A Vehicular Traffic GIS and Simulator for Route Guidance on NY/NJ Highways

J. Holtzman, J. Hui, N. Moayeri, I. Seskar, H. Varma, J. Yip[2]
S. Maric[3], T. Williams[4]
Dept. of ECE, Rutgers Univ., Piscataway, NJ 08855, USA.
Phone (908)-932-5256, Fax (908)-932-5313, jhui@winlab1.rutgers.edu

Abstract

This paper describes the design of a macroscopic traffic simulator using Geographical Information System (GIS) and its implementation for performance evaluation of different IVHS route guidance strategies. The GIS and the associated routing algorithms are intended for deployment in the NY/NJ metropolitan area by TRANSCOM. The GIS allows flexible display of a host of traffic parameters and selection of control actions. Besides GIS and control algorithms, a simulator is implemented to replace the traffic measurements for secondary road network (or even the main traffic arteries in the early phase of the IVHS implementation). The role of the simulator as a tool for designing different control algorithms is also mentioned. The simulator operates by iteratively updating all the densities on all the included roads. The relations for each road are based on well known vehicular traffic models (such as Greenshield's). The dependence between the traffic parameters on different roads is incorporated via relations between a road's traffic parameters and those of its entrance and exit roads.

1.0 Introduction

The current research is conducted to support TRANSCOM's incident detection and congestion monitoring project based on the Electronic Toll and Traffic Management (ETTM)[5] technology in the New Jersey and Staten Island corridor (figure 3). The focus of our research is to provide an integral graphical traffic management software tool to facilitate the process of incident management, route recommendation, and traffic information database management. There is a quickly growing literature on traffic management and IVHS. See, for example, [1], [2], [3]. Section 2.0 briefly describes the ETTM system.

Traffic management systems entail high level of integration of computer and communication technologies as well as the interaction of a human operator. To study the impact of such a system on the actual traffic network, an experimental software tool is under development that provides the necessary functionality for traffic analysis and management. At present, the integral parts of the tool consist of a convenient graphical user interface to support operator activities, a Geographical Information System that facilitates the display of road maps, and a simulator that models the behavior of actual traffic conditions. In the future, we will also include a route guidance module that generates the necessary control strategies to relieve incident of heavy traffic. To assist in the above effort, a separate software study tool is being constructed. It is built to achieve two purposes. First, it is built to help analyze the effect of different route guidance control actions on a hypothetical basis before the result of the study is incorporated into the original software tool. Second, the study tool implements a software structure that provides a guideline for the final implementation.

The current software has been implemented with a GUI/GIS system as well as a simulator for traffic analysis. It is designed in such a way that it does not require the user to understand either software architecture or traffic flow fundamentals. The platform for the graphical user interface (GUI) and the Geographical Information (GIS) is HOOPS, which is a X/Motif like graphic package. All software has been developed on a SUN SPARC station2. Section 3.0 describes the overall architecture of the tool with emphasis on the current versions of the various input modules as well as the GUI/GIS system.

A special feature of the route guidance module is that it will support multiple user-selected traffic control criteria in making routing decisions. This will facilitate the control decision making process which is often a complex process with diverse optimization requirements. Section 4.0 describes the route guidance module.

The simulator is built to support a variety of traffic models, for instance, the Greenshield model. It utilizes a macroscopic simulation technique with a street segment being the basic unit of simulation. Section 5.0 details the simulator and its embedded traffic modeling capabilities.

Section 6.0 concludes the paper by giving an example of the usefulness of the tool.

2.0 ETTM and its application in IVHS

There are several methods for measuring traffic flow and mean speed in highways. Among the conventional methods are inductive loop detection systems and video detection systems. Each of these methods have

1. Research supported by NJDOT and Region II, UTRC.

2. Dept. of ECE, Rutgers Univ. 3. Dept. of ECE, CUNY. 4. Dept. of CE, Rutgers Univ.

5. To be engineered by Faradyne Systems, Inc.

some drawbacks. Inductive loop detection systems are expensive to maintain and not completely reliable. Video detection systems may need significant communication capability and reliable image processing algorithms to handle the video frames, detect any traffic anomaly, and trigger an alarm if necessary.

The system under study for implementation in the NJ/NY area is based on ETTM technology. ETTM tags are to be installed in a fraction of cars in the area. It is expected that more people will subscribe to this service as time goes by, so that eventually a significant fraction of the vehicular traffic in the area will be equipped with these tags. There will also be a sufficient number of ETTM detectors on the roadside. A number of these will be installed at the tolls gates. The main function of the ETTM detectors will be automatic toll collection for cars which subscribe to the service. The remaining ETTM detectors are used for traffic flow monitoring. They essentially measure the travel time of a vehicle equipped with an ETTM tag between two successive ETTM detectors. From these measurements one can deduce the traffic flow and mean speed for all vehicles on the road, whether they are equipped with ETTM tags or not. This is a nontrivial detection and estimation problem. The ETTM system is also expected to detect nonrecurrent incidents such as car accidents. The detection and estimation algorithms will have as their input the raw and/or processed travel time measurements made by the ETTM tags, historical data on normal travel times on that segment of the road and on that particular time of the day, and also some information about the fraction of the cars with ETTM tags traveling on that road.

Like any other detection problem, one would like to maximize the probability of detecting the incidents and yet not have too many false alarms. It is also important to be able to detect incidents in a short period of time from their occurrences, so that actions are taken to correct the situation in a timely manner. It is clear that the system performance will improve with the number of vehicles subscribing to ETTM tags and with the number of ETTM detectors. To have more cars equipped with ETTM tags requires a better market penetration of this technology in the vehicular traffic in the region. To have more ETTM detectors, so that they may be installed at smaller distances from each other, will increase both the initial cost of the system and the future maintenance cost.

Solving the detection and estimation problem with known number and location of the ETTM detectors and known fraction of traffic equipped with ETTM tags is a difficult task. One needs to assume a reasonable stochastic model for the problem, somehow infuse the historical data available into the problem formulation, and then use good engineering judgement and approximations to solve the problem. A less difficult task is the updating of the historical data as the traffic patterns in the region change and more traffic data becomes available. Some solutions for these problems have been proposed in the TRANSCOM's System for Managing Incidents and Traffic (TRANSIT), but more work has to be done to find more comprehensive solutions.

3.0 Input and GUI/GIS Modules

Figure 4 shows the block level diagram of the study tool. It consists of the static, dynamic and historical data input modules, the predicted traffic input module, the data preprocessing module, the incident module, the traffic info display module, the route guidance module, and the simulation module. Figure 5 is a detail diagram of the route guidance module. The following is a discussion of all the modules except the route guidance and simulation modules which will be the topics of the next two sections.

3.1 Static Info Database

The static information database is an input module that deals with the storage and retrieval of the static network information. It includes a geographical representation of the network, links capacity information, link speed limits, reader locations, etc. The static information database is not updated during the operation cycle.

3.2 Historical Info Database

The historical information database deals with past link state data such as link volume and speed according to multiple time of the day. The historical data are updated periodically by the data preprocessing module.

3.3 Dynamic Input Module

The Dynamic data input module reads in real time data such as current link volume and velocity. These data originate from the road side readers and are assumed to have been averaged such that they are suitable for further processing.

3.4 Predicted Traffic Module

The predicted traffic module is the feedback input module that takes the analytical results of the route guidance module and the simulation modules and feeds them as input to the the system for further processing. This makes the effect of past control decisions available for future analysis.

3.5 Data Preprocessing Module

The data preprocessing module is responsible for the preprocessing of data from all the above three input modules. Typically, the real time data collected through the dynamic data module will have a certain degree of error that needs to be calibrated. One possible way this is done is to pass the real time data through a filter that exponentially weighs the importance of the data so that more weight is associated with the most current data and less weight is given to older data. Another way to ensure the usefulness of the real time data is to weigh them against the historical data using a similar scheme. The data preprocessing module monitors the discrepancy between the real time data and the historical data and updates the latter when a consistent difference between the two data sets is noted.

3.6 Incident Module

The incident module pinpoints the location of a traffic incident. An operator can manually input the incident through an operator report. Relevant data such as an estimate of the reduced capacity on the incident link is also an input here.

3.7 Traffic Display Module

The system provides a information retrieval capability through the traffic information module. This module provides the operator with information like travel time estimation, static and dynamic data display, incident location display, and route recommendation display, etc. This information is made easily accessible to facilitate the operator's interaction with the system.

At present, the current software has implemented a GUI and GIS as shown in figure 1. The GUI has provided a user-friendly environment which supports traffic condition display and user command execution. The large window in the center, called the situation window, is where the road maps and the related traffic conditions and reports are displayed. It has menus and buttons around it which correspond to different actions and requests that are available to the user.

The map which is seen in the situation window in figure 1 is a street map of Manhattan in New York City. The underlying system is a GIS tool which displays road networks. Maps from external sources, such as the United States Geological Survey (USGS), can be imported. In case such maps are not available, the GIS module will allow a user to create a road network using its GIS editor. The user must provide the coordinates (fed into the situation window using the mouse) and the street type, i.e., road width, number of lanes, etc. (typed in an interactive dialog box). Modification to existing maps are also allowed using the map editor.

4.0 Route Guidance Module

4.1 User Iterative Incident Routing

Traffic incident management deals with the deployment of traffic control strategies that help alleviate heavy traffic condition at the occurrence of traffic incidents. An effective means of incident control strategy is to inform travellers of alternate pathways before entering the congestion area. To do so, the location of the incident is first detected. Traffic conditions on all monitored streets are gathered. At this point, the data that are collected will be processed and a set of alternative paths will be computed. The routing information will then be disseminated to the drivers through variable road signs or in-vehicle communication systems if they are available.

The process of computation of the alternate paths is an interactive process which requires flexibility. The choice of an appropriate routing strategy can be chosen from a set of possible strategies each based on a specific criterion. At each step, a routing criteria will be specified and the corresponding alternate paths will be generated. The merit of this control decision will be evaluated according to some acceptance criteria which determines whether it is acceptable. If it is not acceptable, the user will choose another routing criteria and repeat the above process until an acceptable control strategy is discovered.

Human and system interoperability is essential in the incident management process. It is very often not clear whether a particular control criteria will satisfy all the requirements imposed by the system and the users of the system. For instance, as will be discussed later, a routing algorithm may minimize total system traffic delay but may result in high individual user traffic delay. In addition, if the driver's perception of an optimal route is determined by the distance travelled and fuel consumption, routing them onto routes that exhibits minimal delay time but long travel distance may not seem to be a good choice as perceived by the drivers. The involvement of an operator in the decision process will allow the incorporation of human intuition which in turn will enhance the quality of the service provided.

4.2 Route Guidance Module

The route guidance module allows the user to select the criteria for a control action and execute the relevant routing algorithm. The module is further partitioned into multiple sub-modules as shown in figure 5. They are the criteria setup sub-module, O-D state sub-module, link state sub-module, shortest path routing sub-module, the flow optimization sub-module, and the manual routing sub-module. We shall describe the structure of the route guidance module here and section 4.0 will give further details of the module.

4.2.1 O-D State Sub-module

To assist diversion of traffic, one needs to estimate the traffic that is expected to flow into the incident link so that they can be diverted early on before reaching the incident point. The O-D estimation sub-module obtains an approximation of this congestion traffic to facilitate the routing algorithms described below.

4.2.2 Link State Sub-module

Algorithms that do not require O-D information make use of link state information to carry out the path finding procedure. An example of this are those algorithms that belong to the category of the shortest path algorithm.

4.2.3 Criteria Setup Sub-module

The criteria setup sub-module allows the operator to specify the set of parameters and the objectives for route guidance. For example, one can optimize individual travel delay or total system delay, to specify the number of alternate paths to be generated and to determine whether delay or distance travelled are to be minimized, etc.

It is uncertain whether one should use the same routing criteria under different traffic conditions. For instance, during light to medium traffic, it might seem reasonable to optimize individual user delay. However, when traffic is heavy, it might seem advisable to optimize system delay. In addition, there are occasions in which drivers might base their judgement of a good route on the actual physical distance travelled. Providing an alternate route that optimizes delay but requires long distance travel might subject the system to driver's questioning of the decision's integrity.

This module provides a convenient interface to specify routing criteria.

4.2.4 Shortest Path Routing Sub-module

The tool provides a set of routing algorithms such as shortest path algorithms based on the concept of link weights. The weight of a link could be the estimated travel delay or physical distance of the link. An efficient shortest path algorithm is Dijkstra's k-shortest path algorithm [4].

4.2.5 Flow Optimization Routing Sub-module

Another category of routing algorithms are the flow optimizing algorithms. Examples are the Frank-Wolf algorithm [5] and the gradient projection algorithm [6]. These set of algorithms relies on the concept of multiple path flows and cost of traffic. Traffic can be thought of as flow which can be diverted among the chosen paths between an OD pair generated in the O-D estimation module. Optimal assignment of flow into these paths results in a reduced cost of traffic.

4.2.6 Manual Routing Sub-module

Beside studying traffic conditions with the above well-known automatic routing methods, one can manually assign traffic and evaluate the effect of the assignment. As stated above, traffic routing is viewed as an iterative process. Allowing leeway to manually assign traffic makes the process more flexible.

5.0 Simulation Module

After the discussion on the structure of the software tool and the functionalities of the route guidance module, we will now turn our attention to the simulation module. This module enables the tool to handle the dynamic nature of the traffic network by realizing a realistic traffic model based on macroscopic traffic quantities, such as density and speed, and an iterative updating mechanism to simulate the traffic action. We will describe the traffic model that the simulator implements and how it is used to simulate traffic.

5.1 Traffic Prediction Module

The traffic prediction module provides facility to predict and evaluate the current routing strategy. It applies the current strategy to the network and predicts the outcome of such a strategy given the current state of the network. It provides the basis to accept or reject the control strategy. If the strategy is accepted, it is recommended to the network operator for deployment. If it fails, the operator will generate a new strategy with the route guidance module. The generation of a good control strategy is an iterative process.

5.2 Traffic Model

On the map, streets are laid out as networks of arbitrary connected and oriented street segments. We will be interested in the "macroscopic traffic models" in which a group of cars in a segment are treated as a unit. The three parameters that specify the vehicle traffic are speed, v [m/second]; density defined as the number of vehicles occupying a unit length of road, k [vehicles/m], and volume defined as the number of vehicles per second passing a certain point on the road, q [vehicles/second]. Each of these parameters is defined for a segment.

Assuming that all the vehicles in the segment have equal speed, the vehicle volume is given by:

$$q = kv \qquad \text{(EQ1)}$$

Intuitively, it is clear that as the vehicle density goes up (the street becomes congested) the corresponding speed must go down. It is shown in [7] that models based on an extension of classical fluid models represent well, in some cases, the relation between the speed and density. The "dynamic equation" of a fluid can be written as:

$$\frac{dv}{dt} = -\frac{\varphi^2}{k}\frac{\partial k}{\partial x}$$

where x is the distance along the road and φ is a nonnegative constant with the dimension of speed.

Since the assumption that the traffic behaves as a classical fluid is too restrictive [8], we are using a more general set of models given by:

$$\frac{dv}{dt} = -\varphi^2 k^n \frac{\partial k}{\partial x} \qquad \text{(EQ2)}$$

where φ is a nonnegative constant, and n is the traffic model parameter. Solving (EQ 2) for v we have:

$$v = \begin{cases} \dfrac{\varphi k_c^{(n+1)/2}}{(n+1)}\left(1 - \left(\dfrac{k}{k_c}\right)^{(n+1)/2}\right), n \neq -1 \\ \varphi \log\left(\dfrac{k_c}{k}\right), n = -1 \end{cases} \qquad \text{(E3Q)}$$

where $(\varphi k_c^{(n+1)/2})/(n+1)$ (free running speed) is the speed at k = 0 and k_c is the density at which $v = 0$ ..

For different values of n we have different macroscopic traffic models [1,2]:

a. Greenshield's or linear model $(n = 1)$

b. Greenberg's model $(n = -1)$

c. Drew's models $(n \in [-1/2, 1/2])$

as well as by using any combination of above for different density ranges - multiregime models. Usually, for single regime models $v_0 = min$[free running speed, street speed limit]. The simulator can accommodate these models, appropriate for different traffic conditions.

5.3 Traffic Simulation

The simulator operates by iteratively updating all the densities in the following manner. The relations between traffic parameters given in the previous section are valid for one street segment. In a larger street pattern, the dependence between the parameters in neighboring segments needs to be considered. As an example in figure 2, we show a typical street. Assuming that all the streets are one-way streets (or equivalently, that the vehicle can turn from A into O_1, O_2, O_3 and O_4 in this example, or up to O_M in general) it is clear that the velocity in A will depend on the traffic density in these streets (as well as on the density in A) and also on the vehicle turning probability. On the other hand, the density in street A depends on the input flow into the street and can be written as:

$$k_A^{(j)} = min\left[k_{A,c}, k_A^{(j-1)} - \Delta k_A^{(j-1)}\right.$$

$$\left. + f(\{\Delta k_{I_i}^{(j-1)}, i = 1, 2, \dots, N\})\right] \quad \text{(EQ4)}$$

where N is the number of streets entering street A (in Figure 5, $N = 4$), j is the simulation step, Δk is the street density change caused by entering an exiting traffic and $k_{A,c}$ is the

maximum density in street A. Note that, in general, N can include also U-turn from street A.

The street density change is related to the flow by:

$$\Delta k_{I_i}^{(j)} = q_{I_i}^{(j)} \frac{\Delta t}{l_{I_i}} = \frac{k_{I_i}^{(j)} v_{I_i}^{(j)}}{l_{I_i}} \Delta t$$

where l_{I_i} is the street length and Δt is the time interval of one simulation step.

The functional dependence f is given as a linear function of the Δk_{I_i}

$$f(\{\Delta k_{I_i}\}) = \sum_{i=1}^{N} P_{I_i} \Delta k_{I_i} \quad \text{(EQ5)}$$

:where P_{I_i} is the probability of a vehicle making a turn into street A from street I_i. Similarly, vehicle speed is given by:where M is the num-

$$v_A^{(j)} = min\left[v(k_{A,c}, v_{A,0}, k_A^{(j)}),\right.$$

$$\left. g(\{v(k_{O_i,c}, v_{O_i,0}, k_{O_i}^{(j)})\}, i = 1, 2, \dots, M)\right] \quad \text{(EQ6)}$$

ber of streets exiting street A (including the U-turns), and $v(k_{A,c}, v_{A,0}, k_A)$ is defined by (EQ 3)

with $k_c = k_{A,c}$ and $v_0 = v_{A,0}$ and g() is taken to

have the same functional form as f() but is a function of M variables, rather than N (it is a function of M exiting streets).

After each density update by (EQ 4), the speeds are updated using (EQ 6).

Extension of the simulation model for a general case of a two-way street is done by considering each direction as a separate one-way segment for each two-way street, with U-turn probability at each intersection (for the U-turn at each two-way intersection particular I_i is equal to one of the O_j's).

6.0 Conclusion

To demonstrate how the tool might be utilized to manage traffic, we will look at an example. In 4.0 we discussed the structure of the final system. We will therefore use it in the following description.

We will have information such as the network map, the link capacities, and the historical link state data, etc. These data will be stored in the static and historical databases. Real-time data is then read from the real-time input module. The operator will use the preprocessing module to weigh the real-time data according to a chosen scheme. When an incident is reported, it is input to the software through the incident module. A dialog box is used to enter either a numerical value representing the intensity of the congestion at the location of the incident or simply to indicate the degree of congestion by light, moderate or heavy traffic using a color coding scheme.

After all the necessary inputs were made, we then turn to the route guidance module. With regard to the traffic situation, the operator will select the criteria for route recommendation. Feasible criteria are minimizing individual driver's delay, minimizing system's delay, minimizing individual travel distance, etc. Experience and intuition are required here in the specification process. The operator might choose to select his own alternate paths to divert traffic or use the shortest path sub-module routines to automatically generate them. After that, the operator might want to manually assign traffic into the alternate paths or use the flow optimization sub-module routines to perform optimal traffic assignment.

The operator is provided with two mechanisms to evaluate the merit of the routing strategy chosen. First, he will be shown the effect of the static traffic assignment process by using the display module. Numerical or color coded displays are available to exhibit the state of traffic after the assignment. Secondly, the operator could input the route recommendation data (the alternate paths and the amount of traffic diversion) into the simulation module, execute the simulation for a particular number of time steps and obtain a dynamic result. Based on the judgement of the operator, he will decide whether the selected strategy is acceptable for deployment or not.

To summarize, a simulation based graphical software tool which provides a convenient GUI/GIS interface and efficient control strategy generation module for traffic management is presented. The tool allows flexible selection of control strategy criteria to be utilized in the control strategy generation process. It allows both static and dynamic control strategy evaluation. The generation of a good strategy requires an iterative process where a new strategy is generated at each step until one that is acceptable according to the specified acceptance criteria is found. The simulation adapts a macroscopic traffic model which relates traffic parameters among different segments of the road. It relies on a time step updating mechanism to model traffic flow.

Bibliography

[1] L. R. Rilett, M. E. Van Aerde. *Modeling Distributed Real-time* Route Guidance Strategies in a Traffic Network that Exhihits the *Braess Paradox.* Proceeding of the VNIS'91 Conference, Dearborn Michigan, 1991, pp 577-587.

[2] K. Chen, S. E. Underwood. *Research on Anticipatory Route*

Guidance. Proceeding of the VNIS'91 Conference, Dearborn Michigan, 1991, pp 427-439.

[3] D. E. Kaufman, R. L. Smith, K. E. Wunderlich. *An Iterative Routing /Assignment Method for Anticipatory Real-time Route Guidance.* Proceeding of the VNIS'91 Conference, Dearborn Michigan, 1991, pp 693-700.

[4] Topkis D. M. *A K-Shortest Path Algorithm for Adaptive Routing in Communications Networks.* IEEE Transactions on Communications, Vol. 36, No. 7, July 1988. pp. 855 - 859.

[5] Zangwill, W. 1969. *Nonlinear Programming: A Unified Approach.* Englewood Cliffs, N.J.: Prentice Hall.

[6] Luenberger, D. G. 1984. *Linear and Nonlinear Programming.* Reading MA: Addison -Wesley.

[7] A. D. May. *Traffic Flow Fundamentals.* Prentic Hall, 1990.

[8] H. Inose and T. Hamada. *Road Traffic Control.* University of Tokyo Press, 1975.

FIGURE 1. Display Interface Module

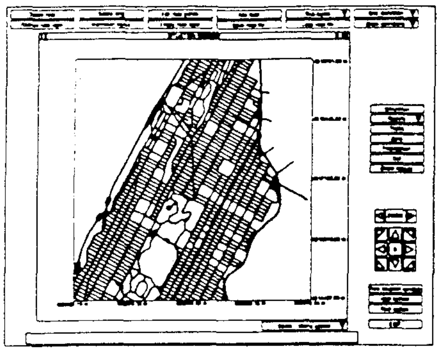

FIGURE 2. Typical street segments

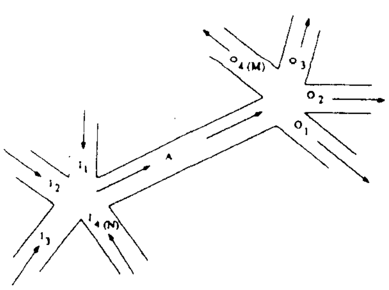

FIGURE 3. Project site in the NJ/NY area

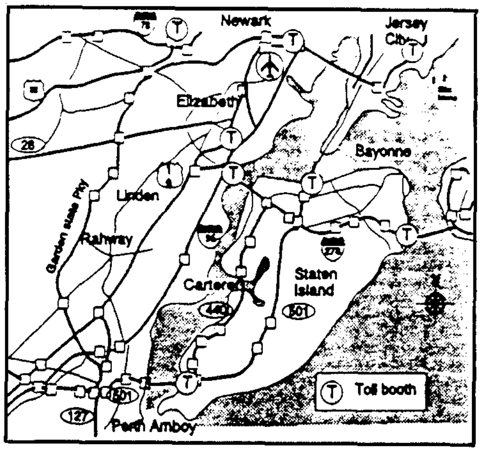

FIGURE 4. Incident Management Tool

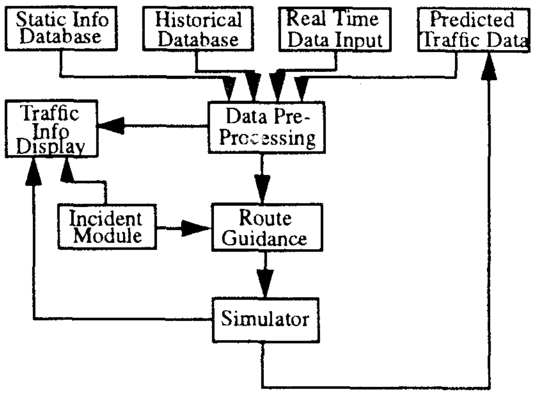

FIGURE 5. Route Guidance Module Detail