Dynamic Programming Approach for Online Freeway Flow Propagation Adjustment

Xuesong Zhou and Hani S. Mahmassani

An optimization framework for online flow propagation adjustment in a freeway context was proposed. Instead of performing local adjustment for individual links separately, the proposed framework considers the interconnectivity of links in a traffic network. In particular, dynamic behavior in the mesoscopic simulation is approximated by the finite-difference method at a macroscopic level. The proposed model seeks to minimize the deviation between simulated density and anticipated density. By taking advantage of the serial structure of a freeway, an efficient dynamic programming algorithm has been developed and tested. The experiment results compared with analytic results as the base case showed the superior performance of dynamic programming methods over the classical proportion control method. The effect of varying update intervals was also examined. The simulation results suggest that a greedy method considering the impact of inconsistency propagation achieves the best trade-off in terms of computation effort and solution quality.

Real-time dynamic traffic assignment (RT-DTA) systems are designed to evaluate alternative traffic control actions and provide information supply strategies by utilizing real-time traffic surveillance data. The internal traffic network representation (for example, a traffic simulation model) embedded in RT-DTA systems serves as a basis for evaluating the effectiveness of different management decisions. To ensure the reliability of such an evaluation, the RT-DTA system requires an important external supporting module to ensure the consistency between actual observations and the estimated state of a traffic network.

In the RT-DTA system (DYNASMART-X) presented by Mahmassani (*1*, pp. 104–135), a consistency-checking function was introduced to reduce the deviations between the selected state measures on each link. Doan et al. (*2*) proposed a more general framework for online monitoring systems in a traffic network. The following are potential sources of error in online traffic management applications: demand estimation, path estimation, traffic propagation, internal traffic model structure, and online observation. Accordingly, the online adjustment system can be divided into two major components, namely, a demand and path adjustment component and a traffic propagation adjustment component, which may be viewed as long-term and short-term update procedures, respectively. Doan et al. (2) also applied the proportion-integral-derivative (PID) control method, a classical rule-based technique in the process control field, in their proposed online monitoring system. The PID adjustment method seeks to restore local consistency of each link quickly but does not directly take advantage of information on interconnected links. However, in some implementations, it may be possible to indirectly reflect global coordination through the determination of nominal values that serve as a basis for local control. In fact, through transfer of flow across links, the control action applied to a given link could significantly affect consistency errors, and hence the required adjustment, on downstream links. Therefore, local adjustment without consideration of upstream and downstream interconnection may not be effective globally. Similarly, an error on a link (e.g., inconsistency between actual and estimated link density) might propagate to other links and increase in magnitude after certain time intervals. The systematic control of error propagation requires development and application of reliable online traffic propagation adjustment procedures.

This study is concerned with the development of an online traffic propagation adjustment approach designed to update the flow representation of a mesoscopic (MESO) simulator in a freeway corridor context. The approach consists of a global optimization framework and a corresponding efficient solution algorithm. After the background on traffic flow models involved in the online flow adjustment module, the optimization framework and related modeling issues are discussed. Next a dynamic programming approach for solving the online flow propagation adjustment problem in a corridor context is described. Finally, the performance of proposed methods is evaluated on the basis of a test case.

BACKGROUND

Here the traffic flow models, as well as the corresponding numerical computation schemes, that are involved in the model presented in the next section are introduced. Depending on the level of representation detail, traffic flow models have been categorized as macroscopic, mesoscopic, or microscopic. The most notable macroscopic model is the first-order model developed by Lighthill and Whitham (*3*) and Richards (*4*). In the first-order model, traffic flow can be described by three basic equations: the fundamental equation, the conservation equation, and the speed-concentration equation. The conservation equation can be derived from an analogy between traffic flow and a one-dimensional compressible fluid:

$$q = kv \tag{1}$$

$$\frac{\partial q}{\partial x} + \frac{\partial k}{\partial t} = g(x,t) \tag{2}$$

where q, k, and v are flow, density, and speed, respectively, and $g(.)$ is the net vehicle generation rate.

Department of Civil Engineering, ECJ 6.2, University of Texas at Austin, Austin, TX 78712.

A commonly used form for the speed-concentration relation specified in the first-order model is that of Greenshields:

$$V = V_f\left(1 - \frac{K}{K_j}\right)^{\alpha} \tag{3}$$

where

V_f = free-flow speed,
K_j = so-called jam concentration, and
α = coefficient with a typical value of 1.

Instead of using a static speed-density relation in the first-order model, Payne (*5*) proposed a higher-order continuum model, intended to capture the mean speed dynamics in traffic flow. It should be noted that the performance of second- and higher-order models compared with that of first-order models remains a subject of debate in the traffic science community.

The foregoing models describe traffic flow as a one-dimensional compressible fluid with partial differential equations (PDEs). Because it is difficult to obtain analytic solutions for these PDEs, many researchers have presented various finite-difference approximations to solve these equations numerically. The general procedure of the finite-difference methods includes two steps. First, time is discretized into small equal intervals and the highway facility is discretized into sections. After an initialization of traffic demand for a given simulation horizon, variables are updated according to a given speed-density relation. In addition, the macroscopic model needs to consider capacity constraints between two consecutive sections. A recent numerical study for different finite-difference schemes has been described by Zhang and Wu (*6*).

As part of a dynamic analysis framework to investigate the effect of commuter decisions, Chang et al. (*7*) developed a MESO model, adapted from an approach used in plasma physics, that represents traffic flow as discrete vehicle groups, or macroparticles. This model updates the movement of vehicles on the basis of prevailing local speeds determined according to a relation between average speed and concentration prevailing in a discrete segment. In contrast, microscopic models offer a more complicated representation by considering stimuli and responses among individual particles. Among the better-known microscopic models is the car-following model proposed by Gazis et al. (*8*).

From a numerical computation standpoint, MESO models can be treated as mixed or hybrid models that combine elements of macroscopic and microscopic models. In other words, individual vehicle positions in the MESO model are updated according to macroscopic flow measurements. Usually, a modified Greenshields equation is specified to ensure that vehicles under the jam concentration still can move forward. Moreover, the supply constraints in MESO models are considered in the following two ways. First, a capacity constraint ensures that the computed flow rate is less than a given maximal flow rate. Second, a physical available space constraint keeps the resulting density below the physically maximal density.

In general, a suitable traffic flow model in the simulator of an RT-DTA system should achieve a balance between data requirements and model accuracy. With a similar data collection effort to that of the first-order macroscopic model, the MESO model offers the added capability of keeping track of vehicle-related information, such as departure time and path. Although microscopic models allow more detailed representations of various driver behaviors, they require an onerous calibration effort. Since the MESO model meets the dual requirement of simple calibration and adequate representation, it has been integrated into major simulation-assignment-based DTA systems, such as the DYNASMART system proposed by Jayakrishnan et al. (*9*), as well as into later systems, such as DynaMIT (*10*).

ONLINE ADJUSTMENT FRAMEWORK FOR FREEWAY SECTIONS

In an RT-DTA system, the flow propagation adjustment module needs to continuously update the internal traffic flow representation to approximate real-world observations. Essentially, the internal traffic simulator can be viewed as a dynamic system consisting of a set of interacting units. From an optimal control standpoint, the online traffic propagation problem is to choose the control variable vector over a traffic network such that the state of internal flow representation is transformed from a given system state to a desired system state at certain time T. Consequently, the principal elements for establishing an online flow propagation optimal adjustment model are as follows: (*a*) the state variables and control variables of internal flow representation, (*b*) the transfer function describing the flow dynamic behavior on interconnected links, and (*c*) the performance index or criterion used to specify the desired state.

System Characterization

The system under study consists of several consecutive freeway links in a corridor. To capture traffic flow dynamics, the physical links are subdivided into smaller sections or segments. For simplicity, on-ramp and off-ramp volumes are not considered in the following illustration, with no loss of generality. The following assumptions are made: (*a*) the necessary surveillance data in terms of speed, density, and flow on each section are available at each observation time point, and there are no online observation errors coming into the online flow adjustment modules; (*b*) the origin–destination demand and paths are assumed to be known and fixed during each online flow adjustment interval; and (*c*) the simulation time interval in the built-in simulator and the observation time interval are equal. Typically, the observation time interval might be equal to or greater than the simulation interval. For instance, the simulation interval used in DYNASMART is 6 s, and observation time intervals may vary from several seconds to several minutes depending on the specific monitoring devices. The reason for making the third assumption is to simplify the system design by applying a synchronous horizon for the different modules.

To introduce the notation and variables used in the formulation, the sections are numbered from 1 to J, and the simulation time step is denoted as t, $t = 1, \ldots, T$. The simulated vehicles travel from Section 1 to Section J; Δt is the length of the simulation interval, and Δx is the section length; $K_{j,t}$, $V_{j,t}$ are the prevailing density and mean speed in Section j during the tth time step, and $Q_{j,t}$ is the transfer flow rate from Section j to Section j + 1 during the tth time interval $[t, t + \Delta t]$.

For Section j in a traffic simulator, the internal state vector is (K, Q, V). There are two important flow rates, the inflow $Q_{j-1,\ t}$ from the upstream section and the outflow $Q_{j,t}$ to the downstream section. The speed of vehicles moving on a section $V_{j,t}$ is the control variable available to manipulate the system state. As suggested in by Doan et al. (*2*), the performance index of each section is measured in terms of the density $K_{j,t}$. Moreover, perturbations to the simulation system are due to unobserved characteristics and randomness in actual traffic flow.

Constraints

The following constraints in the optimization model are intended to describe the dynamic behavior of the simulation state of the MESO model. It is noted that the MESO model can be considered as a finite-difference form of the fundamental equation (Equation 1) and the conservation equation (Equation 2). Thus, it reasonable to apply finite-difference equations directly to represent the internal flow updating procedure in the MESO simulator. It should be noted that the constraints below do not incorporate the speed-concentration equation (Equation 3), since the speed value in the current model is determined by the optimization solution rather than by the equilibrium condition. Although there are other approaches to describe the dynamics of a MESO simulator, such as the time series technique, the finite-difference form seems to provide the most straightforward representation.

First, Equations 4 and 5 describe the corresponding finite-difference form for PDEs (Equations 1 and 2). In addition, Expression 6 is used to model the supply constraint in the MESO model, as mentioned earlier.

$$M_{j,t} = K_{j,t}V_{j,t} \tag{4}$$

where $M_{j,t}$ is the vehicle flow ready to move from Section j to Section j + 1 during the tth simulation time interval.

$$K_{j,t+1} = K_{j,t} + (Q_{j-1,t} - Q_{j,t})\,\Delta t/\Delta x \tag{5}$$

$$Q_{j,t} = \min\{M_{j,t}, C_{j+1,t}, S_{j+1,t}\} \tag{6}$$

where $C_{j+1,t}$ is the inflow capacity, and $S_{j+1,t}$ is the maximal inflow rate allowed by the available space on the downstream Section j + 1 during the tth time interval.

Since the MESO model keeps track of all vehicle positions, available inflow space in Expression 6 can be calculated as the distance from the position of the last vehicle to the end of the section. Furthermore, $S_{j+1,t}$ can be obtained as the maximal inflow value producing maximum density in the available space. By introducing a constraint on the available physical space, this formulation ensures that the first-in-first-out property is adhered to in the vehicle propagation, which is a critical requirement in the modeling of DTA systems. If intersection control elements are included in the simulator, the finite-difference form of the link outflow constraint can be established in a similar manner.

Equations 3 through 6 depict the dynamic model of simulated flow on each section. In addition, to make sure that the control variable $V_{j,t}$ is bounded by the admissible control region, Equation 7 must be considered:

$$V_{j,t} \geq 0 \qquad V_{j,t} \leq V_f \tag{7}$$

It should be noted that the physical bounds for density and flow rate are implicitly satisfied by Expression 6.

Objective Function

In the previous section, the important variables were identified and the required constraints (i.e., the transfer function) were defined for the optimal adjustment model. Next the selection of an appropriate performance index, that is, the objective function in the mathematical model, is discussed. The optimization problem seeks to minimize the difference between the real-world traffic states and the internal state representation in the simulator over a given time horizon. Using density as the state variable, the objective function can be expressed as the deviation of actual traffic density and simulated density at some given time points in the future. However, the system at the current adjustment step (time step t) does not have the actual traffic density measure for the next time step ($t + 1$), so a natural way is to use an anticipated value given the current real-world observation at time t. To simplify the problem, only a one-step prediction is used as the desired state; that is, the decision horizon is equal to the observation time interval. Under the above-mentioned assumption that the simulation and observation time intervals are equal, the performance index can be specified as follows:

$$P = \sum_j (w_j |K_{j,t+1} - K'_{j,t+1}|) \tag{8}$$

where w_j is the weighting factor of Section j, and $K'_{j,t+1}$ is the anticipated density on Section j at time step $t + 1$.

The introduction of the weight w_j in the objective function allows the decision maker to assign different importance levels to different sections during the adjustment. In recognition of the fact that inconsistencies may propagate and amplify as traffic flows downstream from a given section, it is desirable to reduce the inconsistency as close as possible to the source. In a freeway, which has a clear sequential structure, the weight of an upstream section could be set to a larger value than the corresponding weight on downstream sections. A simple heuristic rule to generate these weights so as to favor upstream sections is

$$w_j = J + 1 - j \tag{9}$$

In Equation 8, the anticipated density $K'_{j,t+1}$ can be calculated by a macroscopic traffic flow model, which will involve the numerical approximation of PDEs with finite-difference functions. In this paper, only the simple center difference scheme is applied.

DYNAMIC PROGRAMMING SOLUTION ALGORITHM

Because the optimization problem under study has a special sequential structure, with serial sections along a corridor, it offers a natural application for dynamic programming in a real-time computation setting. The forward-recursive optimization function in the dynamic programming algorithm is expressed as follows:

$$f_j^*(Q_{j,t}, V_{j,t}) = \min_{v_{jt}} [f_{j-1}^*(Q_{j-1,t}, V_{j-1,t}) + P(Q_{j-1,t}, V_{j-1,t})] \tag{10}$$

where $f_j^*(Q_{j,t}, V_{j,t})$ is the minimum cost obtainable from Section 1 to Section j, given outflow $Q_{j,t}$ and adjustment speed $V_{j,t}$ for Section j at time step t, and $P(Q_{j-1}, t, V_{j-1}, t)$ is the performance index specified in Expression 8 in terms of flow rate and speed. The boundary condition is $f^*(Q_0, V_j) = 0$, for all feasible V_j satisfying Inequality 7, where $Q_{0,t}$ is the (assumed) known inflow rate on Section 1.

The optimal solution is the adjustment speed vector over a series of sections, given the inflow rates on the first section. Since the inflow rates $Q_{j,t}$ are continuous variables, they are discretized into different discrete levels corresponding to a finite set of states in dynamic programming.

The dynamic programming algorithm can be expressed in a stage-wise manner. The steps of the proposed algorithm are as follows:

Step 1. Initialize all the cost vectors:
Set $f^*1(Q_{j,t}, V_{j,t}) = 0$ for j = 1, and set $f^*_j(Q_{j,t}, V_{j,t}) = \infty$ from Section 2 to Section J at current time step t, where $Q_{j,t}$ is the inflow level, which ranges between 1 and the maximal flow rate Q_{max}.
Read the given initial traffic demand on Section 1.
Step 2. For each Section $j = 1$ to J, do the following:
Generate the predicted density of Section j from the real-world observation at time step t.
For each arrival level $q = 1, \ldots, Q_{max}$ do the following:
For each speed level $v = 1, \ldots, V_f$ do the following:
Update the departure flow rate and density measure using Equations 4 and 5.
Compute the available space flow rate $S_{j,t}$ with given v.
Check the supply constraint (Expression 6) on Section j; if the constraint is not satisfied, update the current cost function $f^*_j(Q_{j,t}, V_{j,t}) = \infty$.
Apply the simulated density and the predicted density values into Expression 8, and use the recursive function (Equation 10) to update the best minimal function value $f^*_j(Q_{j,t}, V_{j,t})$ for a given inflow rate on Section j.
EndFor
EndFor
EndFor
Step 3. Find a global optimal decision for all $V_{j,t}$ on Section J, which is the minimal value $f^*_j(Q_{j,t}, V_{j,t})$ with $j = J$. Then search the sequence of adjustment speeds by backtracking the cost function value from the last Section J.

It should be noted that the computation of the supply constraint (Equation 8) for Section j requires values for $C_j + 1$, t and $S_{j+1,t}$ corresponding to the next Section j + 1. This expression might invalidate the sequential structure in the dynamic programming approach. To handle this difficulty, the equation can be removed from the constraint set and included as a penalty term in the objective function. This change means that the corresponding constraint checking is performed for Section j + 1. In other words, if this expression is not satisfied, the cost function is set to an infinitely large value. Through this approach, the proposed model retains a clear dynamic programming solution algorithm.

The complexity of the foregoing algorithm is $O(J|Q_{max}||V_f|)$, where $|Q_{max}|$ is the number of transfer flow levels and $|V_f|$ is the number of possible speed levels. The typical values of $|Q_{max}|$ and $|V_f|$ can be 2,200 and 60, respectively. Therefore, even though the algorithm might be easy to implement, its computational requirements may be considerable. A possible approach to reduce the computation cost is to reduce the number of transfer flow levels and speed levels by introducing a coarser discretization scale, such as using only 100 levels for Q and 20 levels for V.

As mentioned earlier, the weight w_j assigned to upstream sections can be set to a very high level. A simple approach for carrying out this idea is to compute only the best scheme using the best objective function value (minimal inconsistency) from the previous section rather than to evaluate all the possible schemes (Q, V). Thus the new return function could be expressed as follows:

$$f^*_{(j)} = \min_{Q_{j,t}, V_{j,t}} [f^*(j-1) + P(Q_{j-1,t}, V_{j-1,t})] \quad (11)$$

where $f^*(j)$ is the minimum cost obtainable for Section 1 to Section j, with a corresponding sequence of outflow $Q^*_{j,t}$, and adjustment speed $V^*_{j,t}$ for Section 1 to j at time step t.

The corresponding algorithm can be modified as indicated below. There are now only two loops for each section and for each speed level, so the complexity of the heuristic algorithm reduces to $O(J|V_f|)$. This approach can be viewed as a greedy algorithm because it only processes the best result from the last section. The modification steps are as follows:

Step 1. Initialize all the cost vectors:
Set $f^*_j(Q_{j,t}, V_{j,t}) = \infty$ from Section 1 to Section J at current time step t, where $Q_{j,t}$ is the inflow level between 1 and Q_{max}.
Read the given initial traffic demand on Section 1.
Step 2. For each Section $j = 1$ to J do the following:
Generate the predicted density of Section j from the real-world observation at time step t.
Read the inflow rate $Q^*_j - 1, t$ with respect to the $f(j-1)$ obtained from last Section 1 and do the following:
For each speed level $v = 1, \ldots, V_f$ do the following:
Update the departure flow rate and density measure using Equations 4 and 5.
Compute the available space flow rate $S_{j,t}$ with given v.
Check the supply constraint (Equation 6) on Section j; if the constraint is not satisfied, update the current cost function with speed v as infinite. Apply the simulated density and the predicted density values into Expression 8 and use the recursive function (Equation 11) to update the minimal cost function value $f^*(j)$ for all possible $V_{j,t}$ on Section j.
EndFor
EndFor
Step 3. Select $f^*(J)$ as the final solution. Then obtain the sequence of adjustment speeds by backtracking the cost function value from the last Section J.

NUMERICAL EXPERIMENTS

Numerical experiments are designed to evaluate the foregoing online flow propagation adjustment approach by comparing the results with those of an analytical base case. The analytical base case is used in preference to a simulated base case because it was desired to test the influence of the finite-difference equations used in the optimization model. If simulation results are used as a base case, the true performance of the finite-difference approximation will be difficult to identify. In the current study, a shock wave situation is tested on a 2.5-mi, three-lane pipeline freeway with five sections ($\Delta x = 0.5$ mi). The traffic demand at the first upstream section is uniformly distributed with 1,462.5 vehicles per hour per lane. With a free-flow speed of 60 mph and jam density of 160 vehicles/mi, the initial density is 30 vehicles/mi. In the last downstream section, the capacity is reduced to a certain level (emulating an incident situation), creating a bottleneck with a density of 90 vehicles/lane-mi, which remains over the whole study horizon. The simulation time interval and observation time interval are set to 6 s. The propagation speed of the shock wave is determined as follows (*11*):

$$W = \left(\frac{Q_d - Q_u}{K_d - K_u} \right) \quad (12)$$

where W is the shock wave speed in miles per hour, and the subscripts u and d indicate the upstream and downstream regions of the shock wave.

The following discussion focuses on the performance of different online adjustment approaches. Consistency in an online context can

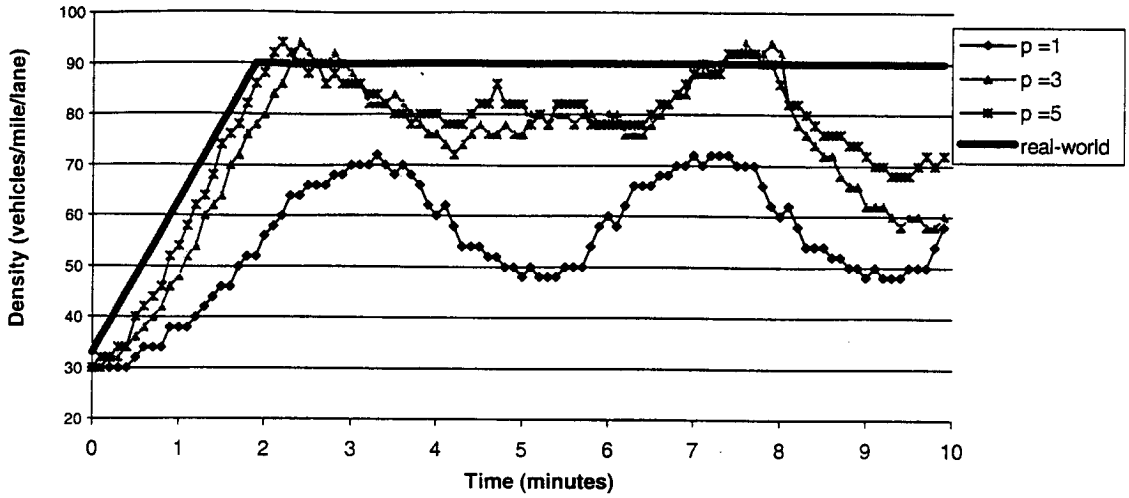


FIGURE 1 Evolution of internal simulated density using proportion control on Section 5.

be measured by the deviation between real-world observations and simulation results. The analytical results in the above incident case are treated as real-world observations. The average deviation and maximal deviation in terms of density are used as two measurements to compare the performance of difference adjustment methods and related settings. In addition to the above measures, density oscillation is also examined in evaluating the relative performance of the different methods.

Effectiveness of Proposed Algorithm

The adjustment method is designed to update the internal traffic density to be as close as possible to the actual traffic flow pattern. The result for the proportional control is plotted in Figure 1, where the thick lines depict the analytical (real-world) results. It is noted that the proportion gain of 1 is not large enough to enable the simulated density to follow the real-world density. After testing of different proportion constants, the calibration results in Table 1 generate the best proportion gain as 5.5. Table 1 also shows that the respective performance of different parameter settings for propagation gains is significantly different. In order to show the overall performance for the entire system, the density changes for all five sections are plotted. As shown in Figure 2, the density of analytical results (real-world observations) on Section 5 starts with 30 vehicles/lane-mi up to 2 min, then jumps sharply to 90 vehicles/lane-mi at Minute 4, and remains at that level thereafter. Similarly, on Section 4, the real-world density begins the change from 30 to 90 at Minute 4 and stays at 90 vehicles/lane-mi after 6 min. Figure 3*a* shows that the best parameter value of 5.5 still fails to dampen the oscillations for the system. Since the first section is the only element that needs to be adjusted during the first 2 min, the proportional control rule can help the simulation track the change of actual flow quite well. After the first 2 min, the shock wave propagates to the downstream Section 4, so Sections 4 and 5 need to be adjusted at the same time. It is noted that the simulated density on Section 5 drops sharply from its desired value. The main reason is that the adjustment on Section 5 did not consider the decrease of inflow from Section 4, which is due to the reduced control speed on Section 4. Moreover, the continuing oscillation in the remaining period clearly shows that an adjustment method that is only based on local conditions may become problematic in a simultaneous control context.

Figure 3*b* shows the evolution of internal density on all five sections using the dynamic programming approach. It is clear that the internal flow representation on each section has been successfully adjusted to follow the real-world flow. Furthermore, the dynamic programming method significantly reduces the amplitude of the oscillation. Specifically, the average density deviation is decreased from 4.900 (proportion gain $p = 5.5$) to 1.856, or by 62.2%, and the maximal density deviation is reduced from 22 to 11.

Impact of Different Update Intervals

In the current study, the update interval refers to the simulation time interval and the observation time interval, both of which have the same length. The update interval plays an important role in determining the computation cost and simulation accuracy. The smaller the update interval, the greater the effort needed to update the internal simulation values. Table 2 shows the average deviation and maximal deviation obtained using different update intervals. The upper bound of feasible update intervals is 30 s because of the numerical computation stability condition that $\lambda = \Delta x / (V_f \Delta t) \geq 1$. The results suggest that the average deviation and maximal deviation for large

TABLE 1 Simulation Results for Different Proportion Gains

Proportion gain	1	2	3	4	4.5	5	5.5	6	7
Average deviation	16.336	8.224	6.868	6.044	5.376	4.984	4.900	7.856	8.488
Maximal deviation	42	26	32	26	24	22	22	34	40

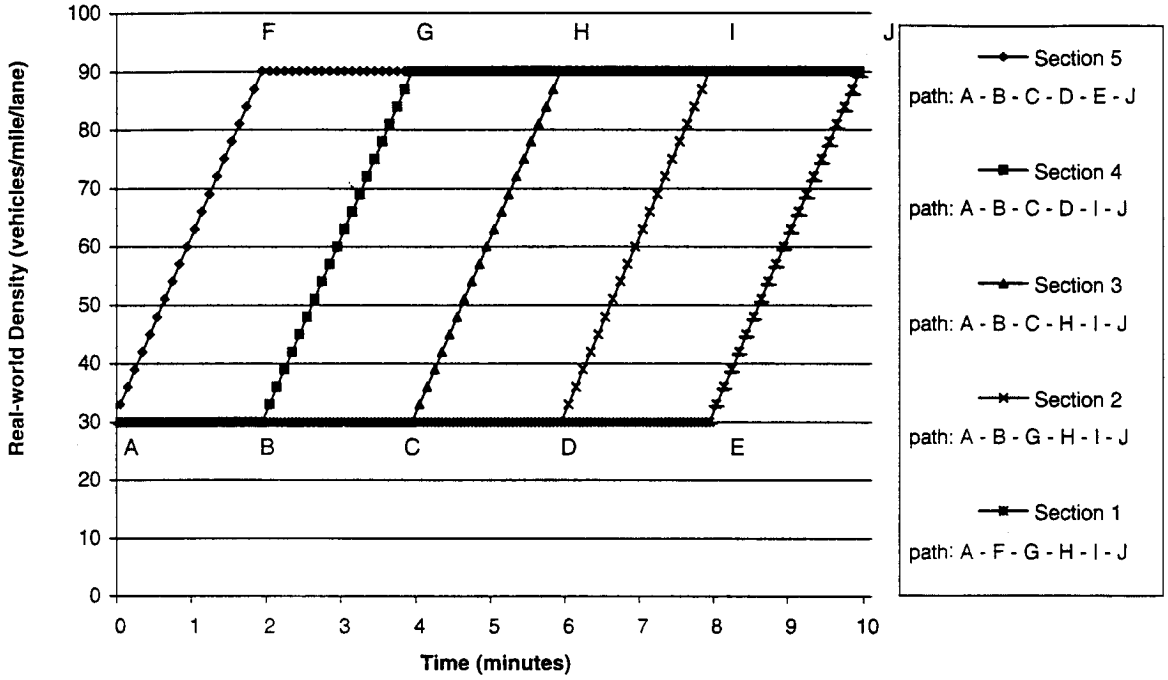


FIGURE 2 Evolution of real-world density using analytic results on five sections.

update intervals tend to be very high. The inconsistency between the actual flow and the internal simulator flow are due to two approximation mechanisms in the dynamic programming formulation, namely, the finite-difference approximation of the dynamics of the MESO model and the use of macroscopic model prediction for the "actual" traffic flow states in the next time step.

To isolate the impact of prediction errors in the proposed approach, a hypothetical case is generated that uses the analytic results as anticipated values. In other words, the approximation errors due to the predictor in the objective function do not need to be considered here, and only the finite-difference approximation to the MESO model accounts for the remaining deviation. The experiment demonstrates that the decrease in average deviation is very small for an update interval of 6 s. One possible reason is that even a simple macroscopic model can do a good job for such a short interval. For the longer intervals, the average deviations were reduced by about 50%, suggesting that the benefit of using a good predictor could be considerable. In order to improve the performance of the current model, it may be necessary to incorporate more advanced finite-difference schemes.

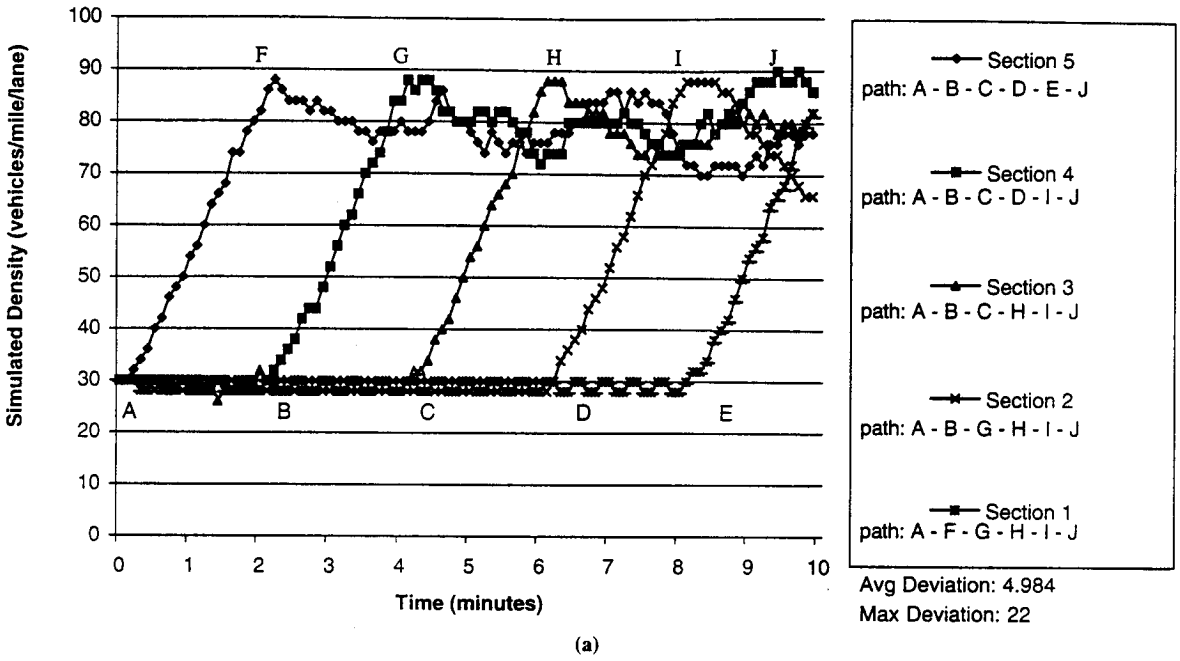
Impact of Different Solution Algorithms

Table 3 shows the performance of three proposed solution algorithms under different update intervals. In particular, the base case applies the dynamic programming algorithm with the same weight in the objective function. The second algorithm, using Expression 9 in the objective function, generally attains better performance relative to that of the base case. Furthermore, as an extreme version of different weights for each section, the greedy algorithm demonstrates quite satisfactory ability to reduce the inconsistency between observed situations and simulated data. The overall result shows that the performance of the online adjustment model is highly sensitive to the form of the objective function. Moreover, these experiments also highlight the importance of controlling error propagation in a traffic network. Taking the computation cost into account, the greedy method is recommended for the online flow propagation adjustment module.

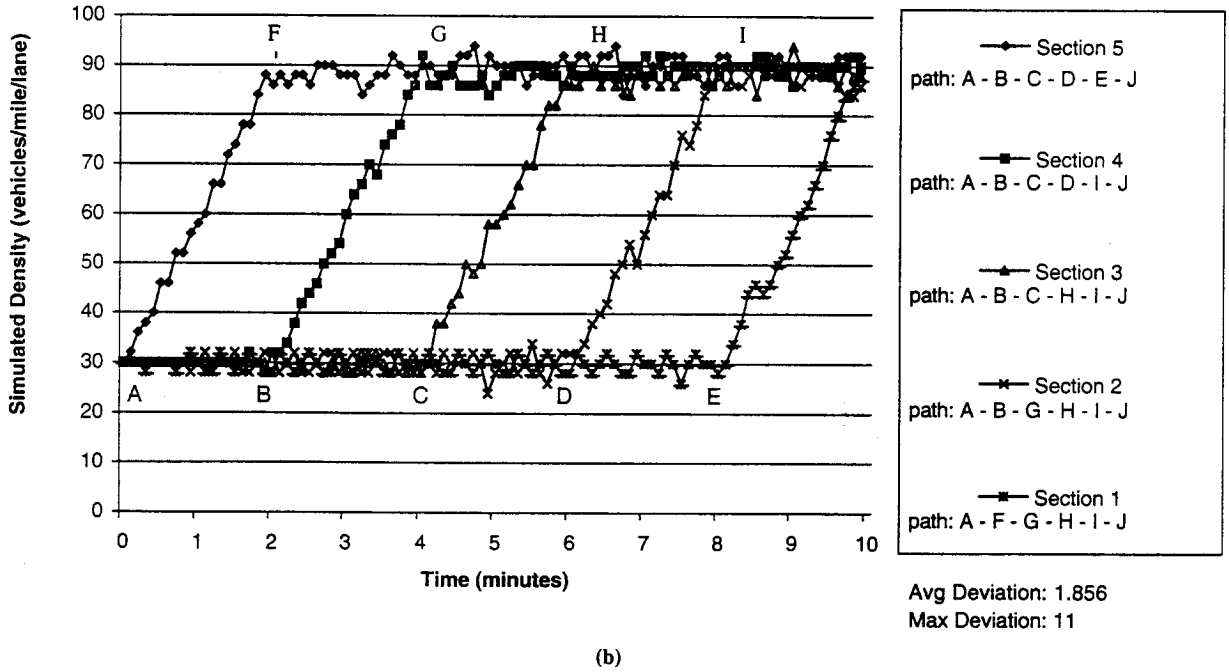
CONCLUSIONS

An optimization framework for online flow propagation adjustment in a freeway context is presented. Instead of performing local adjustment for individual links separately, the proposed framework considers the interconnectivity of links in a traffic network. In particular, the dynamic behavior in the MESO simulation is approximated by the finite-difference method at a macroscopic level. The proposed model seeks to minimize the deviation between simulated density and anticipated density. The latter measure is predicted by a macroscopic traffic flow model based on real-world observations. By taking advantage of the serial structure of a freeway, an efficient dynamic programming algorithm has been developed and tested. The experiment results compared with analytic results as the base case show the superior performance of dynamic programming methods over the classical proportion control method. The effect of varying update intervals is also examined. The simulation results suggest that a greedy method considering the impact of inconsistency propagation achieves the best trade-off in terms of computation effort and solution quality.

To further extend the capabilities of the above models in real-time traffic management systems, investigation of the following aspects is the subject of ongoing and future work. First, numerical experiments using field data are expected to give a better understanding of actual traffic dynamics. Second, it is necessary to investigate the



(a)



(b)

FIGURE 3 Simulation results using proportion control and dynamic programming methods: (*a*) evolution of simulated density using (*a*) proportion gain of 5.5 on five sections, (*b*) dynamic programming method on five sections.

TABLE 2 Simulation Results Under Different Update Intervals

Update interval length (seconds)	6	12	18	24	30
Average deviation	1.856	4.424	5.802	7.488	9.200
Maximal deviation	11	14	18	24	24

TABLE 3 Simulation Results Under Different Objective Functions

Update interval length (seconds)	6	12	18	24	30
Base case with the same weight for objective function					
Average deviation	1.856	4.424	5.802	7.488	9.200
Maximal deviation	11	14	18	24	24
Objective function with the different weights as (J+1-j)					
Average deviation	1.844	2.120	4.01	4.512	4.82
	(99.3%)	(49.9%)	(69.0%)	(60.3%)	(52.4%)
Maximal deviation	10	10	22	30	24
Greedy method					
Average deviation	1.708	3.104	2.563	4.32	6.24
	(92.0%)	(70.1%)	(44.1%)	(57.7%)	(67.8%)
Maximal deviation	10	11	16	26	26

numerical performance in other circumstances, such as in heavily congested traffic. Third, the optimization method for the online adjustment module should be extended from a freeway level to a network level, which would correspond to a more complex but more realistic process.

ACKNOWLEDGMENTS

This paper is based on a research project, Development of a Deployable Real-Time Dynamic Traffic Assignment System, sponsored by FHWA and managed by Oak Ridge National Laboratory. Additional support through the Southwest University Transportation Center is acknowledged. The work has benefited from the contributions of several current and former graduate students, including Nhan Huyhn, Yi-Chang Chiu, and Khaled AbdelGhany, and from discussions with A. Ziliaskopoulos and Y. Kang. The support and encouragement of Henry Lieu (FHWA) and Rekha Pillai are gratefully acknowledged.

REFERENCES

1. Mahmassani, H. S. Dynamic Traffic Simulation and Assignment: Models, Algorithms, and Application to ATIS/ATMS Evaluation and Operation. In *Operations Research and Decision Aid Methodologies in Traffic and Transportation Management* (M. Labbé, M. G. Laporte, K. Tanczos, and P. Toint, eds.), NATO ASI Series F: Computer and Systems Sciences 166, Springer, Berlin, 1998.
2. Doan, D. L., A. Ziliaskopoulos, and H. Mahmassani. On-Line Monitoring System for Real-Time Traffic Management Applications. In *Transportation Research Record: Journal of the Transportation Research Board, No. 1678*, TRB, National Research Council, Washington, D.C., 1999, pp. 142–149.
3. Lighthill, M. J., and G. B. Whitham. On Kinematic Waves II: A Theory of Traffic Flow on Long Crowded Roads. *Proceedings of the Royal Society, London*, Ser. A, Vol. 229, No. 1178, 1955, pp. 317–345.
4. Richards, P. I. Shock Waves on the Highway. *Operations Research*, Vol. 4, 1956, pp. 42–51.
5. Payne, H. J. Models of Freeway Traffic and Control. *Simulation Councils Proceedings Series: Mathematical Models of Public Systems*, Vol. 28, No. 1, 1971, pp. 51–61.
6. Zhang, H. M., and T. Wu. Numerical Simulation and Analysis of Traffic Flow. In *Transportation Research Record: Journal of the Transportation Research Board, No. 1678*, TRB, National Research Council, Washington, D.C., 1999, pp. 251–260.
7. Chang, G.-L., H. S. Mahmassani, and R. Herman. Macroparticle Traffic Simulation Model to Investigate Peak-Period Commuter Decision Dynamics. In *Transportation Research Record 1005*, TRB, National Research Council, Washington, D.C., 1985, pp. 107–121.
8. Gazis, D. C., R. Herman, and R. B. Potts. Car Following Theory of Steady State Traffic Flow. *Operations Research*, Vol. 7, No. 4, 1959, pp. 499–505.
9. Jayakrishnan, R., H. S. Mahmassani, and T. Y. Hu. An Evaluation Tool for Advanced Traffic Information and Management Systems in Urban Networks. *Transportation Research*, Vol. 2C, No. 3, 1994, pp. 129–147.
10. Ben-Akiva, M., M. Bierlaire, H. Koutsopoulos, and R. Mishalani. DynaMIT: A Simulation-Based System for Traffic Prediction and Guidance Generation. Presented at TRISTAN III, San Juan, Puerto Rico, 1998.
11. Haberman, R. *Mathematical Models: Mechanical Vibrations, Population Dynamics, and Traffic Flow*. Prentice Hall, Englewood Cliffs, N.J., 1978.

The authors are of course responsible for all results and opinions expressed in this paper.

Publication of this paper sponsored by Committee on Traffic Flow Theory and Characteristics.