# Rollback Approach for Demand Consistency Checking of Real-Time Traffic Network State Estimation Models

Ala Alnawaiseh, Khaled Abdelghany, and Ahmed Hassan

**The paper presents a real-time traffic network state estimation model with online demand consistency checking and updating capabilities. In contrast to reactive-based methodologies proposed in the literature, the model adopted a time rollback with a corrective actions approach. When an instance of inconsistency between the measured and estimated network state was observed, the model was allowed to roll back in time and promptly resimulated a predefined past period after the appropriate model's parameters were adjusted to minimize the observed inconsistency. A demand correction algorithm was developed and used for demand adjustment for each rollback period. The results of applying the developed model for a test bed network are presented. Results show that the approach improves the model's consistency with real-world observations.**

Traffic congestion has reached alarming levels in many metropolitan areas in the United States and abroad. With the limited ability to expand the physical capacity of the highway system to meet the growing demand, considerable effort has been devoted to developing advanced traffic network management systems to achieve better utilization of the existing network capacity. Advanced traffic network management systems are designed to provide traffic network managers with the capabilities to estimate the current network state, provide short-term prediction of the network congestion dynamics, and develop proactive traffic management schemes that can be deployed to alleviate recurrent and nonrecurrent congestion situations. Simulation-based dynamic traffic assignment (DTA) models have been proposed as the backbone of most systems (*1–4*). A network model is configured to run in real time to provide a minute-by-minute estimation of the current network conditions. Providing accurate estimation is critical to the success of the system's functionality. Accurate estimation defines the prevailing network conditions at the start of any performed network state prediction that evaluates the different proposed traffic management schemes before implementation. Before the estimation model is used in online operation mode, comprehensive offline model calibration is usually conducted to ensure that the model accurately represents a typical day of operation. This initial calibration effort involves estimating the dynamic origin–destination (O-D) demand table, travelers' route mode choice behavior during normal and congested situations, traffic flow models for the highways, and any other parameters specific to the model used. Nonetheless, when the limited data that are usually available for offline calibration are considered, this effort could result in an inaccurate estimate of the model parameters, an inaccuracy which limits the model's ability to provide estimation results that are consistent with the observed real-world conditions. Therefore, consistency checking between the model estimation and the observed real-world conditions needs to be performed frequently. If a state of inconsistency is detected at any time instance, the appropriate model parameters are promptly adjusted to minimize such inconsistency.

Traffic networks, as well as their simulation models, are dynamic systems with memory. An observed network state at any time instance is the evolution of the network's past states. For example, an observed flow breakdown on a highway section could be the result of flow instability downstream of that section in a previous period. Similarly, a flow rate along a highway link is a function of the number of vehicle trips loaded into the network during some past period and the travel speeds experienced by these vehicles along their routes until they reach this link. Thus, for a simulation model to yield a desired state, the model's parameters need to be corrected for a certain past period to replicate the model's past states that would eventually evolve to the desired current state. Consequently, following the rollback approach, if a state of inconsistency is detected, the model's parameters are adjusted for a predefined horizon. The model is then reset in time and is allowed to resimulate promptly this past horizon until it synchronizes again with the real-world clock. The goal is to eliminate the detected inconsistency between the simulated and the real-world conditions for the current time instance and, it is hoped, also for the future horizon. Alternatively, the observed inconsistency should be accepted and used to guide the model adjustment starting at the current time. In that case, until the effect of these adjustments takes place, the simulation model is expected to represent the observed real-world conditions inaccurately, an inaccuracy that could significantly affect the accuracy of any performed predictions and the effectiveness of the generated traffic management schemes during this period.

Several research contributions toward the development of the theoretical and algorithmic aspects of the online consistency checking problem have been reported during the past two decades. Because of the difficulty associated with developing one comprehensive approach that could be used to recognize and simultaneously fix all error sources that contribute to a state of model inconsistency, these research contributions have focused on adjusting the model for only one error source. Developed consistency checking methods could

Department of Civil and Environmental Engineering, Southern Methodist University, P.O. Box 750340, Dallas, TX 75275-0340. Alternate affiliation for A. Hassan: Public Works Department, Faculty of Engineering, Cairo University, Giza 12613, Egypt. Corresponding author: K. Abdelghany, khaled@lyle.smu.edu.

*Transportation Research Record: Journal of the Transportation Research Board, No. 2467,* Transportation Research Board of the National Academies, Washington, D.C., 2014, pp. 30–39.
DOI: 10.3141/2467-04

generally be classified into (*a*) traffic flow propagation adjustment methods or (*b*) O-D demand and path adjustment methods.

As an example of the first class of methods, Doan et al. (*5*) and Kang (*6*) presented a framework for a monitoring system that aims at maintaining consistency between the simulation model and the actual network by applying online and offline adjustments. The proportion-integral-derivative control method is applied to restore local flow consistency on the observed links. However, the method does not directly take advantage of available information on interconnected links. To address that limitation, Zhou and Mahmassani presented a dynamic programming formulation for the online flow propagation adjustment problem (*7*). The idea was to develop a flow propagation adjustment procedure that reflects global coordination between interconnected freeway links through the definition of optimal global values that serve as the basis for local adjustments.

As examples for the second class of demand adjustment methods, Mahmassani et al. (*1*) and Kang (*6*) proposed a real-time consistency updating module that heuristically adjusts the demand level according to the detected discrepancy between simulated and observed link density. Peeta and Bulusu presented a mathematical programming approach for ensuring the online consistency of a simulation-based DTA model (*8*). The program seeks to minimize deviations between real-time traffic measurements and predicted network states. The O-D demand adjustments are determined with an iterative approach that integrates a deterministic DTA problem and a least squares problem in a stage-based rolling horizon framework. Zhou and Mahmassani presented a reactive approach to determine O-D demand adjustments that minimize observed deviations between the simulated and real-world conditions (*9*). Following the reactive approach, the observed inconsistency at the current time interval determines the required adjustments that are applied for the next interval. For example, if the current number of vehicles in the used simulator is lower (higher) than the observed measures, positive (negative) adjustment is applied to the demand input for the next interval. The problem is formulated as a linear quadratic optimization program with inequality constraints with efficient solution algorithms. While the approach is shown to reduce inconsistency for most of the tested cases, the approach targets only short-term corrections as the used transition equation does not consider the predicted demand and the exiting vehicles. In addition, the approach is expected to perform poorly during periods in which the demand is transitioning from peak to off-peak demand or vice versa, as the value of the observed inconsistency could lead to a wrong correction. Zhou and Mahmassani stated that an ideal approach for the online demand adjustment problem is to reset the simulator and to resimulate the network traffic conditions by using the posteriori demand estimate (*9*). While this approach is not supported by any analysis, the authors argued that this approach could be computationally intensive and it was not considered as a possible approach.

This study targets the problem of online O-D demand consistency checking and updating for real-time traffic network management systems. The research was motivated by the need to develop a real-time traffic network simulation model that is consistent with the observed real-world conditions. As mentioned earlier, failure to estimate the current network state conditions accurately is expected to affect the accuracy of the performed predictions and hence the effectiveness of any developed traffic management schemes that are derived on the basis of these predictions. The paper adopts a rollback approach that allows backtracking of the simulation model after the time-dependent O-D demand is adjusted on the basis of the detected discrepancies between the simulation results and the real-world measurements. The O-D demand adjustment problem is formulated as a least squares problem, which minimizes the deviation between the simulated and measured time-varying vehicle counts on observed links. The program is solved by deriving an approximate linearization that can be efficiently solved to meet the system's real-time requirements. Two main contributions are attributed to this research. First, to the authors' knowledge, this paper is the first to introduce a backtracking strategy for consistency checking for real-time traffic network simulation models. It overcomes the limitations of reactive-based strategies that are usually limited to short-term demand corrections. Second, a novel O-D demand estimation methodology is developed. The methodology takes advantage of the structure of the traditional least squares formulation and adopts the separable programming approach to efficiently solve the problem. This paper is organized as follows. The next section formally defines the demand consistency checking and updating problem. The overall framework of the real-time traffic network estimation model with rollbacks is then presented. Next, the mathematical program that is used to solve the demand correction problem at each rollback is presented. The results of applying the developed model for a test bed network considering different operation scenarios are also presented. Finally, conclusions and considered research extensions are discussed.

## PROBLEM DEFINITION

Given is a highway network $G(A)$, where $A$ is the set of links. A subset of these links $A' \in A$ is assumed to be covered by surveillance equipment that provides time-varying link state observations (e.g., vehicle count). The observed state $y_{at}$ for link $a \in A'$ during each observation interval $t \in T$ is given for all covered links, where $T$ is the number of observation intervals in the horizon of interest. A real-time simulation model is developed for this highway network. The corresponding state $\hat{y}_{at}$ for each link $a \in A'$ in observation interval $t \in T$ is estimated by the simulation model. The estimated link state $\hat{y}_{at}$ at any interval $t \in T$ is assumed to be a function of the values of some models' parameters during a past period $R \in T$. The length of period $R$ is expected to depend on the network topology and associated congestion dynamics. $\alpha_{k_{at}}^{t'}$ is defined as the value of parameter $k \in K$ at a past time interval $t' \in R$ that affects the state of link $a$ in the current observation interval $t$, where $K$ is the set of model parameters that can be calibrated. For example, $\alpha_{k_{at}}^{t'}$ could represent the number of vehicle trips between O-D pair $k$ in departure interval $t'$ that use link $a$ in time interval $t$.

The equation $M = \sum_a |y_{at} - \hat{y}_{at}| / \sum_a y_{at}$ is defined as the measure of inconsistency between the model and the real world in interval $t \in T$. Thus, as illustrated in Figure 1, for inconsistency $M \geq \delta$, the model parameters $\alpha_{k_{at}}^{t'}$ are adjusted to minimize this inconsistency, where $\delta$ is the maximum allowed state of inconsistency at any time interval. To avoid false inconsistency alarms, especially in the case of using short observation intervals, the difference between the estimated and observed states could be recorded and averaged over multiple time intervals. Thus, $M = \sum_t^{t-m} (\sum_a |y_{at} - \hat{y}_{at}| / \sum_a y_{at})/m$, where $m$ is a predefined number of past time intervals. The above defines the problem of consistency checking for real-time traffic network state estimation models that is studied in this paper. A solution framework is presented that adopts a rollback approach, which allows the real-time traffic network state estimation model to roll back for a predefined horizon, adjust the appropriate model parameters, and swiftly resimulate this horizon to make sure that the model is consistent with the observed real-world conditions. While the developed

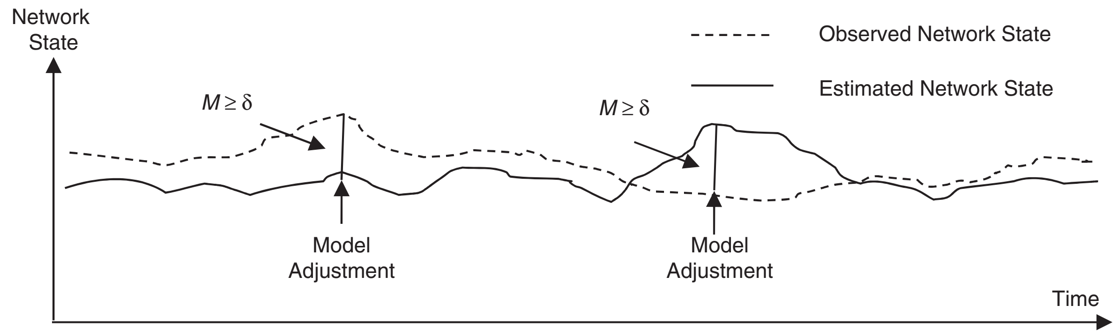



**FIGURE 1 Illustration of consistency checking for real-time traffic network state estimation.**

online rollback framework can be applied to any model parameters, this paper focuses on adjusting time-dependent O-D demand. Developing a methodology that can be used to simultaneously adjust the model's demand and supply parameters is considered as an extension of this research work.

## TRAFFIC NETWORK STATE ESTIMATION AND PREDICTION WITH ROLLBACKS

Figure 2 illustrates the overall framework of the implemented real-time traffic network state estimation and prediction. The framework adopts a rolling horizon approach, which integrates (*a*) a network state estimation module, (*b*) network state prediction module, (*c*) consistency checking and online adjustment module, and (*d*) demand prediction module. The network state estimation module is in the form of a real-time simulation-based DTA model capable of capturing the congestion dynamics resulting from the network's demand–supply interaction. The model is synchronized with the real clock and is used to provide an estimate of the current network conditions. The DTA simulation-based model, DIRECT, which was developed by researchers at Southern Methodist University, is used as the basis for the estimation module. The DIRECT model consists of several components, including (*a*) demand generation, (*b*) travel behavior, (*c*) shortest-path algorithm, (*d*) vehicle simulation, and (*e*) statistics collection. The DIRECT model accepts as input a time-dependent O-D trip table for a prespecified horizon. Each generated traveler is assigned a set of attributes including his or her trip starting time, generation link, final destination, and a distinct identification number. Information about the travelers' historical routes can also be used by the model, if available. Prevailing travel times on each link are estimated with the vehicle simulation component, which adopts a mesoscopic simulation approach. The model uses a set of measures that travelers may use to evaluate the different route options, including travel time, expected vehicle operation cost, and highway tolls. These measures are combined in a generalized cost formula that is used by the shortest-path algorithm to determine the set of superior route options for all O-D pairs. The activation interval for the shortest path algorithm (usually in the range of 3 to 5 min) is set such that the variation in network conditions is captured, while desirable computational performance is retained. Following a mesoscopic approach, vehicles move in the network subject to the prevailing traffic conditions until they reach their final destinations along the prespecified routes. If a traveler receives en route information and presumably complies with the provided information, the traveler's route is updated accordingly.

To ensure consistency between the estimation model and the real network, the estimation module receives a continuous data feed in the form of vehicle count and speed measurements for roadway links equipped with surveillance devices. As described above, the consistency checking module compares, for a predefined number of intervals, the simulation results with the received data. If the difference is greater than the defined threshold, the online calibration module is

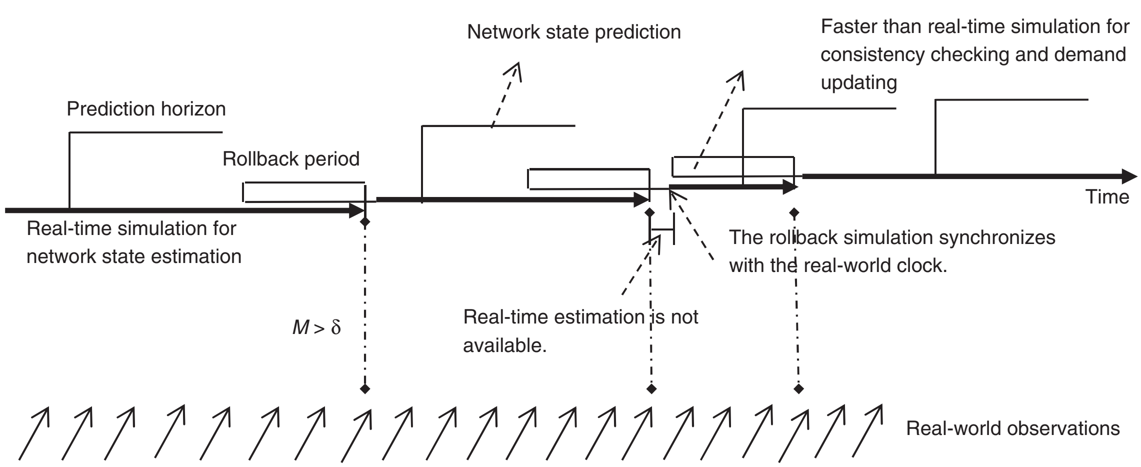



**FIGURE 2 Real-time traffic network state estimation and prediction with rollbacks.**

activated to adjust the model's parameters. For that purpose, as illustrated in Figure 2, a rollback period is defined and the simulation time clock is changed to the start of this rollback period. On the basis of the direction and magnitude of detected inconsistency between the estimated and the real-world measurements, the model's parameters are adjusted. With the new values of the model parameters, the simulation is allowed to run faster than real time until it catches up again with the real-world clock. Until the model is synchronized again with the real clock, the estimation results are unavailable. However, as shown hereafter, the period during which the estimation results are not available is expected to be less than the length of the prediction cycle. Thus, estimation consistency could be achieved without affecting the overall system functionality. On the basis of the observed difference between the estimated and observed vehicle counts on all links that are equipped with surveillance devices, the time-dependent zonal demand used by the model is corrected. As described in more detail in the next section, the demand correction problem is formulated in the form of a least squares problem, which minimizes the difference between the simulated and measured time-varying link counts for the rollback period.

The rollback approach involves several operations. First, any state of inconsistency should be detected, which entails (*a*) defining the appropriate state variables for comparison and (*b*) ensuring that any detected difference between the estimated and measured state is not the result of simulation noise. A complete discussion on the selection of state variables for real-time consistency checking can be found in Doan et al. (*5*). In this paper, the time-varying vehicle counts for all links that are equipped with surveillance devices are used as the state variables. As described above, to avoid false alarms, these state variables were recorded over few intervals. If the difference persists during these intervals, the correction module is activated. Second, the reset of the simulation model requires the definition of the network conditions at the start of the rollback period. This information is made available by storing network state snapshots at a predefined frequency (e.g., 1 to 5 min) while the estimation module is running. Thus, if no snapshot is available at the start of the rollback period, the start time of the rollback period is shifted to the time of the nearest available snapshot. Finally, the O-D demand correction requires accurate information on the time-dependent assignment matrix to map the time-dependent O-D demand correctly to the observed vehicle count on the links. The estimation module tracks the trajectory of each simulated vehicle. Such information is used to estimate the demand assignment matrix. If a rollback period is defined, the assignment matrix that defines this period is extracted for the application of the demand correction algorithm.

The demand correction methodology provides a corrected estimate of the time-dependent O-D demand matrix for the rollback period. It is assumed that information will be available on the demand transition function for all O-D pairs, which defines the expected demand value for a future period on the basis of the estimated value in the current interval. Thus, the demand values for all O-D pairs could be predicted for a defined future horizon. This predicted demand could be used by the estimation and prediction modules to improve the accuracy of their estimation and prediction results, respectively. The prediction module is periodically activated (e.g., every 5 to 10 min) to predict the network conditions for a predefined horizon (e.g., 30 min to 1 h). The prediction module consists of another instance of the network simulation model that is configured to run faster than real time. The estimation module provides a snapshot of the network conditions at the start time of each prediction horizon. This snapshot defines the current location, speed, and assigned route for all travelers in the network. Information on any incident in the network or changes in the network control setting or both is incorporated. Given the predicted demand, vehicles already in the network at the start of the prediction horizon and newly generated vehicles are simulated for the prespecified horizon. Facility-based and area-based measures of performance are continuously reported from the network state estimation and the prediction modules.

## ONLINE DEMAND CORRECTION METHODOLOGY

This section describes the demand correction methodology that was developed to adjust the time-dependent O-D demand for each rollback period. While considerable research work is devoted to the time-dependent demand estimation problem (*10–18*), few of the developed methodologies are targeting real-time operations (*19–22*). The methodology used in this research is a modification of the demand estimation model, which is developed by Hassan et al. to suit real-time operation (*15*). The methodology takes advantage of the structure of the conventional least squares error minimization formulation of the O-D demand estimation problem as presented in Cascetta et al. (*16*). The methodology adopts a separable programming approach to derive an approximate linear formulation of the problem, which can be efficiently solved to meet the system's real-time requirement. Assume that the network is divided into a set of zones $Z$. Also, the rollback horizon $R$ is divided into $R^d$ departure intervals and $R^s$ observation intervals. Traffic originates from origins $I \in Z$ to destinations $J \in Z$ during the different departure time intervals $\tau \in R^d$. Define $y_{at}$ as the observed vehicle count on link $a$ in observation interval $\in R^s$, $\hat{y}_{at}$ as the corresponding state for each link $a \in A'$ in observation interval $t \in T$, and $\hat{d}_{ij\tau}$ as the estimated demand between O-D pair $ij$ in departure interval $\tau \in R^d$. Also, define $P$ as the demand assignment matrix such that an element $p_{ij\tau}^{at}$ in this matrix represents the portion of vehicles observed on link $a \in A$ in interval $t \in r^s$ that belongs to the O-D pair $ij$ and departure interval $\tau \in r^d$. As mentioned above, this link–flow proportion matrix was generated by using the network state estimation module. The simulation-based DTA assigns the vehicle trips to routes and tracks their movements along the links of these routes until they reach their final destination. Thus, the link proportion values $p_{ij\tau}^{at} \in P$ are estimated for the horizon of the rollback period. Cascetta et al. proposed a formulation of the O-D demand estimation problem in the form of a least squares error minimization as follows (*16*).

$$\text{minimize} \sum_{a \in A'} \sum_{t \in R^s} \left( y_{at} - \hat{y}_{at} \right)^2 \tag{1}$$

subject to

$$\hat{y}_{at} = \sum_{i} \sum_{j} \sum_{\tau} p_{ij\tau}^{at} \cdot d_{ij\tau} \tag{2}$$

$$\hat{d}_{ij\tau} \geq 0 \qquad \forall i, j, \text{and } \tau \tag{3}$$

The program above consists of a quadratic objective function with linear constraints, which can be decomposed into terms such that each term includes only one variable that is represented by a convex function. This structure of the problem allows the use of the separable programming approach to solve the problem efficiently as explained in Bazaraa et al. (*23*). The idea is to solve an approximation of the problem through providing a piecewise linear approximation of the nonlinear terms. Given the maximum possible range $c_{at}$ of each decision variable $\hat{y}_{at}$ (i.e., the maximum capacity of the link) and dividing this range into $n$ equal intervals, the value $u^s_{at}$ of the variable $\hat{y}_{at}$ at interval $s$ is equal to $(s \cdot \Delta_{at})$, where $\Delta_{at} = c_{at}/n$.

The corresponding value of the nonlinear term $(y_{at} - \hat{y}_{at})^2$ at interval $s$ can then be numerically evaluated for all intervals in the range of $\hat{y}_{at}$. In other words, by plugging $u^s_{at}$ in its corresponding nonlinear term, the numerical evaluation $v^s_{at} = (y_{at} - u^s_{at})^2$ for each interval $s$ is obtained. Thus, with the new decision variable $\lambda^s_{at}$ used to indicate the optimal interval in the specified range of the variable $\hat{y}_{at}$, the mathematical program given above can be rewritten in the form of the following linear mathematical program.

$$\text{minimize} \sum_a \sum_t \sum_s v^s_{at} \cdot \lambda^s_{at} \tag{4}$$

subject to

$$\sum_s \lambda^s_{at} = 1 \qquad \forall\ a \text{ and } t \tag{5}$$

$$\sum_s u^s_{at} \cdot \lambda^s_{at} = \sum_i \sum_j \sum_\tau p^{at}_{ij\tau} \cdot \hat{d}_{ij\tau} \qquad \forall\ a \text{ and } t \tag{6}$$

$$\lambda^s_{at} \geq 0 \qquad \forall\ s, a, \text{ and } t \tag{7}$$

The optimal value of $\lambda^{s*}_{at}$ determines the optimal interval $s*$ for the variable $\hat{y}_{at}$. Given the convexity of each nonlinear term, the mathematical program yields either $\lambda^{s*}_{at} = 1$ for $s*$ and $\lambda^{s'}_{at} = 0$, $\forall\ s' \neq s*$ or $\lambda^{s*}_{at} = \alpha\ (s*) + (1 - \alpha)(s* + 1)$ and $\lambda^{s'}_{at} = 0$, $\forall\ s' \neq s*$ where $0 < \alpha < 1$. The solution of this mathematical program gives the optimal $\hat{y}^*_{at}$, $\forall\ a$ and $t$, and the corresponding optimal demand values $\hat{d}^*_{ij\tau}$, $\forall\ i, j$ and $\tau$. The approximation problem adds a number of decision variables $\lambda^s_{at}$, $\forall\ a, t,$ and $s$, which depend on the number of observed links in $A$, number of observation intervals in $r^s$, and number of discretization intervals $n$ used to approximate each nonlinear term. As illustrated hereafter, while increasing the value of $n$ provides a better approximation of the problem, it increases its size. Thus, the trade-off between the solution accuracy and the increase in the execution time needs to be examined carefully to choose a suitable value for the parameter $n$. The effect of the value of the parameter $n$ on the overall solution quality is presented in the next section.

## RESULTS AND ANALYSIS

This section describes the results of different experiments that are conducted to examine the performance of the developed real-time traffic network estimation model and the consistency checking module. As illustrated in Figure 3, the study area includes a section of the Dallas North Tollway surrounded by parallel and perpendicular arterials. The network consists of about 400 links and 150 junctions. A time-dependent demand pattern that represents 1 full day is considered by assuming 10-min departure time intervals. The demand pattern is set such that it represents typical morning and evening peak periods with heavy traffic along the freeway in both directions. Several detectors are assumed to be installed along the freeway facility and provide real-world observations at a 10-min frequency. To mimic a scenario of demand inconsistency, an offline simulation run was conducted with a different demand pattern to generate the vehicle count measurements. The real-time traffic network estimation system is configured such that the consistency checking module is triggered every 10 min. The inconsistency measure $M$ is computed as described earlier, and compares the measured and estimated vehicle counts as a ratio of the measured values and averaged over $m$ intervals. As mentioned earlier, if $M \geq \delta$, the simulation clock is rolled back for a period that is equal to $R$.

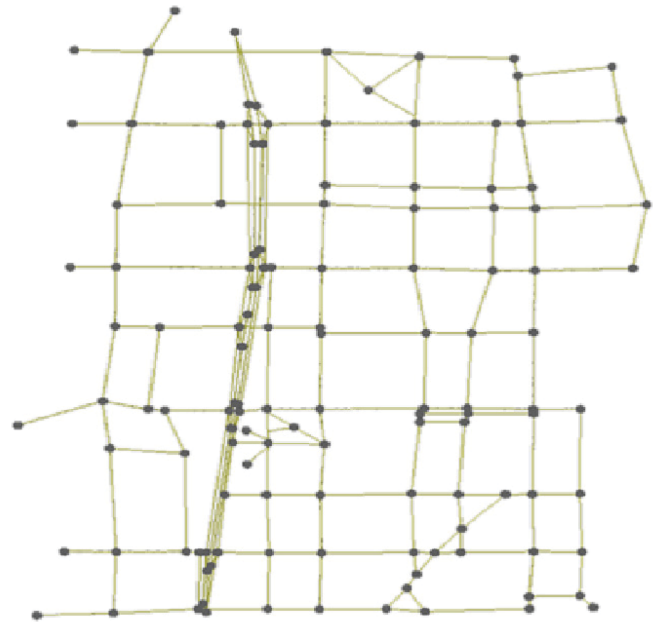


**FIGURE 3 Test bed network.**

Figure 4 provides a closer look at the real-time estimation module for a period that covers the simulated morning peak. The figure shows the inconsistency measure $M$ values at 10-min intervals. A rollback period of 60 min is considered. In this experiment, the measure $M$ is calculated every 10 min by using observations that extend over the past 60 min (i.e., six observations). The threshold value $\delta$ is set at 0.20, and the number of discretization intervals is equal to 10. No demand prediction is conducted in this experiment. As illustrated in the figure, time instances at which $M \geq 0.20$ are colored in light red, indicating that demand correction is required and rollback is executed. Time instances in which the rollback is taking place are colored in yellow. The execution time for each rollback period is recorded, which consists of (*a*) the time to solve the O-D demand updating problem and (*b*) the time to rerun the simulation model for the specified rollback period. For example, a consistency measure value of 0.25 (>$\delta$) is recorded at 7:40 a.m. The demand updating is activated and executed in 3.25 s. Given the corrected demand, the simulation is reset to the start of the rollback period. Then, it runs faster than real time until it synchronizes again with the real-world clock. The execution time for the simulation model is recorded to be 63.27 s. Thus, the total time in which the real-time estimation is not available is 66.52 s. A similar pattern is recorded for all executed rollbacks. The unavailability time is slightly greater than 1 min, which is less than the system's prediction cycle (i.e., 5 min). The next set of experiments examines the effect of the length of the rollback period on the demand correction accuracy. As described earlier, the measured traffic count on a link in a certain observation interval depends on the number of trips that started in earlier departure time

| | Time | Inconsistency Measure |
|---|---|---|
| | 7:20 a.m. | 0.08 |
| | 7:30 a.m. | 0.16 |
| | 7:40 a.m. | 0.25 |
| 60-min Rollback DU = 3.25 s RB = 63.27 s Total = 66.52 s | 6:40 a.m. | 0.01 |
| | 6:50 a.m. | 0.01 |
| | 7:00 a.m. | 0.02 |
| | 7:10 a.m. | 0.02 |
| | 7:20 a.m. | 0.01 |
| | 7:30 a.m. | 0.01 |
| | 7:40 a.m. | 0 |
| | 7:50 a.m. | 0.05 |
| | 8:00 a.m. | 0.15 |
| | 8:10 a.m. | 0.24 |
| 60-min Rollback DU = 3.55 s RB = 77.31 s Total = 80.86 s | 7:10 a.m. | 0.02 |
| | 7:20 a.m. | 0 |
| | 7:30 a.m. | 0.01 |
| | 7:40 a.m. | 0.02 |
| | 7:50 a.m. | 0.04 |
| | 8:00 a.m. | 0.07 |
| | 8:10 a.m. | 0.09 |
| | 8:20 a.m. | 0.11 |
| | 8:30 a.m. | 0.16 |
| | 8:40 a.m. | 0.24 |
| 60-min Rollback DU = 3.18 s RB = 61.09 s Total = 64.27 s | 7:40 a.m. | 0.02 |
| | 7:50 a.m. | 0.04 |
| | 8:00 a.m. | 0.07 |
| | 8:10 a.m. | 0.08 |
| | 8:20 a.m. | 0.05 |
| | 8:30 a.m. | 0.03 |
| | 8:40 a.m. | 0.02 |
| | 8:50 a.m. | 0.04 |
| | 9:00 a.m. | 0.1 |
| | 9:10 a.m. | 0.19 |
| | 9:20 a.m. | 0.28 |
| 60-min Rollback DU = 3.22 s RB = 74.19 s Total = 77.37 s | 8:20 a.m. | 0.05 |
| | 8:30 a.m. | 0.05 |
| | 8:40 a.m. | 0.06 |
| | 8:50 a.m. | 0.06 |
| | 9:00 a.m. | 0.06 |
| | 9:10 a.m. | 0.07 |
| | 9:20 a.m. | 0.08 |
| | 9:30 a.m. | 0.1 |
| | 9:40 a.m. | 0.16 |
| | 9:50 a.m. | 0.23 |
| 60-min Rollback DU = 3.09 s RB = 62.21 s Total = 65.3 s | 8:50 a.m. | 0.06 |
| | 9:00 a.m. | 0.06 |
| | 9:10 a.m. | 0.06 |
| | 9:20 a.m. | 0.06 |
| | 9:30 a.m. | 0.04 |
| | 9:40 a.m. | 0.01 |
| | 9:50 a.m. | 0 |
| | 10:00 a.m. | 0.01 |
| | 10:10 a.m. | 0.09 |
| | 10:20 a.m. | 0.18 |
| | 10:30 a.m. | 0.27 |
| 60-min Rollback DU = 3.27 s RB = 59.88 s Total = 63.15 s | 9:30 a.m. | 0.04 |
| | 9:40 a.m. | 0.03 |
| | 9:50 a.m. | 0.03 |
| | 10:00 a.m. | 0.04 |
| | 10:10 a.m. | 0.05 |
| | 10:20 a.m. | 0.07 |
| | 10:30 a.m. | 0.09 |
| | 10:40 a.m. | 0.11 |
| | 10:50 a.m. | 0.18 |

FIGURE 4 Illustration of online network state estimation with rollbacks (DU = demand updating; RB = rollback).

intervals and that use this link as part of their routes. If these intervals are not within the rollback period, the demand cannot be corrected, and hence the detected state of inconsistency cannot be fixed.

Figure 5 provides the results for this set of experiments. The figure compares the real-world observations for one freeway detector for 1 full day with the corresponding estimated values. The estimated values are obtained from the real-time traffic network estimation module with and without activating the demand consistency checking and updating module. Four values for the rollback period are considered: 10, 20, 30, and 60 min. The real-world observations are obtained with an offline simulation run. The demand for this run is obtained by uniformly doubling the time-dependent demand values used to run the real-time traffic network state estimation module. As such, without activating the consistency checking module, the estimated counts are approximately half the value of the measured ones. The demand updating algorithm corrects the demand to account for this difference. As shown in the figure, as the rollback period increases, the consistency checking module is able to reduce the difference between the measured and the estimated values. Considering a long rollback period is expected to include departure time intervals that contribute to the measured link counts. Correcting the O-D demand for these intervals provides better matching between the measured and estimated link counts as shown in Figure 5*d.* On the contrary, as shown in Figure 5*a,* the demand updating algorithm fails to correct the demand when a short rollback period is used as the relevant departure time intervals are not included in the horizon. Clearly, one should expect this rollback period to depend on the size of the network, the trip length distribution, and the level of congestion. As a rule of thumb, the travel time of the longest path in the network can be used as the length of the rollback period.

The above experiment is extended to examine the performance of the demand updating algorithm in correcting an initial demand of a fluctuating pattern as illustrated in Figure 6 (the green-colored line series). The results of using a 60-min rollback period with a threshold of 0.20 are illustrated for two different detectors along the freeway's northbound and southbound directions. As illustrated in the figure, the algorithm corrects the demand pattern to produce vehicle counts that are consistent with the corresponding real-world observations. The results in Figures 5 and 6 indicate that the demand updating algorithm is effective in correcting demand patterns that could be uniformly greater or less than the real-world demand pattern as well as those with irregular fluctuating patterns.

As described earlier, the demand correcting algorithm adopts the separable programming approach, which provides a linear approximation of the problem through discretizing the time-dependent O-D demand variables. The range for each demand variable is divided into small intervals. A key factor that is expected to affect the quality of the solution is the resolution of this discretization. A smaller discretization interval provides better approximation to the nonlinear objective function, which leads to a better solution to the original

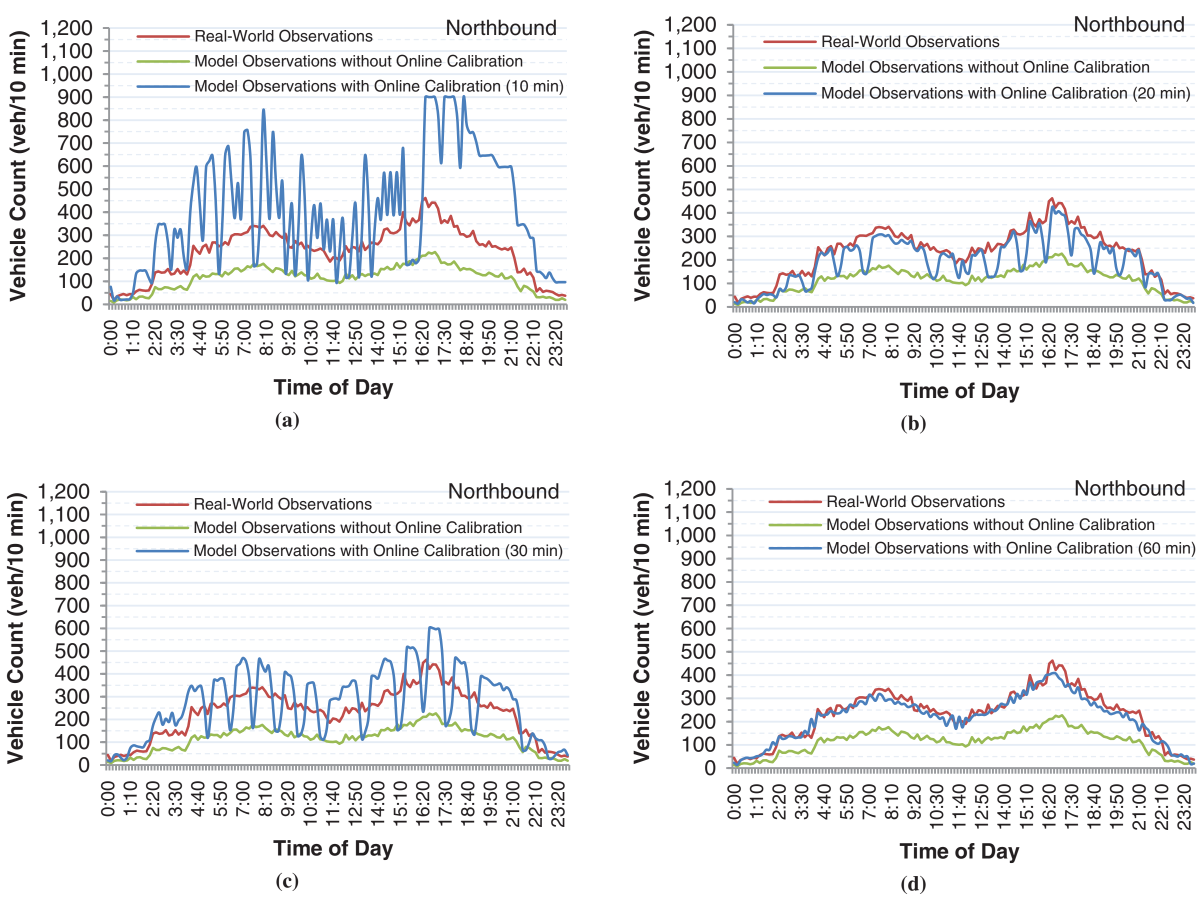



FIGURE 5 Effect of rollback period on traffic network state estimation accuracy for rollback periods of (*a*) 10 min, (*b*) 20 min, (*c*) 30 min, and (*d*) 60 min.

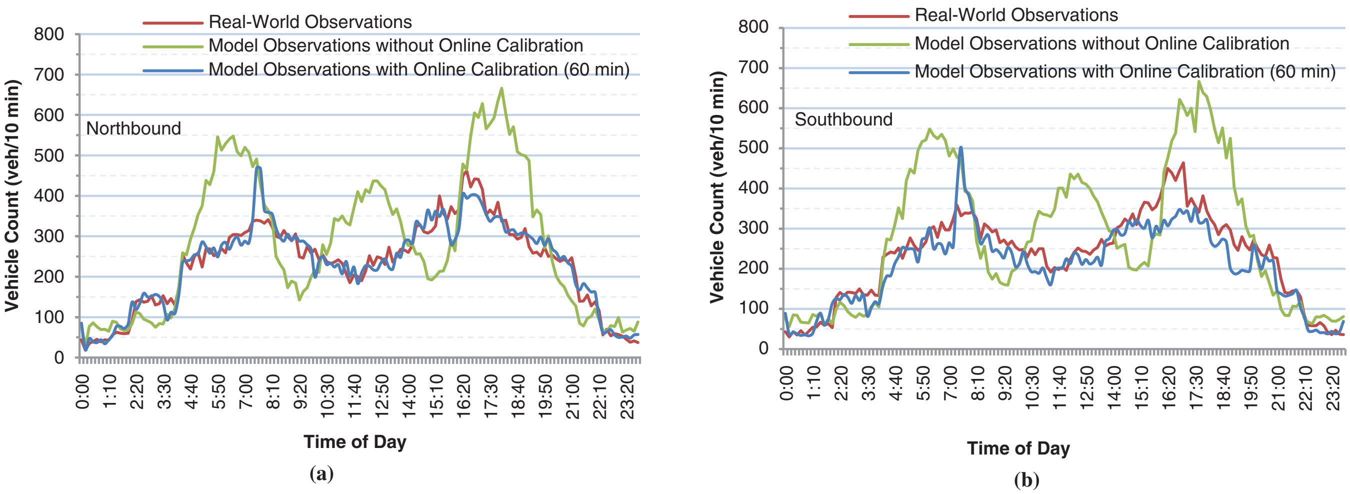



FIGURE 6 Updating fluctuating demand pattern: (*a*) northbound freeway detector and (*b*) southbound freeway detector.

program. The solution quality associated with three different values for the discretization interval is examined. The feasible range of the demand variables is discretized into three, five, and 10 equal intervals. The feasible range is taken as twice the demand value to be updated. In this experiment, a rollback period of 60 min and an inconsistency threshold of 0.2 are considered. Figure 7 illustrates the real-world observations and the corresponding estimated ones for the three discretization values for two different freeway detectors, one northbound and one southbound. As illustrated, increasing the number of discretization intervals (i.e., reducing the length) improves the model consistency by better matching the measured and estimated vehicle counts. The use of 10 discretization intervals provides better matching to the measured vehicle counts than the use of five and three discretization intervals. The use of a small discretization interval required a lower number of rollback calls of the simulation model for a horizon of one full day. For the used threshold value, when 10 discretization intervals were used, 50 rollback calls were executed during the simulated day to maintain the consistency of the model. Whereas for the cases in which five and three discretization intervals were used, 72 and 80 rollback calls were executed. Despite this increase in the number of rollback calls, consistency between the measured and estimated counts was not achieved, as illustrated in Figure 7.

The last experiment examines the effect of activating a demand projection module after each demand updating step. As the past demand is corrected, the demand for a prespecified future horizon is projected and used instead of the initial demand used by the model. The projection is based on a transition function that describes the demand time–varying pattern along the horizon of interest. Historical demand data could be used to derive this function, which is assumed to be given in this experiment. Thus, if the demand for interval $\tau$ is estimated, the demand for the next interval can be obtained as follows: $d_{ij}^{\tau+1} = f(d_{ij}^{\tau})$, where $f$ is the transition function from interval $\tau$ to interval $\tau + 1$. Figure 8 provides the results of this experiment, which compares the real-world observations for one freeway detector for 1 full day with the corresponding estimated values. The results are illustrated for the two cases in which the demand projection module is activated and deactivated, respectively. In both

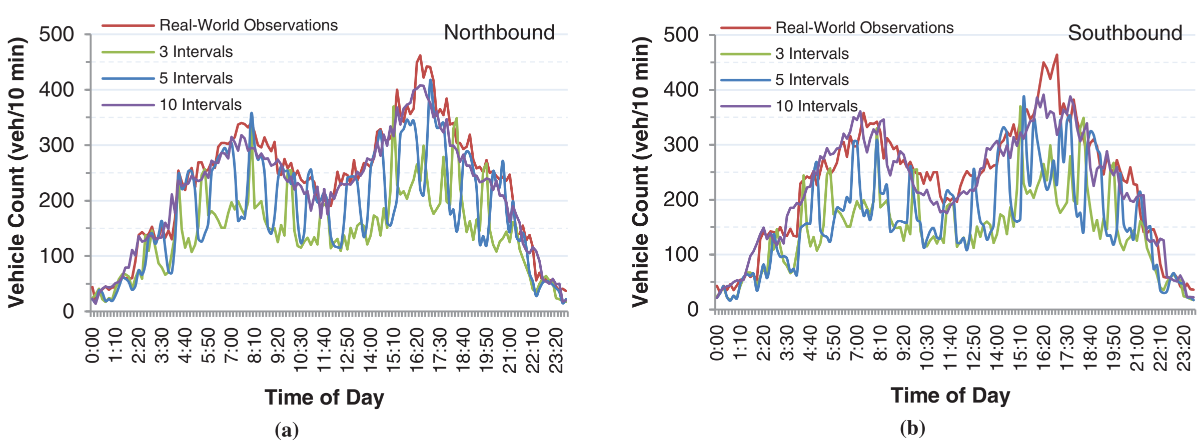
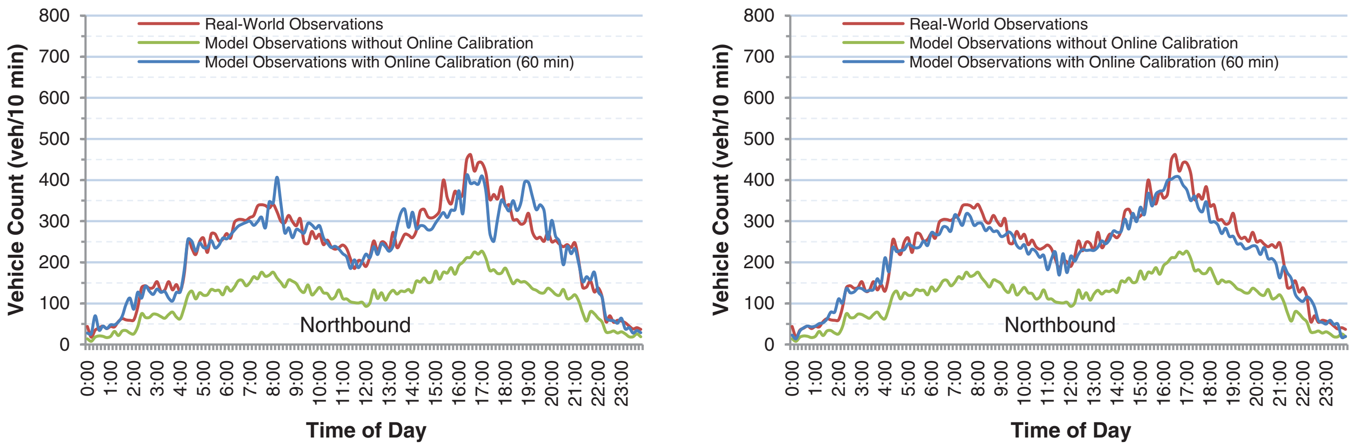



FIGURE 7 Effect of discretization resolution on quality of O-D demand correction: (*a*) northbound freeway detector and (*b*) southbound freeway detector.




(a) (b)

FIGURE 8 Effect of demand projection on number of demand updating calls and consistency results: (*a*) with demand projection, number of demand correction calls = 20, and (*b*) without demand projection, number of demand correction calls = 50.

cases, the results were obtained by using a 60-min rollback period with a threshold of 0.20. As illustrated in the figure, for both cases, the demand was corrected to ensure that the estimated and measured counts were consistent. However, activating the demand projection module to correct the future demand significantly reduced the number of required calls for the demand correction algorithm. When the demand projection module was activated, 20 calls were recorded for the 24-h horizon; whereas, 50 calls were recorded when the demand projection module was deactivated.

## CONCLUSIONS

This paper describes a real-time traffic network estimation model with online O-D demand consistency checking and updating. A rollback approach is adopted in which the time-dependent O-D demand is corrected for a certain past period to account for a detected state of model inconsistency. The model is allowed to roll back in time and quickly resimulates this past period until it synchronizes again with the real-world clock. The demand updating problem is formulated as a least squares minimization problem that is solved by using the separable programming methodology. The experiments conducted show that the approach used was successful in maintaining the consistency of the model with the measured observations. Experiments to examine the effect of the length of the rollback period and discretization interval are presented.

Several extensions are considered for this research work. For example, in this research, only demand correction was considered. As described above, several sources of error could contribute to a state of model inconsistency. Developing a comprehensive consistency checking module that recognizes the different sources of error and activates the appropriate correction procedure is considered as an extension of this research work. In addition, comparing the solution quality of this approach with reactive-based methodologies is another important extension of this work. Also, effort is under way to determine the suitability of this approach for larger networks in regard to the quality of the solution and satisfying the real-time execution requirement. Finally, examination of the effect of improving the estimation results on the prediction accuracy is under way.

## ACKNOWLEDGMENTS

The research presented in this paper was supported in part by the Dallas Area Rapid Transit and the Bobby B. Lyle School of Engineering at Southern Methodist University. The authors thank Koorosh Olyai at DART and Ed Seymour, Chris Poe, Roberto Macias, and Minh Lee at Texas A&M Transportation Institute for their valuable comments.

## REFERENCES

1. Mahmassani, H. S., Y. Hawas, K. Abdelghany, A. Abdelfatah, Y.-C. Chiu, and Y. Kang. *DYNASMART-X, Volume II: Analytical and Algorithmic Aspects.* Technical Report ST067-85-Volume II. Center for Transportation Research, University of Texas at Austin, 1998.
2. Mahmassani, H. S. Dynamic Network Traffic Assignment and Simulation Methodology for Advanced System Management Applications. *Networks and Spatial Economics,* Vol. 12, 2001, pp. 267–292.
3. Ben-Akiva, M., M. Bierlaire, D. Burton, H. N. Koutsopoulos, and R. G. Mishalani. Network State Estimation and Prediction for Real-Time Traffic Management. *Networks and Spatial Economics,* Vol. 1, 2001, pp. 293–318.
4. Peeta, S., and A. Ziliaskopoulos. Foundations of Dynamic Traffic Assignment: The Past, the Present and the Future. *Networks and Spatial Economics,* Vol. 1, No. 3/4, 2001, pp. 233–266.
5. Doan, D., A. K. Ziliaskopoulos, and H. S. Mahmassani. Online 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.
6. Kang, Y. *Estimation and Prediction of Dynamic Origin–Destination (O-D) Demand and System Consistency Control for Real-Time Dynamic Traffic Assignment Operation.* PhD dissertation. University of Texas, Austin, 1999.
7. Zhou, X., and H. S. Mahmassani. Dynamic Programming Approach for Online Freeway Flow Propagation Adjustment. In *Transportation Research Record: Journal of the Transportation Research Board, No. 1802,* Transportation Research Board of the National Academies, Washington, D.C., 2002, pp. 263–270.
8. Peeta, S., and S. Bulusu. A Generalized Singular Value Decomposition Approach for Consistent Online Dynamic Traffic Assignment. In *Transportation Research Record: Journal of the Transportation Research Board, No. 1667,* TRB, National Research Council, Washington, D.C., 1999, pp. 77–87.
9. Zhou, X., and H. S. Mahmassani. Online Consistency Checking and Origin–Destination Demand Updating: Recursive Approaches with Real-

Time Dynamic Traffic Assignment Operator. In *Transportation Research Record: Journal of the Transportation Research Board, No. 1923,* Transportation Research Board of the National Academies, Washington, D.C., 2005, pp. 218–226.
10. Ashok, K., and M. Ben-Akiva. Alternative Approaches for Real-Time Estimation and Prediction of Time-Dependent Origin–Destination Flows. *Transportation Science,* Vol. 34, No. 1, 2000, pp. 21–36.
11. Sherali, H., and T. Park. Estimation of Dynamic Origin–Destination Trip Tables for a General Network. *Transportation Research Part B,* Vol. 35, 2001, pp. 217–235.
12. Nie, Y., and M. Zhang. A Variational Inequality Formulation for Inferring Dynamic Origin–Destination Travel Demands. *Transportation Research Part B,* Vol. 42, No. 8, 2008, pp. 635–662.
13. Etemadnia, H., and K. Abdelghany. Distributed Approach for Estimation of Dynamic Origin–Destination Demand. In *Transportation Research Record: Journal of the Transportation Research Board, No. 2105,* Transportation Research Board of the National Academies, Washington, D.C., 2009, pp. 127–134.
14. Bierlaire, M., and F. Crittin. An Efficient Algorithm for Real-Time Estimation and Prediction of Dynamic OD Tables. *Operations Research,* Vol. 52, No. 1, 2004, pp. 116–127.
15. Hassan, A., K. Abdelghany, and A. Abdelghany. Dynamic Origin–Destination Demand Estimation Using Separable Programming. *Annals of Operations Research.* Under review.
16. Cascetta, E., D. Inaudi, and G. Marquis. Dynamic Estimator of Origin–Destination Matrices Using Traffic Counts. *Transportation Science,* Vol. 27, 1993, pp. 363–373.
17. Nie, Y., and H. M. Zhang. A Relaxation Approach for Estimating Origin–Destination Trip Tables. *Networks and Spatial Economics,* Vol. 10, No. 1, 2010, pp. 147–172.
18. Sherali, H., N. Arora, and A. Hobeika. Parameter Optimization Methods for Estimating Dynamic Origin–Destination Trip Tables. *Transportation Research Part B,* Vol. 31, 1997, pp. 141–157.
19. Tavana, H. *Internally-Consistent Estimation of Dynamic Network Origin–Destination Flows from Intelligent Transportation Systems Data Using Bi-Level Optimization.* PhD dissertation. University of Texas, Austin, 2001.
20. Ding, Y., P. B. Mirchandani, and S. A. Nobe. Prediction of Network Loads Based on Origin–Destination Synthesis from Observed Link Volumes. In *Transportation Research Record 1607,* TRB, National Research Council, Washington, D.C., 1997, pp. 95–104.
21. Dixon, M. P., and L. R. Rilett. Real-Time Origin–Destination Estimation Using Automatic Vehicle Identification Data. Presented at 79th Annual Meeting of the Transportation Research Board, Washington, D.C., 2000.
22. Zhou, X., and H. Mahmassani. A Structural State Space Model for Real-Time Traffic Origin–Destination Demand Estimation and Prediction in a Day-to-Day Learning Framework. *Transportation Research Part B,* Vol. 4, No. 8, 2007, pp. 823–840.
23. Bazaraa, M., H. Sherali, and C. Shetty. *Nonlinear Programming: Theory and Algorithms,* 3rd ed. Wiley, 2006.

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

*The Transportation Network Modeling Committee peer-reviewed this paper.*