**Integrating Traffic Models with Real-Time Data**

Rob Hranac
Berkeley Transportation Systems, Inc.
November 6th, 2009

## Overview

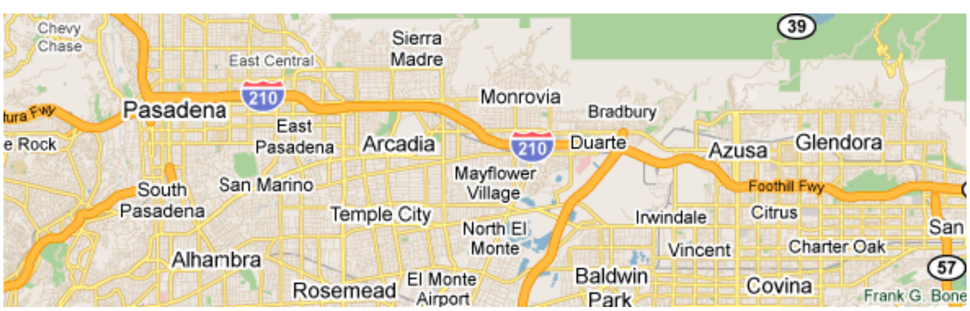
Traffic microsimulation models are an increasingly used tool to support large scale planning efforts, such as corridor analysis. However, developing these model is a non-trivial undertaking in terms of time and cost. These models are quite complex and require a great deal of effort and expertise to develop. Furthermore, these models are static and fragile. Once developed, they rapidly become obsolete as the underlying network and traffic dynamics evolve over time. Maintaining them is a major undertaking and one that is frequently ignored by model owners.

This paper proposes a new paradigm for integrating these models into agency practice. Rather than creating a system that is created one time, in isolation, the work described here combined a real-time data system with an existing traffic model, creating a living traffic model that adapts to changes in the transportation system. This type of modeling system allows agency staff to integrate modeling into their business processes by dramatically lowering the barriers of use of the models.

## Approach

The crux of this approach is to integrate a traffic model with a data system. This has been a recurring theme during the history of more modern traffic modeling software - such as dynamic assignment models - but so far, this effort has mostly met with failure. The thesis of this research is that this failure has mostly stemmed from the fact that most efforts have underestimated the complexity of developing a robust data system. Because of this, the data systems have tended to be fragile, and incapable of supporting a substantial modeling framework. This effort makes the opposite assumption: the key to a functional system is not the modeling software, it is the data system. This effort merges an existing traffic model onto a data system.

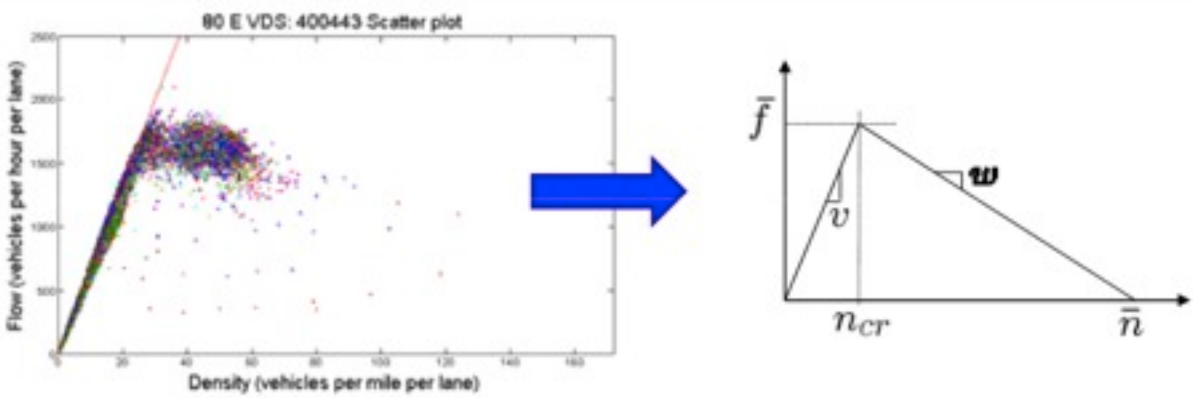
The test network is a California freeway with excellent data coverage: I-210 Westbound, near Los Angeles. The system combines an existing set of modeling tools (TOPL, described below) with an existing set of data tools (PeMS, described below). Preliminary results of this modeling effort are described thereafter. The section of I-210 used for the modeling integration is shown below:



**Traffic Model**

Aurora Road Network Modeler (RNM) is a set of tools for operational planning and management of travel corridors (road networks comprised of freeways and surrounding urban arterials) and analysis of their performance. It is based on the Aurora Object-Oriented Framework for Simulation and Analysis of Flow Networks and is implemented in Java. It contains three core systems, which are defined below: simulator; configurator; and importer.

*Simulator* is a graphical application with interactive user interface that runs traffic simulations on road networks. Traffic flow and density are computed using the macroscopic Cell Transmission Model. It also computes such traffic characteristics as speed, VMT, VHT, delay and productivity loss and generates flow, density and speed contour plots making it convenient to compare with the real traffic data collected by systems (such as PeMS). The user can create simulation scenarios by means of events that change user-specified configuration parameters of the road network at user-specified times. An example of the traffic dynamics model that drives this system is shown below. The model is calibrated using the version of the fundamental diagram shown below:




*Configurator* is a graphical application with interactive user interface whose purpose is to produce XML configuration files for the Simulator. It can be used to build road networks from scratch; edit existing road networks by deleting or reassigning network components or adding new network components; efficiently provision road parameters such as fundamental diagrams and split ratios; and input demand profiles.

*GIS Importer* is a graphical application with interactive user interface that extracts road information from the GIS .shp and .dbf files and saves it in the XML format of Aurora configuration file. This configuration file can be refined in the Configurator and then used by the Simulator. It is recommended to use GIS Importer together with the open source OpenJUMP software used for geometric filtering of GIS data.

**Data System**

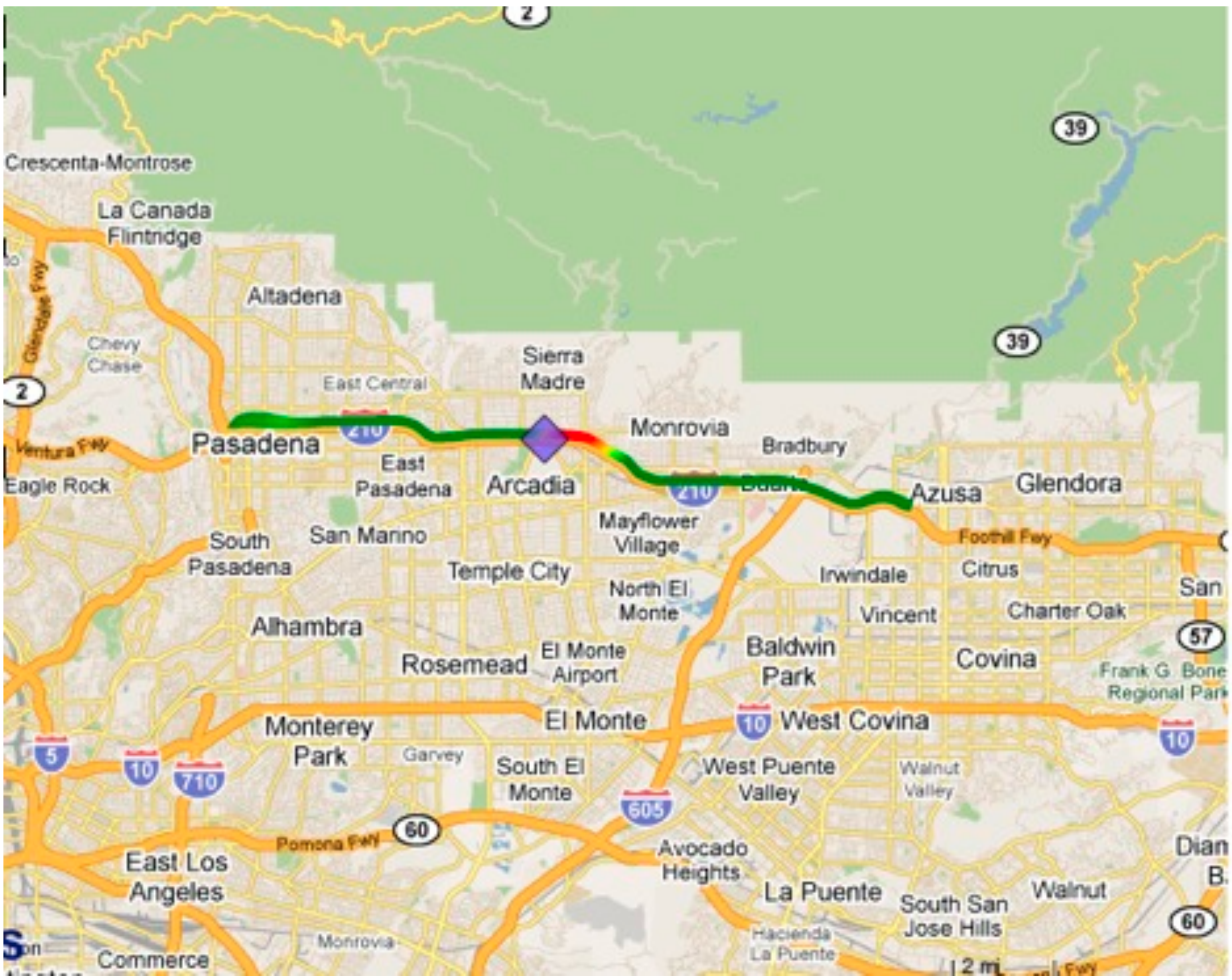
The Performance Measurement System (PeMS) is a real-time data system developed originally for Caltrans. PeMS is a traffic data collection, processing and analysis tool to assist traffic engineers in assessing the performance of the freeway system. PeMS extracts information from real-time intelligent transportation systems (ITS) data, saves it permanently in a data warehouse, and presents this information in various forms to managers, traffic engineers, planners, freeway users, researchers, and traveler information service providers. With PeMS, managers can instantaneously obtain a uniform, and comprehensive assessment of the performance of their freeways. Traffic engineers can base their operational decisions on knowledge of the current state of the freeway network. Planners can determine whether congestion bottlenecks can be alleviated by improving operations or by minor capital improvements. Traffic control equipment (ramp-metering and changeable message signs) can be optimally placed and evaluated. In brief, PeMS helps a wide variety of agency staff make better decisions about investments in their transportation system, by leveraging ITS data they already collect.

In California, PeMS takes in data statewide from a variety of ITS data sources, including over 25,000 traffic detectors of all types (loops, radar, etc.) and several dozen toll tag readers. This data originates in thousands of individual hardware sensors deployed on the freeway system, connected to traffic management centers via a wide variety of communications systems (e.g. cellular data networks, fiber optics, etc.). This data is then transferred from these district traffic management centers through the Caltrans wide area network (WAN) to a centralized set of PeMS servers. The data is stored in PeMS databases as it is received. PeMS software then performs a variety of quality control, measurement, and aggregation functions on this raw data received from the field. In terms of quality control, PeMS routines detect erroneous or missing data and impute values for this data based on a variety of validated statistical techniques. In terms of measurement, PeMS uses this raw data to calculate a variety of higher level measures, such as the location of bottlenecks. In terms of aggregation, PeMS routines follow standard data warehousing models of storing data at multiple levels of temporal resolution, in order to facilitate swift retrieval and processing. All of this processing happens on each batch of received data within a matter of seconds. Users can then access PeMS over the Internet using a secure, web-based interface from any computer with a web browser. The PeMS software architecture is modular and open; it uses a mix of commercial off-the-shelf products and open source systems for integration, computation, and display.

Caltrans has over 25,000 lane detectors deployed on the freeway system. The extensive detection system developed and deployed by Caltrans is the cornerstone of many ITS functions and freeway management strategies, including ramp metering, the display of dynamic messages on changeable message signs (CMS), incident detection strategies, and monitoring the quantity and quality of traffic through the transportation system. The information from these detectors is also distributed in real-time to over 15 Value Added Resellers (VARs) who then package and sell the information to the traveling public. Maintaining a detector system of this size is a massive, complex task. Two tasks in this project enhanced the Caltrans performance measurement systems to add more detailed diagnostics in order to isolate detector failures. This task also added additional sets of interfaces to assist asset management staff in prioritizing and repairing these detectors, once faulty detectors were located.

**Results**

The initial results of the calibrated, real-time integrated model are promising. The model itself faithfully replicates both recurrent and non-recurrent future conditions through a combination of parametric and non-parametric models. Initial results are promising: automated calibration appears to work correctly and back-testing with actual incident data continues. We will continue to test the veracity of the model with quantitative data and hope to be able to present a set of results that. The results of the model are shown visually, below. In this instance, the model is predicting the results of a severe incident (shown as a blue diamond) and the resultant upstream queueing (shown in red):




**Conclusions**

Based on the initial work completed here, several conclusions can be drawn. First, integrating sophisticated traffic models with real-time data is an entirely tractable problem. Where other efforts have stumbled, due to inattention to the data system, when care is given to attach a model to a mature data system, integration is entirely possible.

Secondly, data quality is critical. The data system must have the ability to identify bad data reads (up to a quarter of the data in a standard system) and impute missing data. Without this ability, it is simply impossible to automatically calibrate a model in any practical deployment scenario.

Finally, not all models are equally amenable to this approach. The TOPL model was developed with PeMS integration in mind. Thus, the major calibration parameters of the TOPL model - in essence, a simplified version of the fundamental traffic dynamics diagram for each roadway segment - are tailored to conform to data that PeMS already calculates for each roadway segment. Thus, the model can be automatically calibrated using a reasonable set of already existing data.

Finally, it is worth noting that this type of integration has the ability to radically alter the ability of agency staff to model scenarios by dramatically reducing the cost to individual agency staff in terms of both budget and time. By delivering these results in an online system, this type of approach has the potential to liberate models from the desktops of modelers and put them in the hands of a much broader cross section of agency staff.