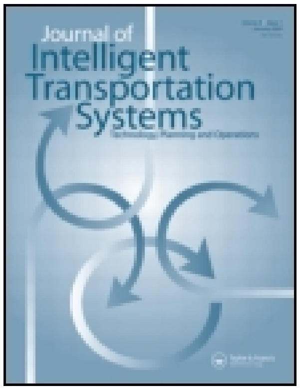

I V H S Journal
Publication details, including instructions for authors and subscription information:
http://www.tandfonline.com/loi/gits18

NEURAL NETWORK MODELS FOR TRAFFIC CONTROL AND CONGESTION PREDICTION

John F. Gilmore [a] & Naohiko Abe [b]
[a] Georgia Tech Research Institute Georgia Institute of Technology , Atlanta, Georgia, 30332
[b] Mitsubishi Heavy Industries, LTD Nagoya Guidance & Propulsion Systems , Aichi-Ken, Japan
Published online: 24 Oct 2007.

To cite this article: John F. Gilmore & Naohiko Abe (1995) NEURAL NETWORK MODELS FOR TRAFFIC CONTROL AND CONGESTION PREDICTION, I V H S Journal, 2:3, 231-252, DOI: 10.1080/10248079508903828

To link to this article: http://dx.doi.org/10.1080/10248079508903828

IVHS Journal, 1995, Vol. 2(3), pp. 231–252
Reprints available directly from the publisher
Photocopying permitted by license only

NEURAL NETWORK MODELS FOR TRAFFIC CONTROL AND CONGESTION PREDICTION

John F. Gilmore[1] and **Naohiko Abe**[2]
*1: Georgia Tech Research Institute
Georgia Institute of Technology
Atlanta, Georgia 30332*
*2: Mitsubishi Heavy Industries, LTD
Nagoya Guidance & Propulsion Systems
Aichi-Ken, Japan*

Advance Traffic Management Systems (ATMS) must be able to respond to existing and predicted traffic conditions if they are to address the demands of the 1990's. Artificial intelligence and neural network are promising technologies that provide intelligent, adaptive performance in a variety of application domains. This paper describes two separate neural network systems that have been developed for integration into a ATMS blackboard architecture. The first system is an adaptive traffic signal light controller based upon the Hopfield neural network model, while the second system is a backpropagation model trained to predict urban traffic congestion. Each of these models are presented in detail with results attained utilizing a discrete traffic simulation shown to illustrate their performance.

Key words: neural networks, congestion prediction, traffic control

INTRODUCTION

The goal of an ATMS is to optimally manage existing transportation resources through the use of adaptive control systems in order to maximize the efficiency and usefulness of all transportation modes. The intelligent, adaptive control aspects of this problem are attuned to the features of neural network systems. Neural networks [Anderson et al., 1988] are computational structures that model simple biological processes usually associated with the human brain. Adaptable and trainable, they are massively parallel systems capable of learning from positive and negative reinforcement.

The basic element in a neural network is the neuron (Figure 1). Neurons receive input pulses (I_i) from interconnections with other neurons in the network. These interconnections are weighted (W_i) based upon their contribution to the neuron. Weighted interconnections are summed internally to the neuron and compared to a threshold value. If the threshold value is exceeded, a binary output pulse (O_i) is transmitted, otherwise the neuron exhibits no output value. A variety of specialized neural network models based upon this simple neuron structure have been developed.

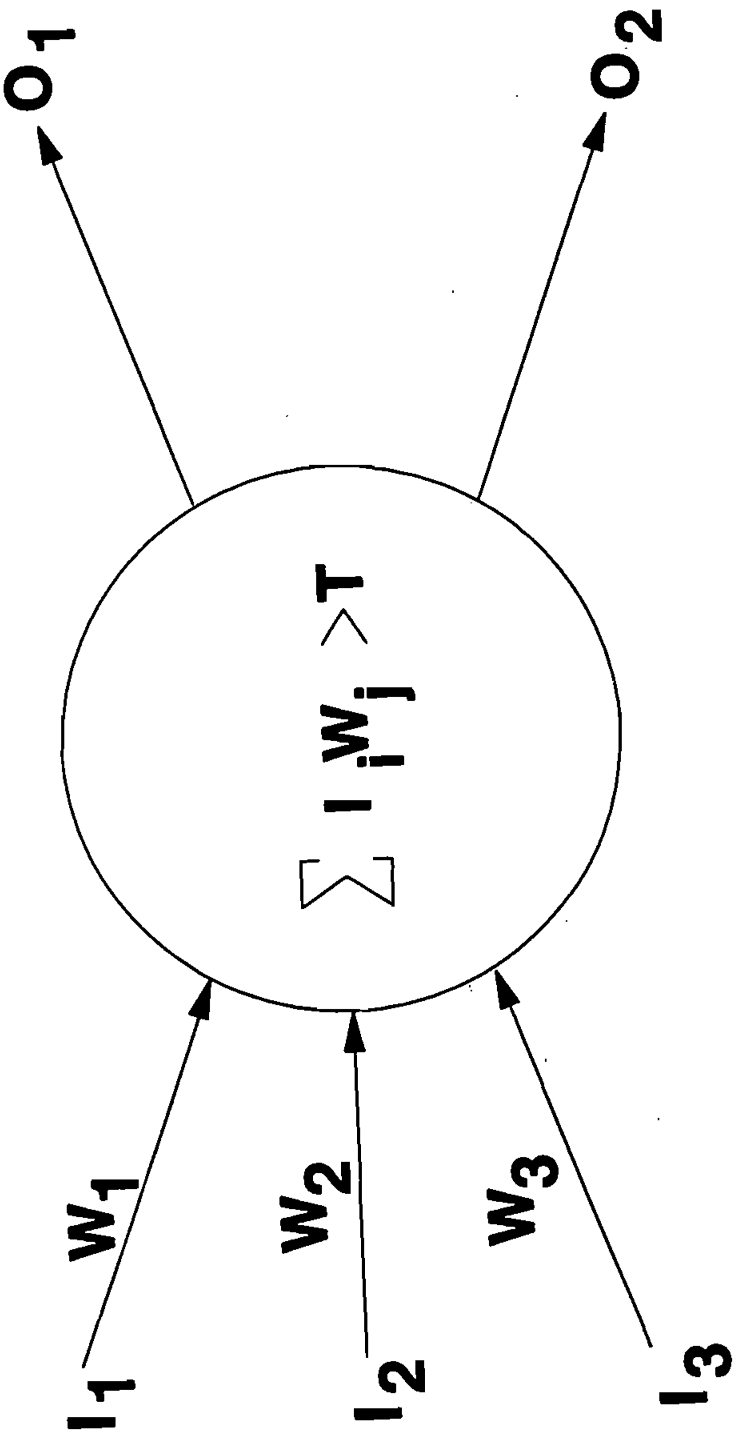

FIGURE 1 Basic Neuron Structure

This paper describes the application of neural network to two ATMS functions. First, an intelligent neural network-base signal light control system capable of adaptively optimizing traffic flow in urban areas is presented. Second, a neural network model capable of learning how to accurately predict traffic congestion is discussed.

Current signal control systems, such as the Los Angeles Automated Traffic Surveillance and Control (ATSAC), use responsive control [Rowe, 1991]. ATSAC selects its timing plans based upon a comparison of actual surveillance data and available model data. This approach is an improvement over the time-of-day timing plan in instances where traffic varies each day. The proposed neural network approach extends the ATSAC philosophy further by applying a neural network optimization model to actual traffic flow data to determine the signal light settings that will produce the most efficient traffic flow in a special event area. Utilizing information on street segment capacities, traffic flow rates, and potential flow capacities, the network model examines the effects of signal light settings in relation to the traffic flow away from a designated area. The system "settles" on control settings that maximize the flow and adaptively changes the signal settings based upon changes in street segment traffic density.

Current developments in advanced traffic control techniques are giving rise to an increasing requirement for reliable near-future forecasts of traffic flow. These predictions are required in order to attain the background information for solving traffic congestion before it develops using methods such as "gating" or "dynamic route guidance." Existing systems such as SCOOT [Robertson et al., 1991] only react to present traffic patterns and, by themselves, do not prevent congestion from occurring. Conventional traffic modeling and simulation procedures can be applied to this problem but have a number of shortcomings, particularly in real-time applications.

Many researchers have demonstrated that neural network methods based on a back-propagation algorithm are able to deal with complicated nonlinear forecasting tasks in stock prices, electricity demand, and water supply [Canu et al., 1990]. By specifying input values representing important traffic congestion attributes, two neural network architecture capable of capturing the underlying characteristics of the transportation domain have been developed. The first model addresses the dynamic control of traffic signal lights in urban environments. The second model is able to learn based upon data from previous congestion occurrences and has produced encouraging results in forecasting congestion on surface streets.

TRAFFIC SIGNAL CONTROLLER

At first glance, a traffic flow system appears to be an interwoven and connected array of road sections whose traffic flow is determined by a series of traffic lights. The control of these traffic lights is vital in order to allow traffic to flow throughout the system with minimal delay. A neural network approach to the control of signals at traffic intersections is proposed. Because the optimal traffic signal configuration is not available a priori, a supervised learning architecture such as a back-propagation network is not suitable. The neural net architecture should do unsupervised learning in an optimization network. The Hopfield neural network model was chosen because of its [a] ability to optimize complex network flows, and [2] direct representational structural mapping of neurons to traffic intersections.

The main parameter in Hopfield is the energy function which is distributively defined by the connection architecture among the neurons and the weights assigned to each connection. The Hopfield model and the traffic-derived energy function it utilizes for intelligent signal light control are described in the following subsections.

The Hopfield Model

The Hopfield model is an additive neural network model. This means that the individual weighted inputs to a neuron are added together to determine the total activation of neuron. This activation is then passed through an output function to determine an output value.

The Hopfield model uses a fully interconnected network of neurons to descend onto an energy function. Since a discrete time simulation is being used, a discrete-time model was adopted. The dynamics of the discrete-time Hopfield Net are given by:

$$U_i = \sum_{j=1}^{N} T_{ij} V_j + I_i$$

$$V_1 = g(U_i)$$

where

T(i,j) are the interconnection weights,
I(i) are the input biases,
U(i) are the internal states,
V(i) are the neuron outputs, and
g(x) is a nonlinear activation function which can be taken as

$$g(x) = \frac{1}{2}\left(1 + \tanh\left[\frac{x}{x_0}\right]\right)$$

which approaches a hard limiter as x_0 tends to zero. An asynchronous update rule was used which means that neurons are randomly chosen to be updated. This asynchronous updating scheme tends to greatly reduce oscillatory or wandering behavior typical of synchronous updating schemes.

Hopfield and Tank [Hopfield et al., 1984 and 1985] show that the dynamics of this model favors state transitions that minimize the energy function

$$E = -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} T_{ij} V_i V_j - \sum_{i=1}^{N} I_i V_i$$

where

> ***Ti,j*** *are the interconnection weights,*
> V_i *are the neuron outputs,*
> I_i *are the input biases, and*
> *N is the number of signals,*

so that the network gradually settles into a minima of this function. The difficulty to any application using a Hopfield Net is to determine a suitable energy function that the network will descend on.

Neural Signal Controller

The goal of traffic management is to maximize the flow of traffic while minimizing incidents and delays in a region. Controlling the timing of signals to increase traffic flow is one method of supporting this goal. The management of traffic flow by a system utilizing the control of signals appears to directly map into a Hopfield network. Since traffic lights have two states, each traffic signal can be modeled as an individual neuron (V_i) in a Hopfield neural network. If a light is on, the traffic will flow in the E-W direction at that intersection, otherwise traffic flows in the N-S direction. I_i can be viewed as the input potential into a node of the network with $T_{i,j}$ viewed as the connections between traffic lights.

The first step in any application of the Hopfield network is the construction of an energy function. The system's primary objective is to disperse traffic away from a special event in the shortest amount of time. In an abstract sense, one method of dispersing traffic is to route vehicles from road segments containing a large number of vehicles and a high capacity percentage (vehicle/capacity) onto road segments with a smaller number of vehicles and a lower capacity percentage. This can be achieved by [a] changing signal lights based on the potential of a given traffic light to increase flow, and [b] synchronizing signals with adjacent traffic lights to maximize overall system throughput.

Initial attempts at creating an energy function that addresses these criteria concentrated on examining the number of vehicles on roadways to determine optimal signal light configurations. Experimentation soon indicated that the percentage of the capacity of an incoming roadway was a better indicator for determining the state of any given traffic signal. Research into the integration of road capacity percentages into the energy equation was then undertaken.

First, a road segment nearing its capacity will trigger a signal to turn on. Second, optimal flow tends to favor near capacity road sections directing their vehicles onto road sections with a lower percentage of capacity. Because of this, a look at the difference of percentages was adopted. The potential of a given signal for traffic flow could be determined by adding the differences between the traffic capacity percentages into the signal and the percentages of capacity of the flow away from the intersection. Thus, the network tends to turn traffic lights on when they have a higher potential for large traffic flow.

The next step in developing the energy equation was to address signal light synchronization. If a signal is on, adjacent signals should tend to turn on to further increase traffic flow. A term was added to the energy function to reflect this tendency of an adjacent traffic light to turn green based on the potential of adjacent traffic. With this addition, the final energy function specifying the optimal control of the traffic lights in the simulation was defined as:

$$E = \sum_{i=1}^{N} V_i P_i \Bigg(\sum_{l=1}^{2} \left(\left(a_l + \sum_{j=1}^{4} r_{l,j} D(a_l, b_j) \right) - \left(a_{l+2} + \sum_{j=1}^{4} r_{l+2,j} D(a_{l+2}, b_j) \right) \right) +$$

$$\sum_{k=1}^{2} \left(V_{i,k} \sum_{j=1}^{3} S_{k,j} D(b_k, c_{k,j}) - V_{i,k+2} \sum_{j=1}^{3} S_{k+2,j} D(b_{k+2}, c_{k+2,j}) \right) \Bigg)$$

where

a *is the capacity percentage of the link entering an intersection,*
b *is the available capacity percentage of a link,*
N is the number of traffic lights,
M & L *are the # of outgoing roadways,*
P(i) *is the priority of a traffic light i,*
t(i) *is the time that traffic light i has been on,*
C(b) *is the capacity percent of road section b,*
D(a)(b(j)) *is the difference cap(a)-cap(b(j)),*
V_i *is the output of neuron i representing traffic light i,*
r_j *is the % of vehicles turning from a onto b(j), &*
s_k *is the % of vehicles turning from b(j) onto c(k).*

The energy function tends to favor signal lights with large capacities flowing through their intersections. The P_i term provides a prioritization weight of individual traffic lights in the simulation. This furnishes the traffic engineering a mechanism to give added weight to heavily congested streets (e.g., the main street leaving a sports arena at the end of an athletic event). A $(1+t_i)$ denominator term was initially used to negatively weight traffic lights that have been a fixed state for an extended time duration, but experimentation indicated that this term was out weighed by the traffic signal energy summations. A symbolic representation of energy function components is illustrated in Figure 2.

The energy equation derived is applicable to standard traffic intersections consisting of two crossing roads. Current research is exploring the derivation of energy equations unique to specific types of intersections (e.g., three road intersections, turn lanes, etc). The Hopfield model supports the use of different energy functions without further modification.

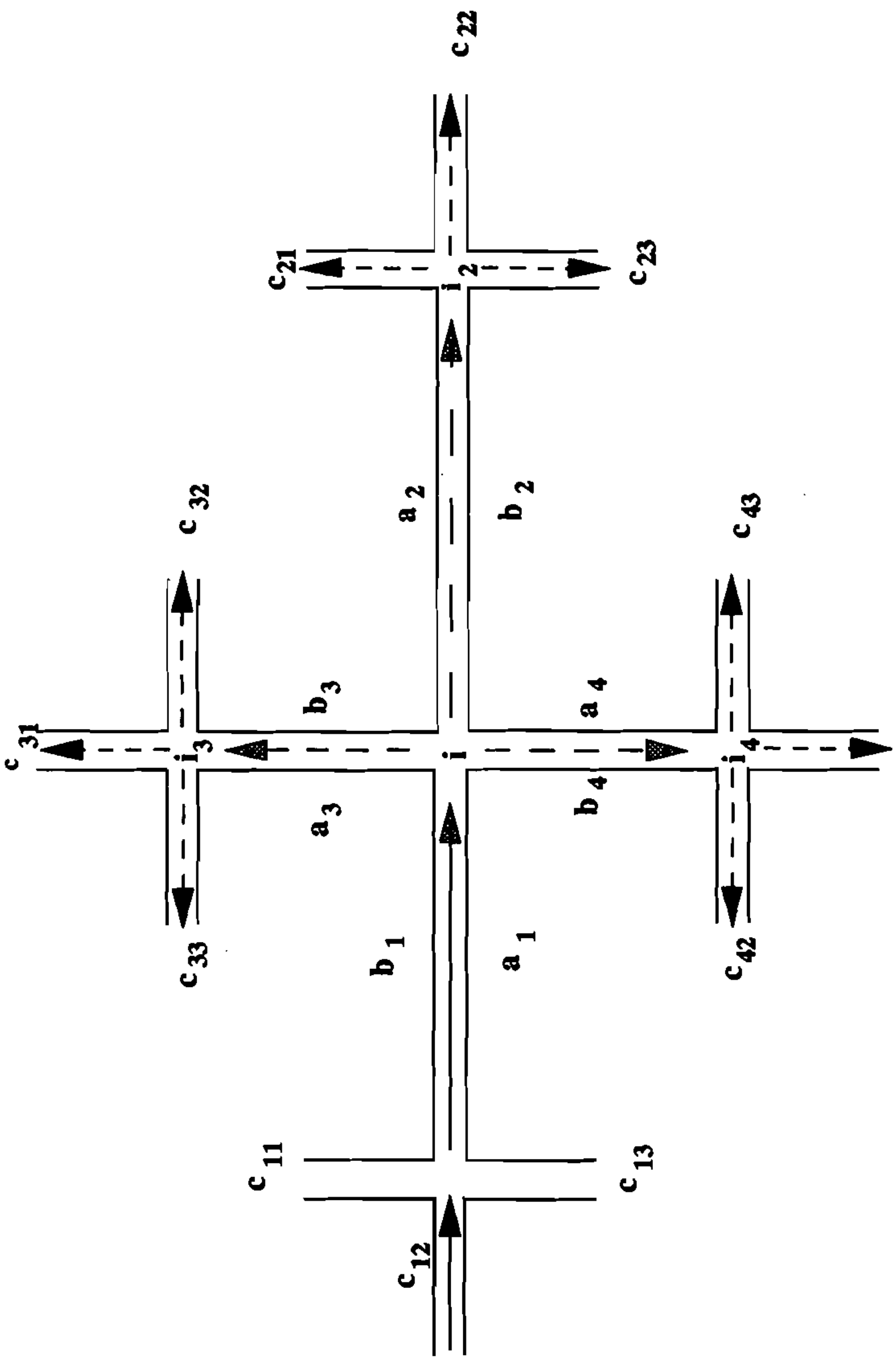

FIGURE 2 Traffic Signal Control Energy Function Derivation

NEURAL NETWORK CONGESTION FORECASTING

An ATMS must not only control current traffic, but also predict where congestion will occur. Predicting congestion so that preventive actions may be taken in advance will greatly alleviate traffic gridlocks. This section describes the results achieved utilizing a back-propagation neural network algorithm to predict the traffic flow on surface street in metropolitan areas. The neural network is trained in two phases. First, an initial learning phase determines the most appropriate connecting weights for data on a typical business day. Second, adaptive learning is employed to learn the special case traffic classes and adapt the weights to the present situation. In the adaptive learning phase, the error function is computed by placing a restriction on the weight changes so that the knowledge learned through the initial learning phase is retained. The prototype system is tested through computer simulations, with results indicating that the application of the neural networks to traffic congestion forecasting is promising.

A multi-layered network consisting of three completely connected layers (i.e. the input layer, the hidden layer, and the outer layer) was developed to address the traffic forecasting problem. The learning was achieved using a back-propagation algorithm with a sigmoid transfer function. This function was very suitable to the traffic problem as traffic flows always possess a saturation characteristic.

The number of input neurons used in the initial prototype network was 48. These were composed of the target flow (T in Figure 3) and three inflows into the simulated area (I_1, I_2, and I_3) which are closely relevant to the target flow. Each variable was normalized to [0,1] by using the capacity percentage of the road for the target flow and the maximum values of observed data for inflows, respectively. It should be noted that *all* data was sampled every 5 minutes, though time sampling in the systems is variable. The outputs are target flows in the next 30 minute period, thus 6 neurons are required. Although there is much difficulty in determining the number of neurons for the hidden layer, 12 neurons were chosen without any effort for optimization in this study. The number of hidden layers appeared to have no measurable effect on prediction performance.

The forecasting algorithm contains two phases. First, the learning phase is used to compute the optimal weights of the neural network for a typical pattern. The second is an adaptive forecasting phase to adapt the weights to the present traffic flows and forecast future congestion.

Training

The training data consisted of traffic flow histories from a typical business day, and the network connecting weights were computed by following the standard back-propagation learning rule described below:

$$W_{ji}(n+1) = W_{ji}(n) + \Delta W_{ji}(n)$$

$$\Delta W_{ji}(n) = -\eta \frac{\partial E}{\partial W_{ji}} + \alpha \Delta W_{ji}(n-1)$$

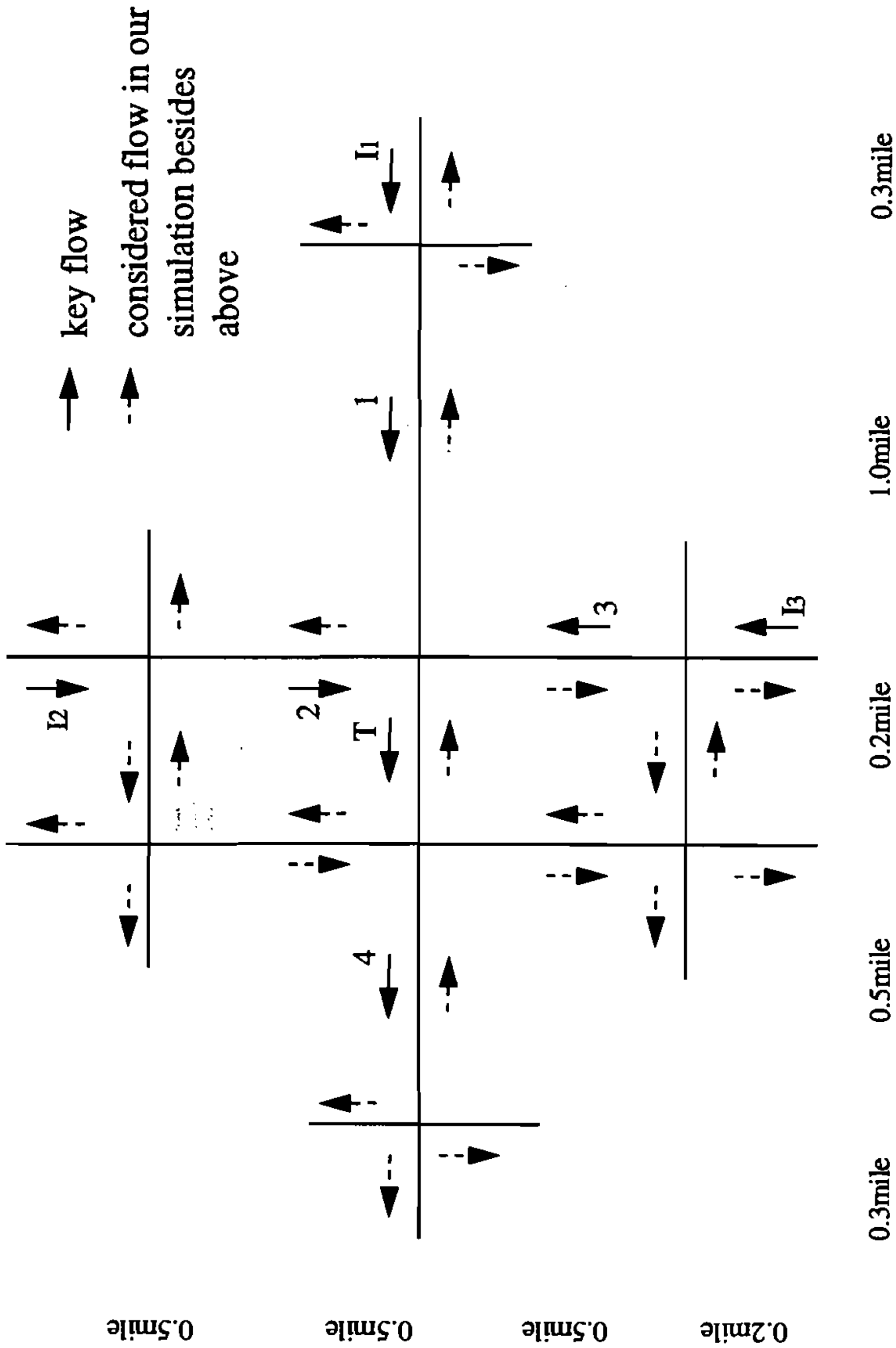

FIGURE 3 Traffic Congestion Evaluation Network

where

$$E = \frac{1}{2}\sum_t (Y_t - O_t)^2$$

Y_t = t^{th} desired output

O_t = t^{th} actual output

Once the initial learning is completed, the network is ready for the adaptive and forecasting process. The connecting weights are corrected according to the forecasting error of the last few data through back-propagation. In order to retain the knowledge learned in the initial learning phase, an error function with a term to suppress changes of weights was implemented:

$$E = E_o + \lambda E_1$$

where

$$E_o = \frac{1}{2}\sum_t (Y_t - O_t)^2$$

$$E_1 = \frac{1}{2}\sum_t \left(\frac{W_{ji} - W_{ji}^0}{\Delta W_{max}} \right)^2$$

$W_{ji}^0 = initial\ weight$

$\Delta W_{max} = maximum\ value\ of\ |W_{ji} - W_{ji}^0|$

In addition, a shifting learning method was used to take the latest available data into account. The network was adapted to data for the past 2 hours, and would then forecast traffic flow for the next hour. In the learning period, all of outputs corresponding to input data (up to the last 30 minutes) are available for training data. A portion of the outputs corresponding to input data for the last 30 minutes, however, are not available. This approach adapted all the weights to the training data (except for the last 30 minutes) and only the weights between the available outputs and the hidden layer to the training data from the last 30 minutes. After the adaptation process, a traffic flow forecast for the next 30 minutes is generated.

Forecasting

The data described in section 4 was used for the initial congestion prediction learning. In order to generate test data, random fluctuations (+20%) were added to the inflow traffic patterns shown in Figure 4, and random fluctuations were also

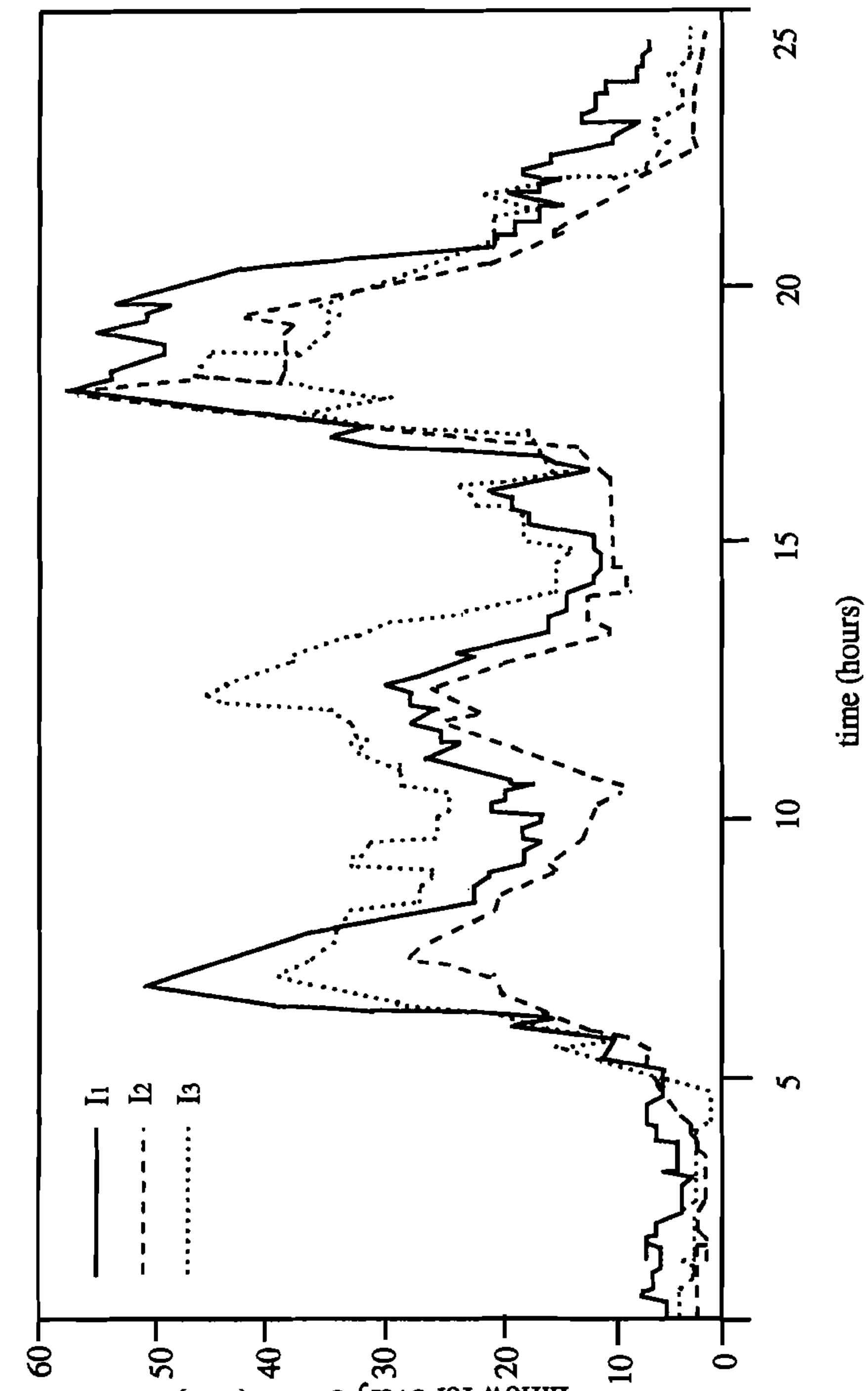

FIGURE 4 Inflow With Random Noise Fluctuations

added to splitting rates of all flows. Figure 5 shows the traffic flow patterns of inputs I_1, I_2, and I_3 chosen to generate the data on a typical business day used for training the neural network. Simulations were run over a 24-hour period of time. Although the simulation model was basic, the results are consistent with practical phenomena. Three forecasting criteria, PAAE, standard deviation of absolute error, and PITP, were used to evaluate the system performance [Srirengan et al., 1991]. PAAE (Percentage Average Absolute Error) and standard deviation of absolute error are calculated as

$$PAAE = (1/N)\sum_t |Y_t - 0_t| / (target\,range) * 100$$

$$STDEV = \sqrt{(1/N)\sum_t \left(|Y_t - O_t| - (1/N)\sum_t |Y_t - O_t|\right)^2}$$

where note that target range is equal to 1 and therefore PAAE is equivalent to

$$(1/N)\sum_t |y_t - o_t|$$

PITP (Percentage of Incorrect Turning Points) is the percentage of times the prediction of the system is an increase (or decrease) in a period when in the actual result is the opposite.

Three cases (as shown in Figure 6) were tested for comparison: the case without the adaptive learning (Case 1), the case with the adaptive learning using the standard error function (Case 2), and the case with the adaptive learning using the proposed error function (Case 3). In all instances, the correct trend was forecasted in more than 85% of the time for the test set. Although the forecast accuracy between Cases 1 and 2 did not always improve, it was clearly always improved in Case 3. This indicates that the error function was successful in guiding the network's adaptive learning. In addition, it appears that the forecast accuracy of the next 5 minutes is superior and forecast of the next 30 minutes is poorest in all cases. This is because the data for the next 5 minutes has a stronger correlation with the input data, as compared to data for the next 30 minutes data. Having access to real-time data through an optical link would improve the system's forecasting ability across all time periods.

TRAFFIC FLOW SIMULATION

Traffic data was generated using a discrete traffic simulator of downtown Atlanta. This area was selected as it is the site for the 1996 Summer Olympics and as such presents the most challenging traffic management problem the United States will witness in the 1990's.

Modeling traffic flow is generally analogous to current in an electrical circuit, or liquid flowing in pipes. Just as Ohm's law gives us a relationship between voltage, current and resistance, when considering traffic streams during uninterrupted flows,

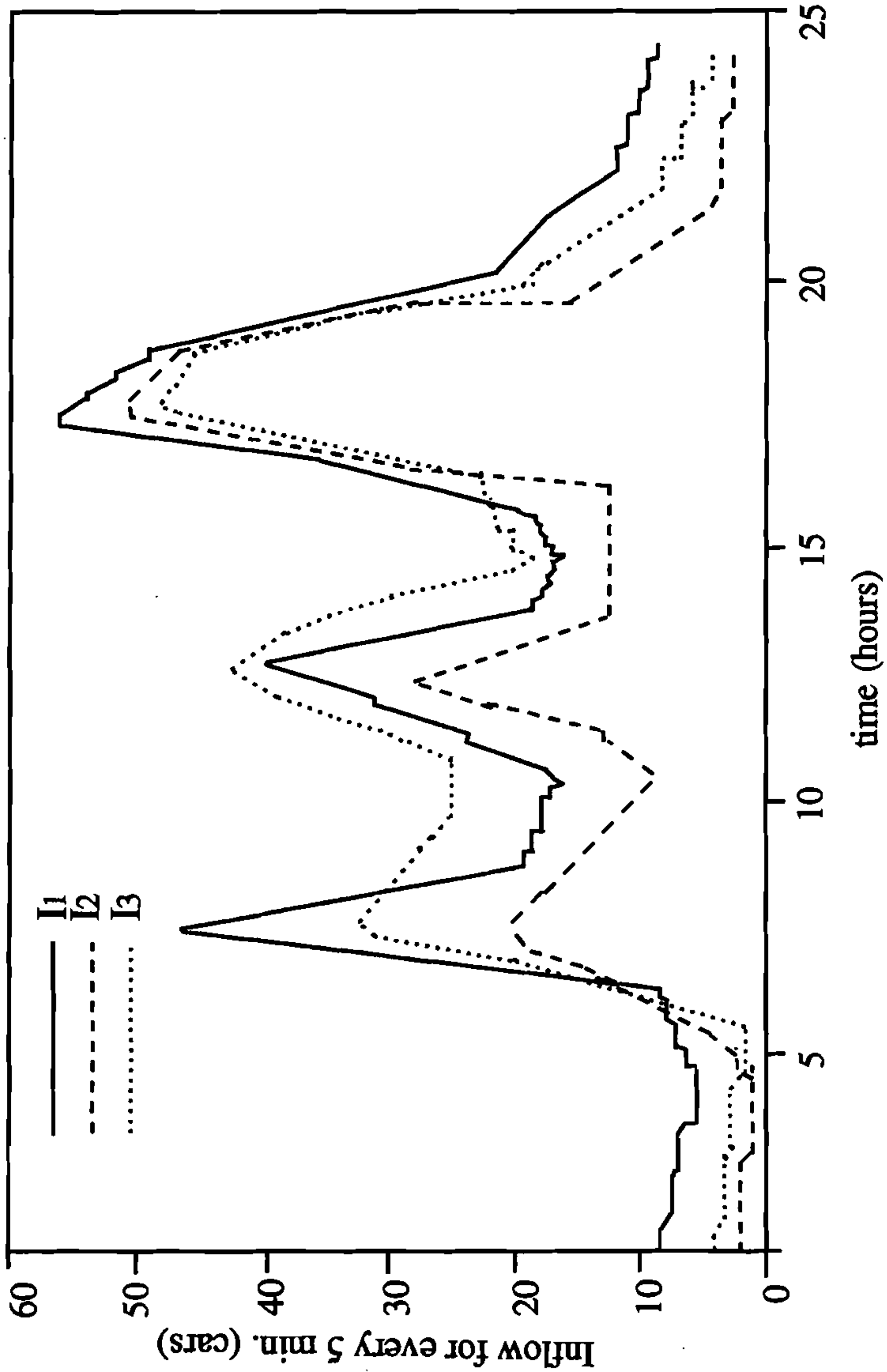

FIGURE 5 Traffic Flow over a 24 Hour Week Day Period

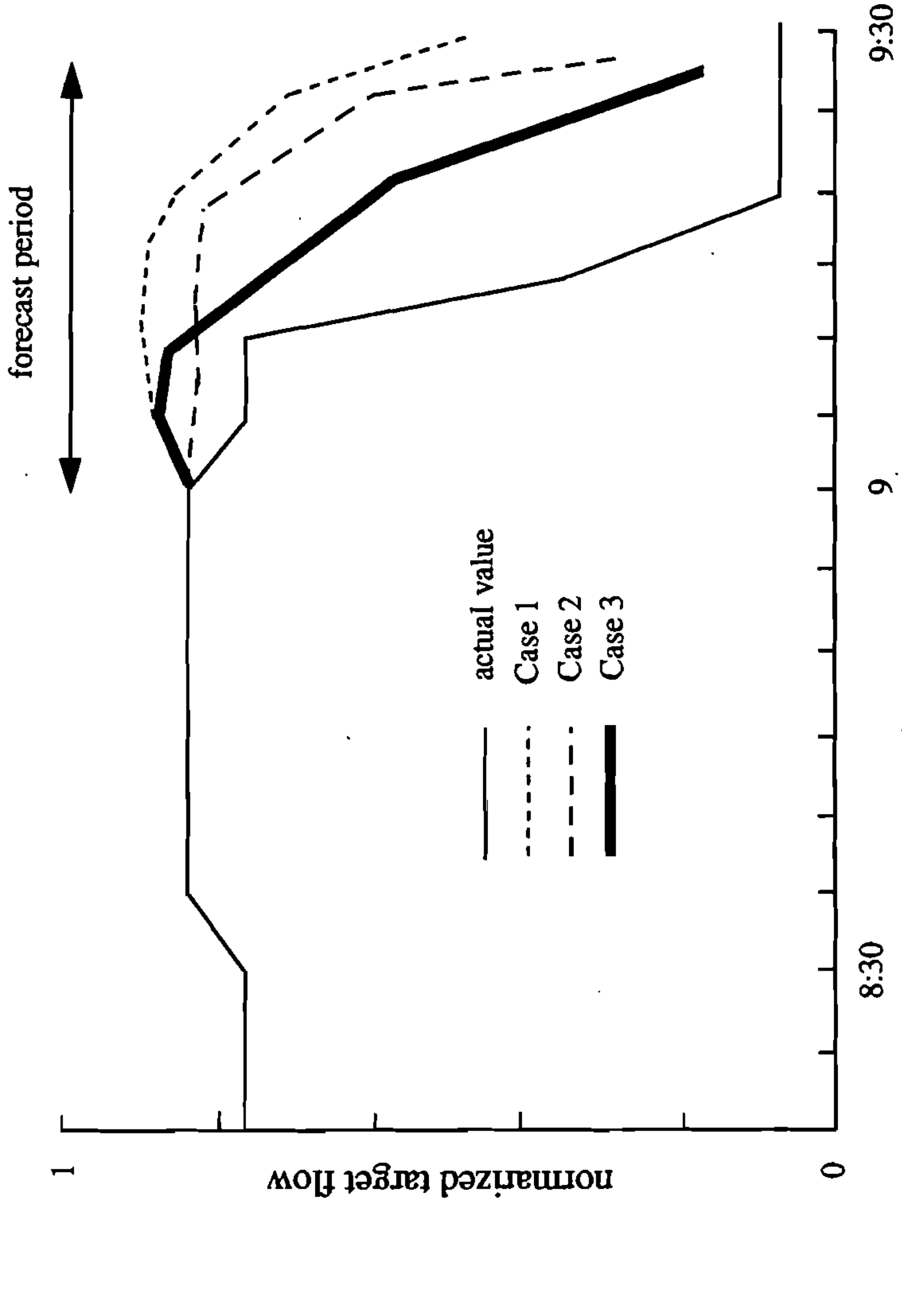

FIGURE 6 Traffic Flow Prediction Using Three Different Learning Rates

it is critical to realize a relationship between velocity, traffic density, and flow rate. A dimensional analysis of the variables yields the following relationship:

$$Q\ (vehicles/hr.) = U\ (mile/hr.) * K\ (vehicles/mile)$$

where

U = *velocity (mile/hr.)*
K = *traffic density* = K *(vehicles/mile), and*
Q = *flow rate (vehicles/hr.)*

In addition to this basic equation of flow, a relationship between speed and density is needed [Homburger, 1982]. As the number of vehicles on a roadway increases to a congestion level, the average velocity of traffic will decrease. In this simulation, a linear speed-density model is used as the basis of emulating traffic streams as they depart Atlanta Fulton County Stadium after a special event. The model is as follows:

$$U = U_f\,(1 - K / K_j)$$

where:

Uf = *free speed or speed at free flow conditions (approximated as 5 mph above the speed limit)*
Kj = *jam density at which all vehicles are stopped (vehicles/mile)*

$$U = U_f\,(1 - K / K_j)$$

It should be noted that both these parameters are unique to each section of road in the simulation. Although this speed-density model is simplistic, it does have good correlation with field data.

Once a speed-density model has been chosen, a speed-flow model can be derived:

Given: $Q = UK$ and $U = Uf\,(1 - K / Kj)$

Then: $U - Uf = -Uf * (K / Kj)$ or $K = Kj * (1 - U / Uf)$

By substitution:

$$Q = UK = U * K_j\,(1 - U / U_f) = K_j\,(U - U^2 / U_f)$$

From these equations, flow rates departing each intersection can be calculated. However, when some traffic is turning, compensation factors must be included to approximate the reduction in flow rate.

In addition to basic traffic flow analysis, the simulation also models shock waves in traffic streams. A shock wave is the propagation of different densities and velocities within a stream, due to signal lights, bottle necks, accidents, etc. For example if a traffic light turns red, all vehicles on the roadway controlled by the intersection will not stop instantaneously; rather a stopping wave will propagate backwards from the traffic light through the controlled roadway. This wave will proceed with a certain velocity and all traffic within this wave will not be moving, while traffic outside of the wave will still be moving. Similarly a starting wave will propagate backward throughout roadway traffic when a signal light changes from red to green.

The stopping wave equation is:

$$Uw = Uf(1 - (N + 1)] = -Uf * N$$

The starting wave equation is:

$$U_w = -(U_f - U2)$$

Where:

Uw = wave velocity (miles/hr.)
U2 = 1/2 final velocity (miles/hr.)
N = K/Kj, and
K = nominal density of traffic on the road.

The above equations are used solely to determine the rate at which vehicles flow from a source to a sink. In conjunction with the flow rates, the shock wave equations determine any delay times before flow can proceed.

The function of the simulation is to determine the amount of time necessary to move traffic from finite sources to infinite sinks by manipulating the traffic signals along the roads surrounding the stadium. In this instance the finite sources are parking lots and the infinite sinks are the interstates and certain roads.

The jam density and the free velocity for each section of roadway relevant to movement away from the stadium was approximated. These parameters are not only critical in calculating the flow rates from each section of road, but the jam density is also the capacity of the road.

Traffic movement starts from the parking lots to the adjacent streets. Once traffic is on these adjacent streets it is passed on to other streets, this process continues until traffic has reached an infinite sink. Flow from a source to a sink will proceed only if the sink is not at capacity and the signal light controlling the flow allows for movement; flow rates from parking lots are approximated to equal to the flow rate from a small street and will continue as long as the exit is not blocked by congested traffic on the adjacent road. At intersections where traffic turns, the flow rate is distributed, based on pre-defined percentages, to a variety of destinations.

A one second resolution local clock is used to update all calculations. Each second the simulation loops through all calculations to determine densities, flow rates, and velocities. Exploiting the configuration of the signal lights, the simulation is able to determine where shock waves are occurring and their velocities. Additionally, the simulation determines from which sources to which sinks traffic will move and updates the number of vehicles on each section of road.

RESULTS

The graphical interface displays the current traffic flows, capacities, and potentials for each street segment on a high resolution color monitor for viewing by the user. An example of the neural network traffic signal controller display for the Georgia Dome and Omni sports arenas is shown in Figure 7. North, south and east of the Dome and the Omni are the parking lots for spectators attending arena events. Each lot has a number indicating the number of vehicles currently in the lot. For example, the lot north of the Dome currently contains 782 vehicles.

Traffic data was generated using a discrete traffic simulator [Homburger, 1982] of downtown Atlanta. This area was selected because it is the site for the 1996 Summer Olympics and presents the most challenging traffic management problem the United States will witness in the 1990's. The total number of vehicles on each street segment is also indicated by a numeric value with the top number indicating traffic flow to the right and the bottom number traffic flowing to the left. A single number indicates a one way street.

Street lights are designated by a dash in an intersection with the flow of traffic (e.g., the green light) moving in the same direction as the dash. For example, the signal light at the intersection of International and Techwood is currently red on International, but green on International at its intersection with Spring Street. The clock in the top right hand corner displays the actual travel time the simulation has been running (e.g. one minutes in this example). Also note that all the lights on Northside Drive are green to allow traffic to flow smoothly.

Figure 8 shows the map one minute later. The lights on International Avenue at Techwood Drive have both changed to allow traffic to flow in the opposite directions. The lights on Northside have also changed for the same reason.

Figure 9 indicates the flow of traffic after ten minutes of driving time. This example has assumed that the only traffic flow in the area was from the arena parking lots. This was done to illustrate multiple venue traffic flow interaction. The figure shows that within ten minutes after the Dome and Omni events have ended traffic is efficiently being disbursed.

The graphical user interface has been designed to provide transportation engineers with several levels of abstraction during system operation. Figure 10 is a display of a larger area of downtown Atlanta, but with only primary streets indicated. The Fulton County Stadium area previously shown can be seen in less detail in the bottom right portion of this figure. ATMS operators may select the granularity of display they wish to view based upon their current interests.

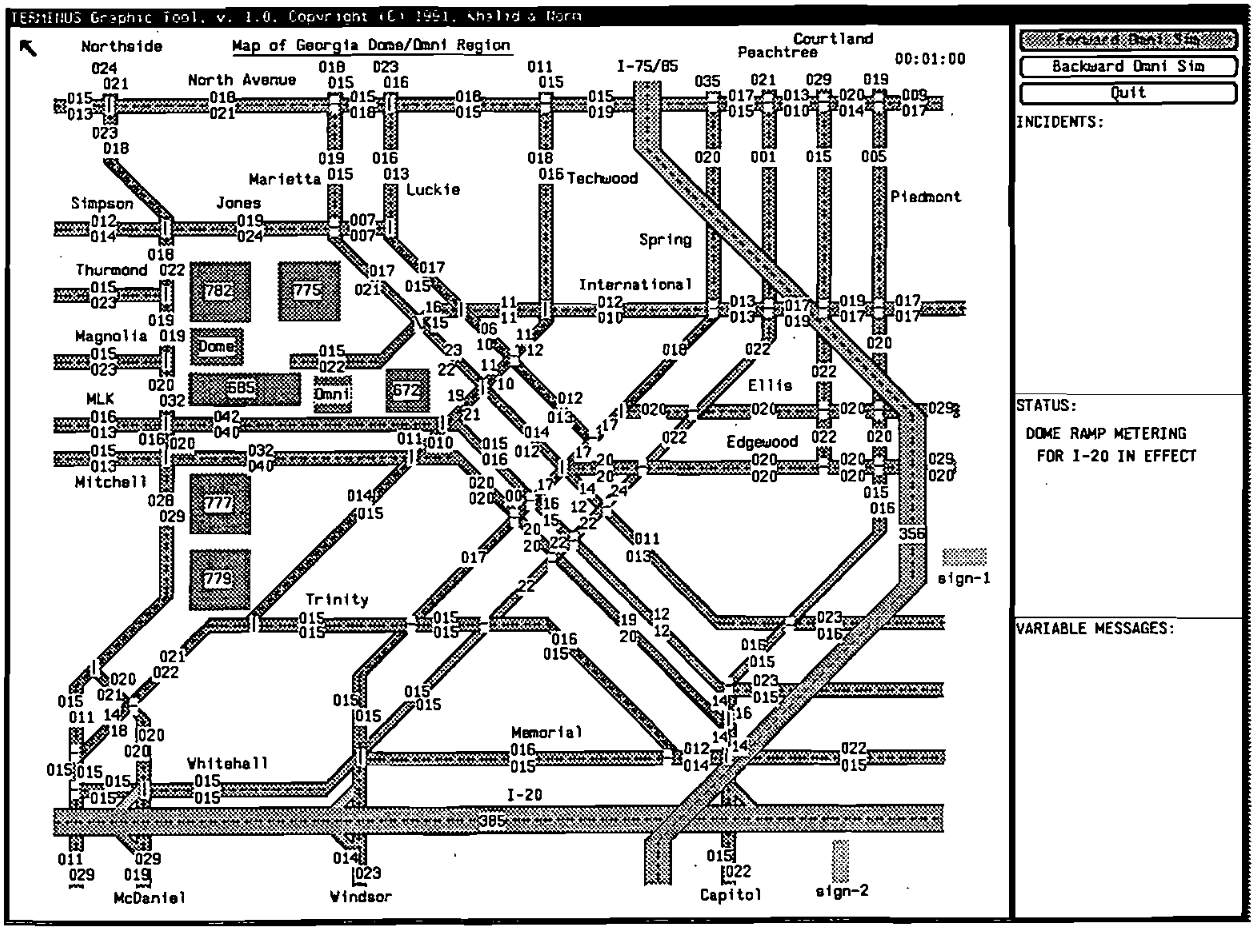

FIGURE 7 Example Neural Network Traffic Signal Display

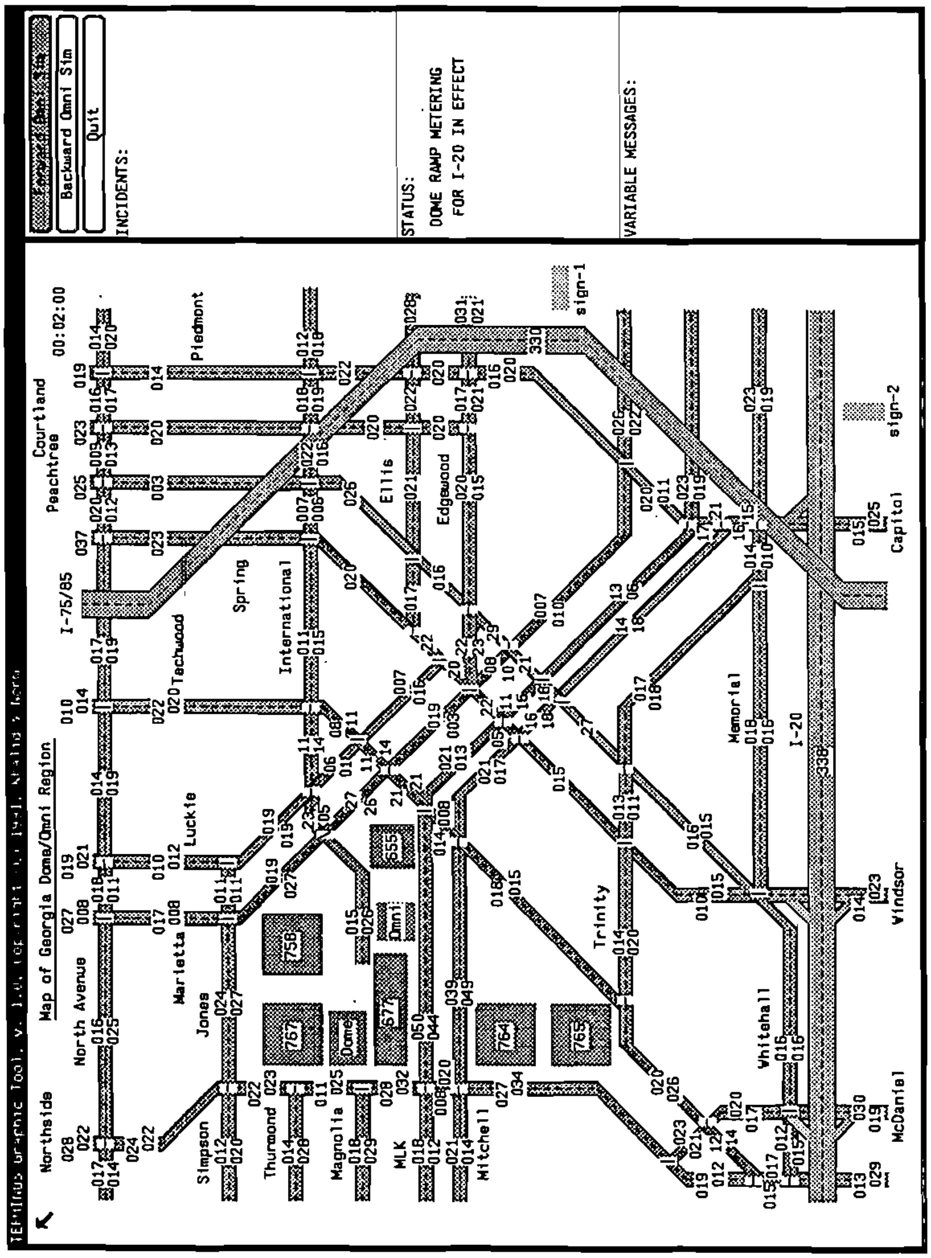

FIGURE 8 Traffic Flow After Two Minutes of Simulation

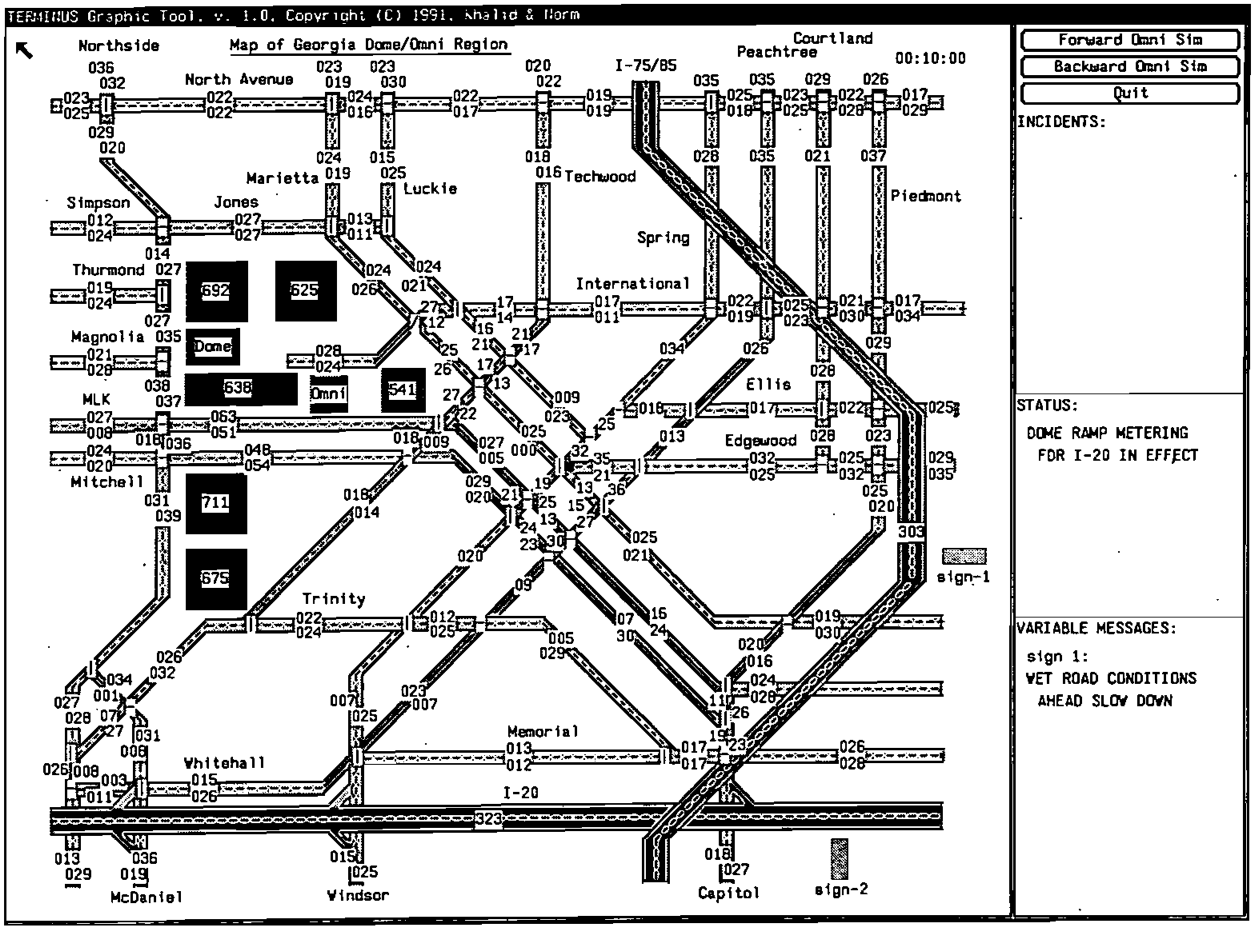

FIGURE 9 Traffic Flow After Ten Minutes of Simulation

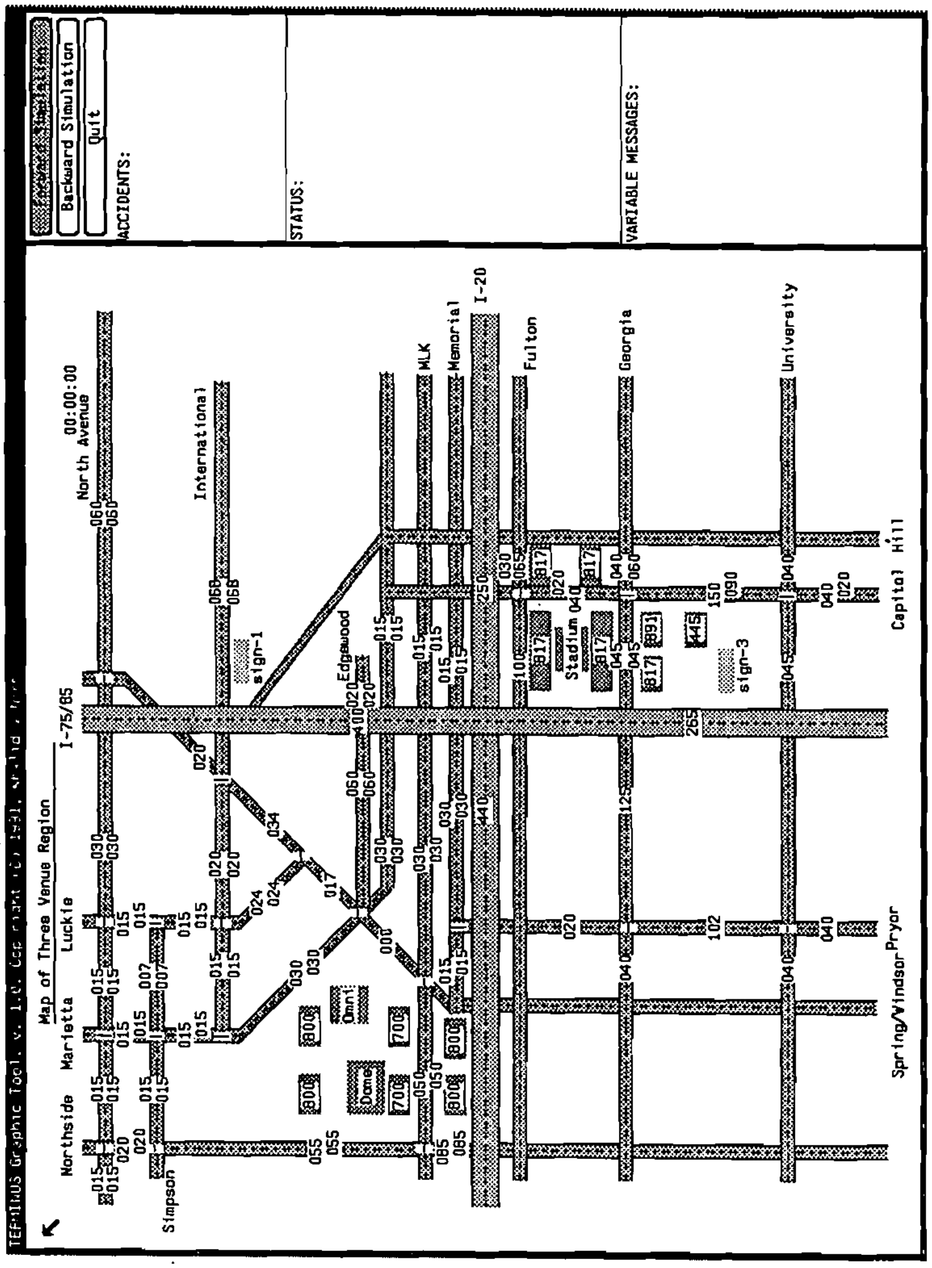

FIGURE 10 Omni, Georgia Dome, and Stadium Display Configuration

SUMMARY

Two applications of neural networks in advance traffic management have been presented. The intelligent traffic signal control has been developed and applied to several special case traffic situations including the multiple venue traffic congestion anticipated during the 1996 Olympics in Atlanta. The traffic congestion forecasting system has shown promise in predicting congestion based upon learning the factors that contribute to traffic jams and gridlocks.

Developed independently, research is continuing to integrate these systems into an ATMS blackboard architecture containing additional subsystems for incident detection, emergency vehicle management, ramp metering, and traffic monitoring. In this configuration results of predicted traffic congestion would be posted to a blackboard data structure. This action would activate the traffic signal control system which would attempt to divert traffic from the predicted congestion area. A more detailed interaction of the ATMS blackboard knowledge sources can be found in [Gilmore et al., 1994]

Research is currently underway to expand the capabilities of both of these systems. The initial Hopfield energy function has proven to be a valuable optimization function, but the creation of intersection specific energy functions (four way intersections, three way intersections, four way stop, etc) associated with their representative neurons promise to improve traffic flow performance. Though capable of learning behavior, backpropagation training is a time consuming task for large (5000+ link) systems. Efforts are underway to train on smaller networks with the goal of transplanting this behavior knowledge to larger networks. Research has shown reductions in learning time of over 67% in some cases.

REFERENCES

Anderson, J.A. and Rosenfeld, E., editors, *Neurocomputing: Foundations of Research*, MIT Press, Cambridge, Massachusetts, 1988.

Canu, S., Sobral, R. and Lengelle, R., "Formal Neural Network as an Adaptive Model for Water Demand," International Neural Network Conference-Paris, July 1990, Volume 1, p. 131–136

Gilmore, J.F. and Elibiary, K.J., Forbes, Harold C. and Payne, Kevin A., "TERMINUS: A Knowledge-Based Traffic Management System," submitted to Intelligent Highway Vehicle Systems Journal, Langhorne, PA 1994.

Homburger, W.S., *Transportation and Traffic Engineering Handbook*, editor, Institute of Transportation Engineers, Washington, D.C., 1982.

Hopfield, J.J., "Neural Networks and Physical Systems with Emergent Collective Computational Abilities," Proceedings of the National Academy of Sciences, Volume 79, p. 2518–2577, 1982.

Hopfield, J.J. and Tank, D.W., "Neural Computation of Decisions in Optimization Problems," Biological Cybernetics, Volume 52, 1985.

Robertson, D.J. and Bretherton, R.D., "Optimizing Networks of Traffic Signals in Real Time—The SCOOT Method," IEEE Transactions on Vehicular Technology, February 1991, Vol. 40, No. 1.

Rowe, E., "The Los Angeles Automated Traffic Surveillance and Control (ATSAC) System," IEEE Transactions on Vehicular Technology, February 1991, Vol. 40, No. 1.

Srirengan, S. and Looi, C. K., "On Using Back-propagation for Prediction: An Empirical Study," International Joint Conference on Neural Networks-Singapore, 1991, Vol. 2., p. 1284–1289.