Dynamic Data Driven Application Simulation of Surface Transportation Systems

R. Fujimoto, R. Guensler, M. Hunter, H.-K. Kim, J. Lee, J. Leonard II,
M. Palekar, K. Schwan, and B. Seshasayee

Georgia Institute of Technology, Atlanta, GA 30332, USA
{fujimoto@cc, randall.guensler@ce, michael.hunter@ce,
john.leonard@ce, schwan@cc}.gatech.edu

Abstract. A project concerned with applying Dynamic Data Driven Application Simulations (DDDAS) to monitor and manage surface transportation systems is described. Building upon activities such as the Vehicle-Infrastructure Integration initiative, a hierarchical DDDAS architecture is presented that includes coupled in-vehicle, roadside, and traffic management center simulations. The overall architecture is described as well as current work to implement and evaluate the effectiveness of this approach for a portion the Atlanta metropolitan area in the context of a hypothesized emergency evacuation scenario.

1 Introduction

The complexities associated with managing future transportation systems will predominantly reside in the interaction between sensing systems, data acquisition systems, demand models, communications, and traffic simulations. Systems soon to be launched under the Vehicle-Infrastructure Integration (VII) initiative will deploy a variety of roadside and mobile sensing platforms capable of collecting and transmitting transportation data [1-3]. Old paradigms associated with the use of basic information to identify incidents and mitigate recurrent congestion will fall by the wayside as new technologies come online. Centralized locations, local area hubs, and individual vehicles will receive fixed sensor and mobile vehicle data to collaboratively monitor and evaluate system performance, perform simulation runs, and plan routes through the system. Such systems will be interactive, iterative, and adaptive. As these systems evolve, they can be increasingly used in developing responses to rare or non-recurring conditions such as those occurring in emergency response scenarios.

As sensors proliferate, the amount of transportation-related data will increase by orders of magnitude. Yet, communications bandwidth and computer processing capabilities will be limited by availability and cost. Effective and efficient system management will require real-time determinations as to which data should be monitored, and at what resolutions. Distributed computing can help mitigate system limitations. Data collection, data processing, data analysis, and simulations performed by system agents (sub-network monitoring systems, base stations, vehicles, etc.) will lessen communication bandwidth requirements and harness surplus computing capacity. Middleware to manage the distributed network, synchronize data and results among autonomous agents, and resolve simulation output conflicts between agents

using disparate data sets become critical activities in such a system. With proper feedback, the connections between autonomous agent simulations will allow each simulation to adapt to better model reality.

2   A Motivating Example

The following scenario illustrates how the envisioned distributed traffic simulation system would operate in detecting and responding to an emergency. Suppose that a large explosion of unknown origin occurs. The vehicle-to-vehicle (V2V) sensor network normally used to transmit data for traffic operations management observes the loss of communication among multiple vehicles and roadside devices. Base stations on the arterial network (inside traffic controller cabinets) normally used to compile data from fixed detectors and instrumented vehicles and to simulate vehicle activity and traffic flow responses to changes in signal timing plans, identify the loss of communications among the devices within the local vehicular network. Instrumented vehicles, roadside sensors, and roadside data waypoints unaffected by the blast begin to record a significant disruption in traffic flow (decreases in speed, reductions in through volumes, etc.), reporting these data to local roadside nodes and to the regional traffic operations center (TOC). In response to the loss of sensors and controllers, the remaining upstream and downstream nodes automatically reconfigure to operate without the data from the affected vehicles and sensors, and without the simulation outputs of the affected nodes. Automated GIS routines at the TOC identify the location of the explosion based upon the data outage and transmit detailed site hazard information to the emergency management center (EMC).

The roadside base stations, in-vehicle computing systems, and other computers collectively process available network and agent data to establish an initial event boundary for use in emergency response planning. This facilitates the implementation of any pre-planned response initiatives (i.e. the initial deployment of emergency response personnel). The sensor network and distributed simulation system also allows for the implementation of a dynamic emergency response plan. In a dynamic response scenario, roadway base stations and in-vehicle systems (real-time V2V and fixed sensor data) identify immobile vehicles, recalculate roadway capacities, and simulate local traffic after the event. In-vehicle, local area and TOC based simulations are utilized to monitor traffic operations, forecast traffic flow, and derive an evacuation plan designed to clear vehicles from the impacted area, direct inbound traffic away from the impacted area, and provide inbound priority to emergency response vehicles. Navigation computers in vehicles outside the zone receiving data via the V2V and roadside network and are instructed to plan paths around the traffic impact zone, using data that are continually repackaged and distributed by the system to optimize zone clearance. Within minutes of the blast event, multiple sensor networks will have transformed monitoring data into useful information for emergency response planners and area evacuations.

We envision that systems will be composed of a heterogeneous collection of in-vehicle, roadside, and traditional computation and sensor nodes (e.g., TOC/EMC-based servers) that must analyze current system states, predict future states, and rapidly adapt to unexpected disruptive events on short time scales. These systems

will be constrained by available communications bandwidth, data transfer costs, and distributed computing processing capacity.

This project is developing adaptive techniques to meet the challenging QoS requirements of transportation systems, providing capabilities such as in-flight data processing, continual awareness of end user needs, and dynamic adaptation to platform changes. Modeling and simulation research is focused on developing real-time, agent-based, data driven simulation models to be deployed within the transportation infrastructure. These models are being designed so that they can be replicated and integrated with other models through a distributed simulation framework, as well as interact with centralized, larger-scale models running in traffic management centers on high performance computing platforms. Finally, we envision multiple layers of autonomous on-line simulations operating in vehicles, signal controllers, and traffic management centers designed for a variety of purposes and sharing state predictions and model results.

3   System Architecture

As sensor networks continue to proliferate communication systems will evolve to the point where the detailed traffic data from these sensor networks become ubiquitously available throughout the network. This data will facilitate real-time fixed location and in-vehicle simulation. A notional diagram of the envisioned system is depicted in Figure 1. The hierarchical architecture includes sensors, in-vehicle computer systems and other elements as end nodes. Intermediate nodes include computer systems housed in traffic signal controllers and other roadside equipment. Regional traffic management centers (TMCs) form the highest level of the hierarchy.

A major improvement in traffic monitoring is derived from the integration of data from onboard vehicle sensors. The high-resolution data provided from instrumented vehicles allows researchers to examine minute changes in real-time traffic flow and regional stability of traffic performance characteristics over time. Each vehicle will simulate its region of interest, downloading as needed data from roadside sensors, vehicle sensors, and traffic controllers that act as servers for local vehicular simulations. Each vehicle downloads the map of its regions of interest from the server as it travels along the roadway. It will cache this map for future use. It obtains information such as location, speed, and direction of travel of other vehicles in its surrounding regions through in-vehicle sensors, roadside servers, and potentially neighboring vehicles in real-time. Aggregate data such as vehicle density and flow rates from other regions of interest may also be obtained through the server. Based on this data, future projections are made using simulation. This projected data might be used to make routing decisions, for example. As new information is obtained, simulation parameters or state variables are updated and projected future states are modified accordingly. Results of the simulation are continuously uploaded to the server. Many vehicles may be simulating the same region and updating information to the server. The server aggregates the information obtained from roadside sensors as well as simulation results in order to provide an accurate picture of the region.
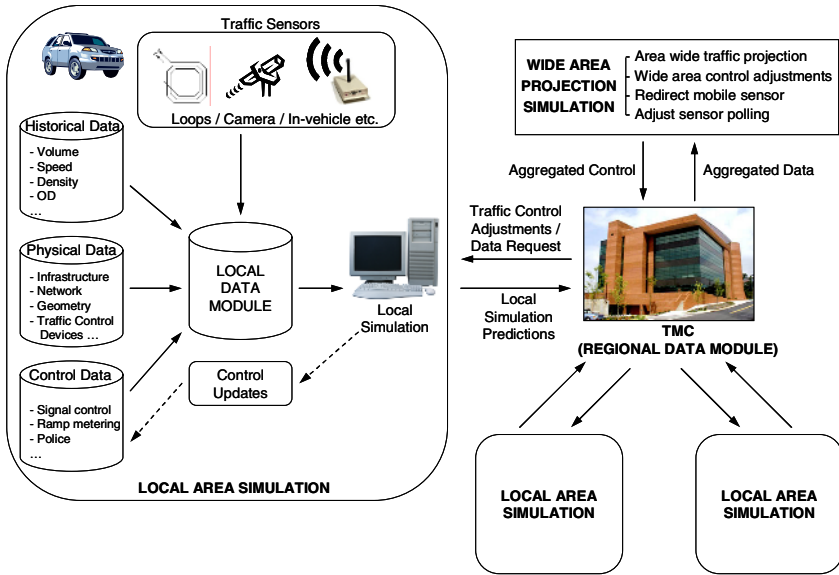
Fig. 1. Notional diagram of a dynamic, data driven intelligent transportation system

As communications-equipped vehicles continue to proliferate, their data will be extremely useful in monitoring real-time roadway operating conditions. A primary limitation encountered in integrating their data into the transportation monitoring network will be communications bandwidth. In addition, it is unlikely that centralized simulation and planning operations, such as those currently operated in regional traffic operations centers, will be able to integrate sufficient processing capacity to handle all of the potential data available within the transportation system, even if sufficient communications bandwidth were available. Thus, the types of systems being deployed as part of VII give rise to the need for a multi-level simulation architecture. Simulations will operate in at least three levels. In-vehicle simulations will typically be used for purposes such as route planning or travel time prediction. Traffic controller simulations will be used for purposes such as adjusting signal timings. The TMC simulation simulates larger regions, and can be used for city-wide operations planning and traffic monitoring.

We term systems such as these *ad hoc distributed simulations* because they are composed of loosely coupled, largely autonomous simulation, each modeling a different portion of the overall system, often at different resolutions. Unlike conventional distributed simulations that are created by partitioning the system into non-overlapping pieces (e.g., entities or agents), the simulators making up an ad hoc distributed simulation may model overlapping regions. Conflicts must be resolved. Further, if a simulation wishes to model new portions of the road network, it may obtain future state projections from other nearby simulators rather than recomputing this itself. A key research question that is being explored concerns the extent to which these ad hoc distributed simulations can collectively project accurate future states of the system.
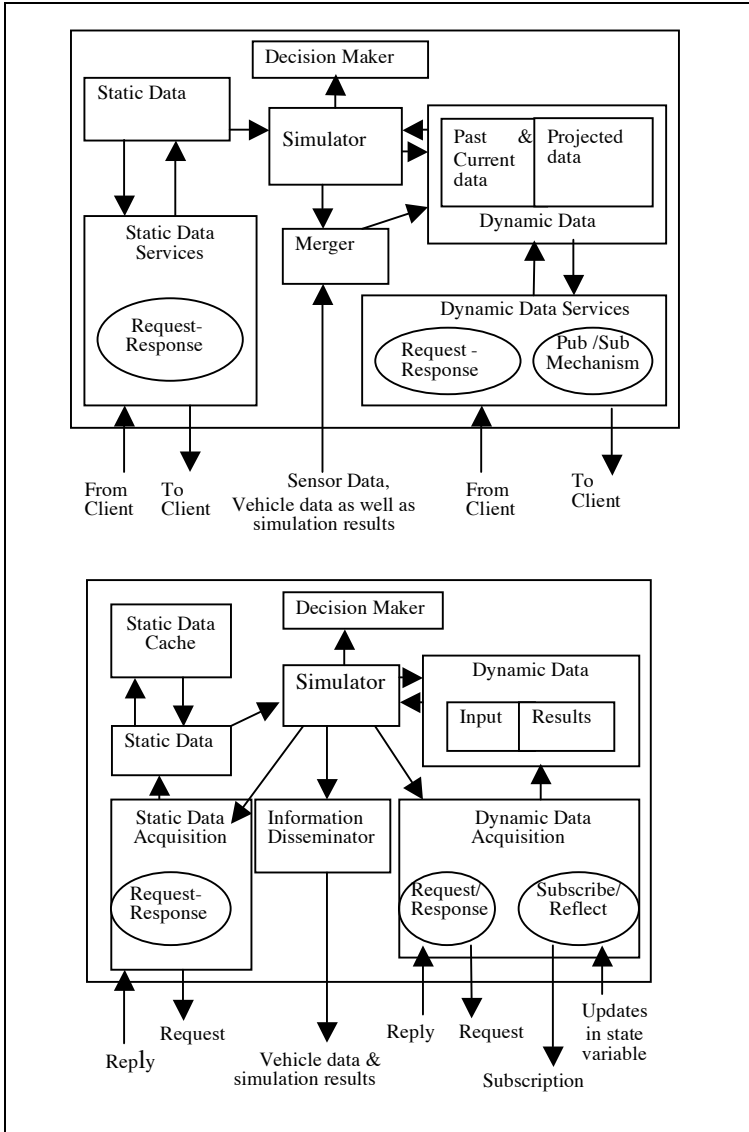
Fig. 2. Software components of (a) server (top) and (b) client (bottom)

The ad hoc distributed simulations considered here use a client-server architecture to coordinate amongst themselves. Traffic controller and TMC simulations act as servers in this framework, and propagate information up and down the hierarchy to obtain a global view of the entire system.

The communication problems arising out of mobility in a cooperative setting bear some similarity to those occurring in the robotics domain. Heterogeneous robot teams cooperate in mapping and exploring unfamiliar terrain [4], as well as in path planning

[5]. Due to the high failure of robots [6], as well as the need to conserve power, many adaptive algorithms have been proposed [7]. Such adaptive algorithms can find use in our efforts. Though the objectives are different, the middleware in both the domains has the common feature, in that it needs to adapt to a dynamic environment, and tolerate failure, to provide a reliable multi-node communication platform to the higher layers. Similar efforts are also being employed in the area of enterprise computing [8] where the middleware can detect dynamic behavior in the environment and automatically reconfigure to achieve the application's quality requirements.

A server (e.g., a simulator in a traffic signal controller or the TMC) in the mobile system architecture is depicted in Figure 2 (top). Static data includes information such as the road network that remains constant for long periods of time. Static data services are provided to clients through a request/response protocol. Dynamic data such as vehicle densities and flow rate continuously changes over time. Dynamic data can include both current and projected values obtained via simulations. Current values are obtained from sensors and clients. These values are used to calibrate the simulations and also predict future system states.

Projected data is used for decision making. This data is stored in the form of Time-Space Memory, i.e., values of state variables over different time values. Dynamic Data Services provide access to this data. Multiple simulations may simulate overlapping regions; the merger module is responsible for resolving inconsistencies and conflicts. The simulator is a server that predicts future system states. Results of the simulation are used for decision making. For example, if the server is present in the traffic controller then results may be used for signal time optimization.

The client architecture (Figure 2) includes similar components. Static and dynamic data is obtained from servers, and stored in a local cache for future use. Dynamic data can be broadly classified as input to the simulator and simulation results. Input data includes current values of state variables and is used to calibrate the simulation. Results of the simulation include the projected state. Vehicle data such as location, speed, route, etc., as well as simulation results are broadcasted by the vehicle to update the state variables at the server. The individual simulations incorporate live data into the simulations as updates occur. Model parameters must be changed depending on the real time data in order to accurately model the traffic system.

4   Transportation Simulation Models

Throughout the duration of an emergency situation high-resolution mobile, in-vehicle simulations may be utilized to help observe, control, and manage traffic operation in a local area. Two major distinctions are readily apparent between mobile, in-vehicle simulation and traditional transportation simulations. First, an in-vehicle simulation must be able to process a dynamically changing roadway topology. Throughout the duration of the simulation the roadway topology must be updated to represent the area focused around the vehicle. Current transportation simulation models assume a fixed roadway topology, the exception primarily being the modeling of road closures. Second, to be useful an in-vehicle simulation must have the ability to adapt to real-time inputs (volumes, signal indications, etc.), a feature uncommon in current transportation simulations. For example, an emergency scenario represents a non-recurring situation

with potentially dramatic changes in traffic patterns and traveler behavior. Facility operations (e.g. traffic signals, ramp meters), pedestrian movements, changing traveler destinations, police traffic control, etc. all will impact operations in a real-time manner. Even where potential emergencies are modeled well in advance it is highly unlikely that reliable predictive models will be obtained, as people may not follow the evacuation plan, pedestrians may not observe traffic controls, vehicles may not stay within lane lines, and off-road operation may occur.

In addition to the in-vehicle simulation models we are also utilizing existing microscopic simulation platforms to develop several models for combined and complementary experimental analysis. The first model is a CORSIM simulation that incorporates the northern half of the Atlanta area. (CORSIM – short for corridor simulation - is a microscopic, stochastic traffic simulation model developed by the Federal Highway Administration for modeling individual vehicle movements on a second-by-second basis for the purpose of assessing the traffic performance of highway and street systems.) The study area lies in three counties (Fulton, DeKalk, and Cobb) within the Atlanta metropolitan area. The freeway network within the study area includes 8 interchanges between freeways and 80 on/off ramps. The current model includes approximately 1,214 miles of surface streets and 289 miles of freeway.

Data requirements for a model of this size are significant, including network geometry (link length, free flow speed, grade, number of lanes, turn-bay et al.), traffic control (sign, signal control plan), and traffic flow data. The data for the CORSIM model was drawn primarily from the Atlanta Regional Commission's greater Atlanta travel demand model (number of lanes, link lengths, free flow speeds, the presence of an HOV lanes, turn movement ratios, and traffic flows, etc.), USGS one-foot resolution aerial photos (precise lane configurations, turn-bay presence and length, roadway curvature, on/off-ramp configurations, length of acceleration/deceleration lane, etc.) and state and local government agencies (primarily signal control and volume data). The current CORSIM network contains 3,227 nodes (1,940 surface nodes and 1,287 freeway nodes) and 5,974 links (4,764 surface links and 1,210 freeway links). A second model being developed covers a significantly smaller geographic area, centering on the Georgia Institute of Technology. This model is being developed at higher resolution, incorporating unsignalized intersections and major traffic origins/destinations. The Georgia Institute of Technology centered models will be utilized to support initial simulation and field experiments. Three separate simulation packages (CORSIM, VISSIM, and SimTraffic) are being coded, allowing for the exploration of potential bias introduced by selected simulations.

5   Project Status

At the time of this writing, implementations of three simulation models representing different levels of the simulation hierarchy (in-vehicle simulation, a regional Georgia Tech model, and a larger Atlanta area model) have been created. Further, runtime infrastructure middleware used for coupling the simulators has also been developed based on the client-server architecture described earlier, and an initial distributed simulation utilizing multiple in-vehicle simulations has been created. Middleware termed IFLOW offering data management functions such as processing of streaming

vehicle data and Quality of Service support has been developed. Other research is examining deployment of test systems in vehicles using mobile laptop computers and wireless communication infrastructure on the Georgia Tech campus.

References

1. Werner, J., Details of the VII Initiative 'Work in Progress' Provided at Public Meeting. 2005.
2. Bechler, M., W.J. Franz, and L. Wolf. Mobile Internet Access in FleetNet. in KiVS 2003. 2003.
3. Werner, J., USDOT Outlines the New VII Initiative at the 2004 TRB Annual Meeting. 2004.
4. Robert Grabowski, Luis E. Navarro-Serment, Christiaan J.J. Paredis, Pradeep K. Khosla, Heterogeneous Teams of Modular Robots for Mapping and Exploration, Autonomous Robots, Volume 8, Issue 3, Jun 2000, Pages 293 - 308
5. David Silver. Cooperative path-planning. In AI Programming Wisdom, 2006.
6. J. Carlson and R. Murphy. How UGVs physically fail in the field. IEEE Transactions on Robotics, Volume 21, Issue 3, Jun 2005.
7. Keith J. O'Hara, Ripal Nathuji, Himanshu Raj, Karsten Schwan and Tucker Balch. AutoPower: Toward Energy-Aware Software Systems for Distributed Mobile Robots. IEEE International Conference on Robotics and Automation. 2006.
8. Karsten Schwan et al. Autonomic Information Flows. Technical Report GIT-CERCS-05-22. Georgia Insitute of Technology. 2005.