some of the core issues of information integration (Calvanese *et al*, 1998). However, in this paper, we focus on the problem of providing a unified environment to define integrated traffic simulation scenarios and execution models from user point of view. Our framework employs semantically enhanced models and meta-models as the core design principle. A domain ontology model acts as the key component to enable data and application integration, and is also the basis for the intuitive user interfaces that allow domain experts to interact with our framework. The prototype integration system has been deployed and tested to address challenging use cases.

The rest of the paper is organized as follows. We review existing traffic simulation tools in Section 2 and traditional integration approaches in Section 3. Section 4 discusses the design of a semantic framework for integrated traffic modeling and integration. Section 5 represents case studies which apply the framework to representative simulation and traffic control applications. Finally we draw conclusions and summarize the paper in Section 6.

2. A REVIEW OF TRAFFIC SIMULATION TOOLS

Traffic modeling and simulation are among the most useful approaches for traffic engineers and researchers to conduct traffic engineering, theory studies and operations evaluation analysis (Cai *et al*, 2004). The general idea is to build a computer model for traffic systems that reflects real-world conditions. Once the model is constructed, engineers could experiment with different configurations and determine their impacts on the system. From its emergence, computer simulation in traffic analysis has been developed from a research tool of limited group of experts to a widely used technology in research, planning, demonstration and development of traffic systems. The elementary application areas of simulation mainly remained the same, but the applications have grown a lot in size and complexity.

Existing traffic system simulation tools and models can be categorized into macroscopic, microscopic and mesoscopic ones according to their modeling granularity. Macroscopic models (e.g., Elloumi *et al*, 1994; Buisson *et al*, 1996) tend to model traffic as a continuous flow, often using formulations based on hydrodynamic flow theories. These models are mostly used for planning applications and operations control design involving large networks and long time periods. Microscopic models (e.g., Duncan *et al*, 1997; CORSIM; VISSIM) capture the behavior of individual vehicles and drivers in great details including interactions among vehicles, lane changing, and behavior at merge points. Increasing the modeling detail and hence the granularity of simulation improves accuracy of the simulation results but also increases the complexity of computation and sensitivity to modeling errors. Microscopic simulators are suited to model vehicle interactions at high-level of detail especially in the evaluation of ITS, but are limited to small areas, due to the avalanche of input data and extensive calibration requirements. Mesoscopic models (e.g., Jayakrishnan *et al*, 1994; Ben *et al*, 2002) maintain individual vehicle representation but with a more aggregate representation of traffic dynamics usually by using speed-density relationships and queuing theory approaches. They provide a middle ground with their ability to model large networks with limited network coding and calibration effort, while providing a better representation of the traffic dynamics and travel behaviors (i.e. route choice) than their macroscopic counterparts.

According to the types of target systems they are capable to model and simulate, traffic simulation tools might also be subdivided into terminal simulators, roadway network simulators etc.

Terminal simulators model special infrastructure inside terminals – yards, gates, logic of freight flow, etc. – in an attempt to simulate traffic flows inside terminals accurately. For example, TermSim (Ioannou *et al*, 2007) is a microscopic marine terminal simulator. It models the inbound/outbound gates, import/export yards, ship yard, train station inside marine ports and their impact on the truck flows. To analyze the movement of trucks in and out of the terminal, TermSim is keeping track of each individual truck as a separate object. The truck object is being followed around the yard, as it performs its various functions, such as loading a full container from the import yard; unloading a full container at the export yard; picking up an empty container which will be taken out of the terminal and loaded with export goods, etc., thus simulating the movement of trucks inside the terminal at a microscopic level. TermSim simulator constitutes a complete simulation environment where a number of model parameters are configurable, such as truck arrival rates, gate processing times for inbound and outbound gates, ship and train arrivals, inflows and outflows to the storage yard, yard capacities etc.

Traffic network simulators model the traffic flow on roadway networks. The most widely used network simulation tools include CORSIM, VISSIM and a number of mesoscopic roadway simulators. These tools are designed to simulate vehicle movements on combinations of surface transportation networks and typically support signal control and other operational strategies.

CORSIM is a microscopic, time step roadway simulation model designed for the analysis of freeways, urban streets and corridors. It's capable to simulate different intersection controls, almost any surface geometry including number of lanes and turn pockets and a wide range of traffic flow conditions (Owen *et al*, 2000). CORSIM encodes network based on a link-node structure. The links represent the roadway segments while the nodes mark a change in the roadway, an intersection or entry points. During simulation, CORSIM moves simulated vehicles and updates traffic signals every time step.

VISSIM is a microscopic, time step and behavior based simulation model also capable to analyze the full range of functionally classified roadways and public transportation operations. It differs from CORSIM in terms of network coding structures, signal modeling logic, car-following models, etc. VISSIM consists of two primary components: simulator and signal state generator (SSG). The simulator generates traffic and is where the user graphically builds the network. VISSIM employs a link-connector network model. The lack of a node structure provides users the flexibility to control traffic operations and vehicle paths within connectors which represent intersections or interchanges. The SSG is where the signal control logic resides. Users have the ability to define and emulate any type of signal control logic.

A number of mesoscopic traffic network models exist in the literature. DYNAMIT (Ben *et al*, 2002), for instance, uses individual vehicles moving along segments according to speed/density relationships and queuing models. Lanes are simulated in detail only when congestion builds up and queues develop. DYNASMART (Jayakrishnan *et al*, 1994) also uses speed/density relationships but has adopted a more detailed representation of signalized intersections to model delays at these facilities. In contrast to most existing simulation tools which are time-based, MEZZO (Wilco *et al*, 2006) is an event-based mesoscopic simulator. Events in MEZZO are defined by vehicles entering a link, exiting a link, making a new route choice, etc. Changes in traffic states are calculated only when event happens.

3. RELATED WORK

Integrated use of traffic simulators and computation tools are among the new trends of solving traffic engineering problems. Objectives of integrated traffic simulation include a) studying complex scenarios involving domain aspects modeled by multiple simulators; b) evaluating generic control algorithms; c) improving simulation performances in terms of granularity and computation efficiency. Based on types of integrated tools, we categorize existing integration systems as being homogeneous or heterogeneous. Homogeneous systems integrate simulators which model the same aspect of traffic domain (e.g., roadway network) while heterogeneous systems integrate simulators modeling different domain components (e.g., roadway and marine terminal). A few examples of homogeneous integration are MEZZO-VISSIM (Burghout *et al*, 2007), Paramics-DYNASMART (Sahraoui *et al*, 2005). VISSM-TerSim (Ioannou *et al*, 2007) and TraNs (Piorkowski *et al*, 2007) are examples of heterogeneous system.

MEZZO-VISSIM simulates different areas of a traffic network at different levels of detail. It combines the strengths of meso simulation of large scale networks with less calibration with those of micro simulation of interested areas in greater detail. The integration architecture consists of a new module that contains components shared by both tools: a database with information of the entire network graph, the travel behavior component with route choice models and path generation algorithms. Each time a simulated vehicle makes a route choice, the common module is consulted. Both simulators also update the network database regularly with link conditions in their sub-network.

VISSIM-TerSim allows joint modeling and simulation of traffic in marine ports and traffic networks. It consists of three modules: TermSim, VISSIM and TermCost. An external program is used to execute VISSIM COM commands to access its simulation data, allowing VISSIM to work as an automation server to export objects. TerSim employs a client program to collect data from VISSIM through the server interface and converts it into its inputs. Also TerSim writes its outputs as the corresponding input object of VISSIM. The TermCost is an offline cost model to analyze the simulation data.

TraNS, as a heterogeneous integration platform, combines the capabilities supported by a network simulator ns-2 and those supported by a traffic simulator SUMMO for realistic evaluations of VANETs (vehicular Ad-Hoc networks) applications. TraNS employs the Traffic Control Interface (TraCI) module for interlinking road traffic and network simulators. TraCi module translates the information exchange within VANET simulated by ns-2 into atomic mobility commands and feeds the signal to SUMMO to manipulate the mobility of individual vehicles.

The key difference between our effort and the ones mentioned above is that our framework is not targeted for a single problem and a single class of end users. In the above systems, a fixed set of tools are tightly coupled and reconfiguration of these systems needs extensive reimplementation. Inspired by a wealth of work on model/ontology based tool integration (e.g., Benjamin *et al*, 2007), we adopted Semantic Web and workflow technologies to realize our vision of a modular and extensible traffic simulation framework. A semantic ontology model forms the conceptual basis for data and application integration. A workflow approach is used to handle executions of integrated tools, which is especially attractive as it hides the low-level details of tool integration from domain experts.

Semantic Web has received much attention in the research community, based on which many information integration systems have been proposed recently. The work in Semantic Web area most pertinent to our work are those which focus on data and asset management including integrated asset management in smart oil field (Ram *et al*, 2007) and work on metadata catalogs (Singh *et al*, 2003). While our problem is more complex as the information stored in traffic simulation models is neither as well structured as in databases nor as easily accessible. Integrated simulation requires simulators to exchange data at runtime makes our problem even harder.

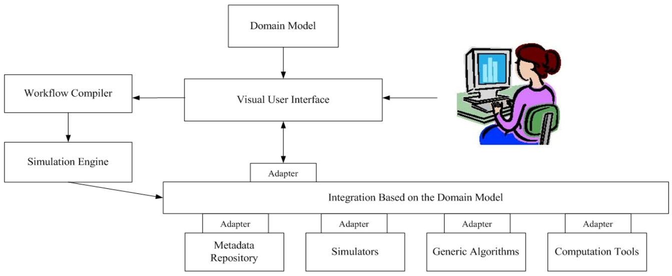


Fig. 1. Architecture of the integration framework

Many application integration issues are addressed extensively using workflow technologies such as in the Kepler (Ludascher *et al*, 2005) and SODIUM systems, which we believe are the closest to our work. Especially, SODIUM accepts high-level domain specific specification of workflows and automatically translates them to executable workflows, which may invoke web-services, grid services or P2P services. Although we do not address the problem of integrating distributed applications or services in our framework, we also use abstract workflows to model the execution and data flow patterns in simulation which provide a high-level view of tool integration to domain users.

The convergence of ideas from the Semantic Web and workflow technologies is novel and we believe it holds great potential for enabling IT solutions to solve modeling and tool/application integration problems in transportation domain. Development of a full scale integration environment capturing all aspects of traffic systems is a long-term and incremental effort. The emphasis of this work is on the development of the fundamental structure and the integration of a selected set of popular simulation tools which will clearly demonstrate advantages of our approaches for integrated traffic modeling and simulation.

4. THE FRAMEWORK OF INTEGRATED TRAFFIC MODELING AND SIMULATION

Our framework for integrated traffic modeling and simulation enables the integration of heterogeneous traffic simulators, generic algorithms and field data into a unified environment. We envision the integration system to be an evolving framework allowing additions, changes of underlying components on a modular basis while maintaining the integrated nature. Some of the key design objectives are:

Single view of information: The framework hides the disparity of data formats, simulation models and tools from end users and provides a single logical space for simulation design.

Generic and reusable architecture: The framework is configurable for a variety of simulation scenarios without the need for extensive redesign or reimplementation. Legacy simulation models can be searched, integrated and re-configured to create new simulations.

Loose coupling of simulation tools: A mediator data schema (ontology model) enables loosely coupled tool integration architecture because all the tools can now read from and write to a shared database. Indirect coupling of disparate tools leads to a modular and highly extensible framework.

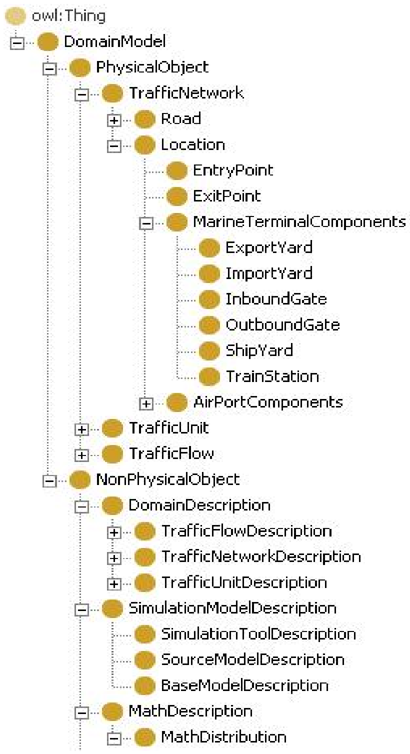


Fig. 2. Domain ontology

4.1 Overview of the Framework Architecture

We adopted a model-based approach to achieve the design objectives. In this approach, a semantic domain model is used as a primary enabler of data and application integration. Fig. 1 shows the architecture of the system (Qunzhi *et al*, 2008). It consists of five major components: the domain ontology model, metadata repository, workflow compiler, simulation engine and user interface. The domain model captures domain concepts and serves as the mediator data schema for integration of heterogeneous data and models. Data disparity is addressed by mapping between generic data/models and the domain ontology. The metadata repository stores information about simulation data and models in a canonical form that will be programmatically accessible to any component (e.g., simulators, algorithms) that wishes to read from or write to this database. Execution and data flows of integrated simulations are modeled as abstract workflows which can be interpreted by the workflow compiler and translated to low-level executable workflows. The simulation engine is a workflow host application which actually handles executions of underlying tools.

4.2 Ontology-based Modeling Integration

Traffic simulation tools employ a wide variety of models (e.g., terminal model, link-connector network model and link-node network model) to capture transportation domain aspects. Most of simulation models are stored in weakly structured formats, maintaining (keeping data and model collections consistent, correct, and up-to-date), understanding, searching and integrating model information are difficult problems which have not been well addressed before.

Semantic Web provides an ontology-based framework that allows data to be shared and reused across model, application and even community boundaries. It has been used for data and application integration in various domains such as health care (Tran *et al*, 2007; Van *et al*, 2008), biology (Taswell *et al*, 2008) and oil industry (Cong *et al*, 2008). An ontology is a collection of key concepts and their inter relationships collectively providing an abstract view of an application domain. It defines a common vocabulary across the system and forms the conceptual basis for integration. Ontology-based system design is especially valuable for the design of complex computer-based systems that involve interactions among applications which are not specifically designed to interact with each other. It requires to formalize relevant domain knowledge as a meta-model which represents a set of conceptual building blocks (entities) and the relationships between the entities.

In our framework, using the ontology language OWL we defined a traffic model which captures domain concepts, their attributes and relationships. OWL is attractive for defining domain models because it is built on open standards, provides the requisite level of expressiveness to capture the rich relationships in the transportation domain, has precisely defined semantics and is accessible through flexible APIs. In general, we classify the domain objects into physical and nonphysical classes. Physical class has subclasses like traffic network, terminal components, traffic units etc. Nonphysical objects include property objects such as traffic network description, routing decisions, ship schedule etc. Domain ontology is an evolving paradigm which can be extended by domain experts to capture new domain concepts to meet integration requirements. As shown in Fig. 2, to allow model a target traffic system involving metropolitan roadway networks and marine terminals, both network components and marine terminal components such as road, ship yard, train station, import/export yard and inbound/outbound gate need to be defined in the domain model. The attributes of domain elements such as length of road, processing rate of inbound gate, capacity of import yard etc are also modeled in the ontology.

Domain experts create integrated simulation models through *model publisher* and *scenario editor* modules of the user interface. The *model publisher* is used to generate and store ontology-based simulation models (base models). To enable integrated modeling, generic models designed using different simulation tools need to be translated to the ontology representations. For each integrated simulators, the publisher implements a model parser which is able to extract information from source simulation models and represent it using the ontology vocabulary. Base models are then stored in the metadata repository for later referencing. Model management including searching, viewing and retrieval are achieved using the SPARQL query language. The *scenario editor* allows domain experts to load models from metadata repository and configure "what-if" simulation scenarios through a unified interface in the context of decision making. A scenario intrinsically represents an integrated simulation model, which is able to capture complex traffic systems by composing models defined using different simulators.

4.3 Workflow-based Application Integration

Simulation of transportation systems captured by integrated models need seamless integration of underlying software applications on a collaborative computing platform. In our vision, the technical requirements for application integration are as follows:

· It can activate and control execution engines without the help of Graphic User Interfaces (GUI) of specific tools, and can advance the simulation according to a given simulation cycle.

· At the intermediate states, which are the states at the end of every simulation cycle, it can access data objects produced by integrated applications.

· It can model and automatically handle interactions such as data exchange pattern and execution logic between integrated applications.

Traditional approaches for traffic simulation integration were accomplished in a point-to-point fashion which is relatively static, inflexible, and is not scalable. Interactions between tools are typically hardcoded for specific simulation runs. Integrating new tools to such a system requires extensive system redesign and programming implementation. In our system, for each tool the framework implements a data adapter which is able to transform generic data produced by the tool into the general representation defined by the domain ontology model. Such data can be pushed to the metadata repository during simulation and is understandable across the system. Simulators and other computation tools read from the shared repository for data exchange. This hub-spoke communication architecture enables a modular and extensible integration framework.

Based on the communication platform, a workflow approach is used to provide flexible control and rapid execution of underlying tools. Workflow technology is an effective method to realize computing process control, process integration and process reconstruction. It has been used in many large-scale complex applications such as structural chemistry experiments (Baldridge *et al*, 2005), biological experiments (Oinn *et al*, 2006), network simulation (Eric *et al*, 2006) etc. Workflow is the depiction of a group of activities organized by control paradigms, declared as the work of a person, work of a simple mechanism or work of a group of machines, software applications etc. It provides the glue for distributed applications and services. In our framework, simulation execution patterns are modeled as abstract workflows using a XML schema (e.g., execution sequence, simulation time step for each simulator and input/output relationships are modeled in the schema). Domain experts can customize the execution and data flow of underlying tools by configuring the workflow paradigm. Configured user workflows will be interpreted by the workflow compiler and translated to low-level workflows which can be invoked by the simulation engine to actually execute the simulations.

Various executable workflow languages/paradigms like BPEL, XScufl, Windows Workflow Foundation (WF) and Kepler have been proposed and successfully used. Among them, many characteristics of the WF make it an attractive choice to be used in our framework. These include the ability to embed workflow engine/runtime in a host application and to control and monitor the workflow status which makes it possible to write advanced workflow management environments which builds on the basic functionality framework. For example, in our case it is possible to embed the simulation engine, which is a WF workflow host application, within the environment used to define the high-level user workflows. The simulation engine is a simple state machine WF workflow application. It consists of four states: initiate simulation, step execution, data exchange and stop simulation and supports interactive execution of multiple simulators. Low-level details of handling execution and data flow are hidden from end users and managed by the workflow runtime.

5. EXAMPLE APPLICATIONS

Management of integrated simulation is complex and tedious especially for domain scientists such as control theorists and has become a barrier to high-quality research. In order to certify the proposed framework is feasible and effective for integrated traffic modeling and simulation, we studied a set of simulation experiments using the framework. Our prototype implementation of the integration system is based on Windows operating system with .NET framework 3.0 and java JDK 6.0. The framework provides intuitive interfaces to integrate tools, define and execute simulations and hence facilitates domain experts focusing on developing their solutions to transportation problems. In the following we use representative examples to highlight the utility of our framework.

5.1 Marine Terminal – Traffic Network Simulation

In the first example, we employ the integration framework to study freight transportation near a marine terminal. The problem is to maintain high terminal throughput while at the same time manage congestion and maintain traffic efficiency in the traffic network outside the terminal. This by itself is very challenging due to multiple factors that affect each mode of freight transportation as well as the complex interactions between different modes of transportation. Neither a terminal model nor a traffic network model can be used alone to evaluate new strategies and make decisions. For instance, a model of a marine port may be used to predict an increase in terminal throughput as a result of a change in cargo handling strategies or introduction of a new technology. However, unless this evaluation includes a simulator that models the entire transportation chain, such an evaluation may easily ignore the fact that improved terminal throughput could cause an increase in congestion on the traffic network used by trucks to serve the terminal. To study this problem, integrated modeling and simulation of the entire transportation chain is needed.

In our experiment, we assume the marine port is modeled using TermSim and the traffic network connected to the terminal is modelled using VISSIM. The major steps taken to study the target freight transportation systems are as following:

Publishing base models: Domain scientists design base terminal models and traffic models using TermSim and VISSIM respectively. These models can be re-configured or modified to represent new concepts or operational/control strategies of the modeled sub-systems. The generic terminal and network models are published to the integration environment, translated to ontology representations and stored in the common repository so that they can be referenced later to compose the freight transportation system.

Integration of TerSim and VISSIM: TerSim and VISSIM are both microscopic, time step simulators. Fig. 3 shows a flow chart of the execution logic of the integrated simulation. In each cycle, the two simulators execute for a given time step and then exchange data. The simulation stops when the total simulation time is reached. As we observed, this type of execution logic is popular in integrated simulations and can be realized by using the four-state state machine WF workflow we discussed in Section 4.3. In our framework, users can merely select an abstract workflow paradigm which can be converted to the WF workflow, configure the simulation time step, input/output relationships of composed terminal and traffic network models (e.g., inbound or outbound gate of the terminal is connected to a specific road of the traffic network) to enable the integrated simulation. Low-level details of tool integration, data conversion etc. are handled by the simulation engine automatically.

Design space exploration: Domain experts can configure and simulate a set of intuitive "what-if" scenarios to evaluate their designs in either terminal or traffic network system. The framework enables users to bind the pre-configured execution workflow to a scenario, load network models and terminal models to the *scenario editor* and customize simulations in an integrated manner. Various terminal operational strategies and local traffic conditions (e.g., increasing the processing rate of inbound/outbound gates, changing the routing decisions of trucks outside the terminal etc.) can be modeled and the impact can be evaluated intuitively.

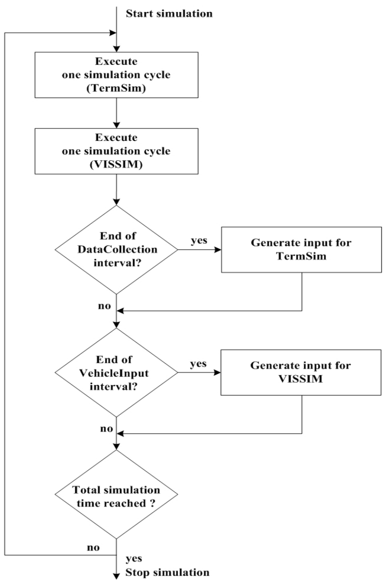
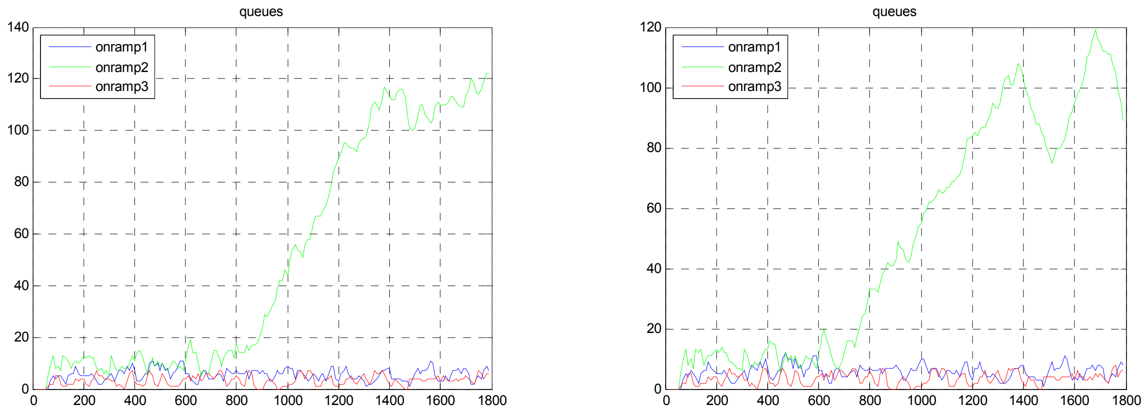


Fig. 3. Control logic of integrated TerSim-VISSIM simulation



(a) (b)

Fig. 4. Ramp queue length with (a) no control and (b) applying roadway controller

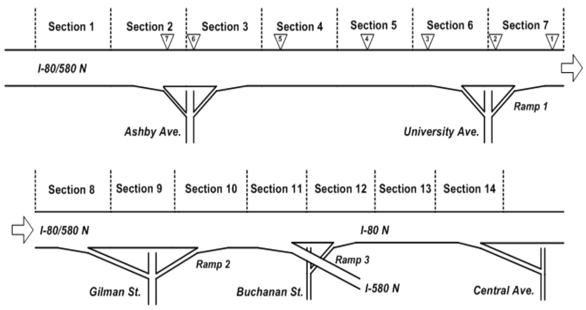


Fig. 5. Layout of the freeway model (Hwan *et al*, 2007)

5.2 Highway Ramp Controller Evaluation

Domain scientists often conduct simulations to evaluate traffic control or optimization algorithms. In the second example, we employ the integration framework to evaluate the roadway controller proposed in (Hwan *et al*, 2007). The control system that integrates ramp metering and speed limits is designed to relieve congestion, suppress shock waves and improve safety on a freeway stretch. The algorithm is implemented using Matlab and the highway is modelled using VISSIM.

In this case, VISSIM is integrated with the Matlab engine. Matlab is one of the most widely used tools for control system design and simulation. Providing the framework support the integration of Matlab with simulation tools is especially useful for transportation research. In this experiment, for each simulation cycle the roadway controller takes the vehicle flow rates as inputs and calculates the metering rates and speed limits for the highway. The execution logic can also be modelled by the four-state state machine workflow. At the step execution state, the simulation engine advances the highway simulation in VISSIM and calls the Matlab algorithm to compute the control input for the next cycle. While at the data exchange state, the simulation engine access the VISSIM kernel functions to collect the vehicle flow rates and load the data to Matlab workspace through the metadata database, and also create signal and speed commands for VISSIM according to the control inputs.

Table 1. Simulation results

scenario	stdK	stdV	TTS1	TTS2	TTS3	TTSA
1	7.4	26.2	295.6	99.0	62.8	457.4
2	7.1	26.1	298.8	99.0	55.3	453.1

Using the simulation framework, we followed the three steps described in the first example to evaluate the performances of the roadway controller. Assuming there is a disturbance on the freeway (see Fig. 5) at section 11 which is near to ramp 2, two scenarios are simulated with 1) no control; 2) applying the ramp metering and speed controller. Fig. 4 shows the queue length at ramp 2 for both cases and Table 1 shows the simulation results. The total travel time (TTSA) for case 2 is less than that in case 1 and the average queue length at ramp 2 for case 2 is smaller than that in case 1. From the results we can conclude the controller reduces the freeway congestion. The impact of the roadway control is not significant in this case, the simulation framework facilitates to optimize the algorithm by interactively configuring the roadway controller and evaluating the simulation results.

Except providing a simplified integration process and an integrated design space exploration environment, a crucial advantage of the integration framework is that it is an evolving framework which allows domain scientists insert new tools into the system and enable the integration with existed modules.

6. CONCLUSION

In this paper, we discussed a semantic framework for integrated traffic modeling and simulation. The key components of the system include the semantic domain ontology, shared metadata repository and workflow-based simulation engine. The domain model provides a common vocabulary across the system and forms the conceptual basis for data and application integration. The metadata repository enables model addressing and the hub-spoke data exchange mechanism which reduces the complexity of passing heterogeneous data among traffic simulation tools. The workflow based simulation engine enables rapid prototyping and execution of simulations. Compared to traditional solutions for integrated traffic simulation, the proposed framework is more flexible and extensible. It can realize tool integration in great ease and provides intuitive interfaces for design space exploration.

ACKNOWLEDGEMENT

This work was supported by the Metrans Center at the University of Southern California, Los Angeles, and its funding agencies, the US Department of Transportation and California Department of Transportation (Caltrans).

REFERENCES

Arnaud Doniec, Stéphane Espié, René Mandiau and Sylvain Piechowiak, Multi-agent Coordination and Anticipation Model to Design a Road Traffic Simulation Tool, *4th European Workshop on Multi-Agent Systems*, 2006.

Baldridge K.K., Greenberg J.P., Sudholt W., Mock S., et al, The Computational Chemistry Prototyping Environment, *Proceedings of the IEEE*, Volume 93, Issue 3, 2005.

Ben-Akiva, M., et al., Simulation of Traffic Demand-supply Interactions within DynaMIT, *Transportation and network analysis: current trends*, 2002.

Benjamin, P., Akella, et al, Using Ontologies for Simulation Integration, Winter Simulation Conference, 2007.

Buisson C., Lebaque J.P., et al, A Discretized Macroscopic Model of Vehicular Traffic Flow in Complex Networks Based on the Godunov Scheme, *CESE'96*, 1996.

Burghout W. et al, Hybrid Traffic Simulation with Adaptive Signal Control, *Transportation Research Record*, 2007.

Byungkyu Park, Hongtu Qi, Microscopic Simulation Model Calibration and Validation for Freeway Work Zone Network - a Case Study of VISSIM, *IEEE Intelligent Transportation Systems Conference,* 2006.

Cai Yuanli, et al, On the Computer Simulation for Traffic Systems, *Southern African Transport Conference.*, 2004.

Calvanese, D., De Giacomo G., Lenzerini M., et al, Information Integration: Conceptual Modeling and Reasoning Support, 3rd *IFCIS Intl. Conf. on Cooperative Information Systems*, 1998.

Cong Z., V. K. Prasanna, et al, Data Component Based Management of Reservoir Simulation Models, *IEEE Intl. Conf. on Information Reuse and Integration*, 2008.

CORSIM, http://mctrans.ce.ufl.edu/featured/tsis

Duncan G., Littlejohn J.K., High Performance Microscopic Simulation for Traffic Forecasting, *IEE Colloquium on Strategic Control of Inter-Urban Road Networks*, 1997.

Elloumi N., Haj-Salem H. et al, METACOR: A Macroscopic Modeling Tool for Urban Corridor, *Proceedings of Triennial Symposium on Transport Analysis*, 1994.

Eric Eide, Leigh Stoller, et al, Integrated Scientific Workflow Management for the Emulab Network Testbed, *USENIX Annual Technical Conference*, 2006.

Fernandez B., Jesualdo T. et al, A Semantic Web-based System for Managing Clinical Archetypes, *IEEE 30th Annual International Conference of the Engineering in Medicine and Biology Society*, 2008.

Hwan Chang, Yun Wang and Ioannou P., An Integrated Roadway Controller and its Evaluation by Microscopic Simulator VISSIM, *Proceedings of ECC'07*, 2007.

Ioannou P., Chassiakos A., Valencia G. and Hwan C., Simulation Test-bed and Evaluation of Truck Movement Concepts on Terminal Efficiency and Traffic Flow, *Metrans Project 05-11 Final Report*, 2007.

Jayakrishnan, R., et al, An Evaluation Tool for Advanced Traffic Information and Management Systems in Urban Metworks. *Transportation Research C*, 1994.

Leonard D.R., Power P, et al, CONTRAM: Structure of The Model, *Crowthorn TRL Report RR 178*, 1989.

Ludascher B., Lin K., et al, Managing Scientific Data: From Data Integration to Scientific Workflows, *GSA Today, Special Issue on Geoinformatics*, 2005.

Oinn T., Greenwood M., Addis M., Alpdemir N. et al, Taverna: Lessons in Creating a Workflow Environment for the Life Sciences, *Concurrency and Computation: Practice and Experience*, Vol. 18, No. 10, 2006.

Owen, L.E., et al, Traffic flow simulation using CORSIM, *2000 Winter Simulation Conference*, 2000

Qunzhi Zhou, A. Bakshi, V. K. Prasanna et al, Towards an Integrated Modeling and Simulation Framework for Freight Transportation in metropolitan Areas, *IEEE Intl. Conf. on Information Reuse and Integration*, 2008.

Piorkowski M., et al, TraNS: Joint Traffic and Network Simulator for VANETs, *13th ACM MobiCom*, 2007.

Ramakrishna S., V. K. Prasanna, A Semantic Framework for Integrated Asset Management in Smart Oilfields, *IEEE Intl. Symp. on Cluster Computing and the Grid*, 2007.

Sahraoui A., Jayakrishnan R., Microscopic-macroscopic Models Systems Integration: a Simulation Case Study for *ATMIS, Simulation,* 2005.

Singh G., Bharathi S., Chervenak A., et al, A Metadata Catalog Service for Data Intensive Applications. *ACM/IEEE conference on Supercomputing*, 2003.

Taswell C., DOORS to the Semantic Web and Grid With a PORTAL for Biomedical Computing, *IEEE Trans. on Information Technology in Biomedicine*, Vol. 12, Issue 2, March 2008.

Tran Quoc Dung, Kameyama, W., A Proposal of Ontology-based Health Care Information Extraction System: VnHIES, *IEEE Intl Conf. on Research, Innovation and Vision for the Future*, 2007.

Tsalgatidou A., Athanasopoulos G., Pantazoglou M., et al, Developing Scientific Workflows from Heterogeneous Services, *ACM SIGMOD Record*, vol. 35, 2006.

Van Niekerk J.C., Griffiths K., Advancing Health Care Management with the Semantic Web, *3rd International Conf. on Broadband Communications, Information Technology & Biomedical Applications,* 2008.

VISSIM, http://www.english.ptv.de/cgi-bin/traffic/traf_vissim.pl

Wache H., Vogele T., Visser U., et al, Ontology-based integration of information: a survey of existing approaches, *Proceedings of IJCAI-01 Workshop: Ontologies and Information Sharing,* 2001.

Wilco Burghout, Haris N. Koutsopoulos and Ingmar Andreasson, A Discrete-Event Mesoscopic Traffic Simulation Model for Hybrid Traffic simulation, *IEEE Intelligent Transportation Conference,* 2006