

On-line calibration of behavior parameters for behavior-consistent route guidance

Alexander Paz, Srinivas Peeta *

School of Civil Engineering, Purdue University, 550 Stadium Mall Drive, West Lafayette, IN 47907-2051, USA.

Abstract

This paper calibrates on-line the parameters of a controller-estimated driver behavior model used in a deployable behavior-consistent approach for real-time route guidance by checking the consistency between the time-dependent actual and estimated system states. The behavior model has a fuzzy multinomial logit structure where the systematic utility component is obtained using aggregate behavioral *if-then* rules. The weights of these rules are calibrated through a fuzzy on-line calibration model using the unfolding traffic volume measurements. The on-line calibration is done within the deployment framework of the behavior-consistent approach where the drivers' likely response is factored in determining the route guidance strategies. The generalized structure of the calibration component enables it to simultaneously incorporate other sources of state inconsistency such as traffic flow model parameters. The results indicate that the calibration model can enhance the accuracy of system state estimation, leading to the increased effectiveness of the behavior-consistent route guidance. It provides the ability to more accurately predict drivers' likely route choices by using aggregate *if-then* rules, and consequently, aggregate level data. This is attractive in a deployment context as it implies reduced data needs at a disaggregate level, a difficult proposition in the real world.

© 2008 Elsevier Ltd. All rights reserved.

Keywords: on-line calibration; real-time route guidance; behavior parameters; behavior consistency; fuzzy control; deployment.

*Corresponding author. Tel.: +1 765 494 2209; fax: +1 765 496 7996.
E-mail address: peeta@purdue.edu. (S. Peeta).

1. Introduction

The development of deployable strategies for real-time information-based network control to enhance system performance requires simultaneously addressing significant methodological problems related to traffic flow dynamics, driver behavior, demand modeling, and information processing, typically within a dynamic traffic assignment (DTA) framework. This is because a realistic, effective, and anticipatory solution to the problem requires the estimation of traffic network states resulting from complex interdependencies among network flow spatio-temporal interactions, driver response behavior, supply characteristics (such as the information provided), and related sources of randomness.

A significant body of literature exists for each of the aforementioned methodological problems. DTA models have predominantly focused on robustly capturing the traffic flow dynamics while seeking to achieve some system-wide objectives. However, they mostly tend to pre-specify driver behavior. For example, they categorize driver behavior and/or assume driver classes with restrictive pre-specified objectives such as user equilibrium (UE) or system optimal (SO). However, such simplicity is not consistent with the real-world and is consequently problematic. The traffic network states unfolding over time are fundamentally dependent on driver behavior which is a key source of complexity due to the spatio-temporal interactions that result from driver route choice decisions. Hence, the incorrect modeling of driver behavior can negatively impact the prediction of the traffic network states and the effectiveness of information-based control strategies. Peeta and Yu (2004, 2006) highlight realism issues arising from the rigid representation of driver behavior under information provision, and the consequent barriers to developing effective operational paradigms for information-based traffic network management.

In contrast to the emphasis on traffic flow modeling in DTA models, traditional route choice models focus primarily on the socio-economic characteristics of drivers and the physical attributes of the routes. In addition, route choice models under information provision also consider some information-related attributes. However, they typically do not consider the spatio-temporal interactions resulting from the individual driver route choice decisions. To address this aspect, route switching decisions are typically modeled through en-route driver behavior models. Mahmassani and Jayakrishnan (1991) seek to capture the network-level traffic flow interactions by simulating driver en-route switching decisions assuming boundedly-rational driver behavior based only on route travel times. Abdel-Aty (1998) develops a nested logit model to predict en-route routing decisions for incident-related congestion under real-time information provision. Srinivasan and Mahmassani (2000) propose a multinomial probit framework specifying compliance and inertia as two factors that influence driver route choices under real-time information. While en-route driver behavior models consider spatio-temporal interactions and driver behavior to predict traffic network states, they are descriptive and do not address the controller objectives of enhancing system performance. Even when controller objectives are addressed in conjunction with a behavior model, the control mechanism does not engage them interactively and is based on a sequential logic. That is, there is no mechanism to ensure that controller-recommended routes are consistent with drivers' likely route choice decisions.

To illustrate the interdependencies between the network states and the route recommendations, Bottom (2000) develops a conceptual framework for the consistent route guidance problem. The framework explicitly recognizes the importance of estimating driver reaction to the information provided. The solution methods involve solving a fixed point problem

formed by three alternative composite maps: (i) route assignment fractions to network states, (ii) network states to guidance messages, and (iii) guidance messages to route assignment fractions. Although the framework illustrates many of the critical algorithmic and computational aspects of the problem, it still models driver behavior using a traditional DTA approach. In addition, the proposed solution methods are computationally intensive, precluding real-time deployment.

To ensure consistency between controller objectives and the driver decision-making process, Paz and Peeta (2007, 2008a) develop a behavior-consistent traffic routing approach where the controller factors the drivers' likely reactions to the information strategies while determining them. The behavior-consistent approach uses a controller-estimated driver behavior model to predict the proportion of drivers taking routes under the potential information strategies that the controller is iteratively seeking to determine. This implies solving a fixed-point problem where the controller-estimated driver behavior depends on the information strategies and vice versa. Hence, the behavior-consistent approach enables the simultaneous consideration of the controller objectives and driver behavior. Paz and Peeta (2008b) illustrate trade-offs between the controller objectives and the driver acceptability of the controller-recommended routes. They suggest that higher compliance rates by themselves do not necessarily translate to better performance, and that the route quality relative to the controller objectives is as important. Therefore, due to the aforementioned fixed-point relationship, the prediction accuracy of the controller-estimated driver behavior model is a key aspect of the behavior-consistent approach. This is because an incorrect prediction of the drivers' likely reactions to the information strategies can result in the generation of erroneous information strategies, negatively impacting network performance. From a deployment standpoint, this implies the need to calibrate the controller-estimated driver behavior model and represents the motivation for this research.

Procedures for the on-line calibration of traffic estimation/prediction systems vis-à-vis route guidance typically seek to correct for systematic inconsistencies so as to minimize the gap between the predicted and actual (observed) network states unfolding in real-time. While there are several potential sources of inconsistency, the characterization and solution methods for the problem have evolved from simple reactive approaches that adjust network-level factors or control strategies to consistency-seeking models (Peeta and Yu, 2006) that assign primacy to the behavioral aspects. Peeta and Bulusu (1999) propose a generalized singular value decomposition based method that adjusts the number of drivers on each route to minimize the error between the prediction and observed system states. They view the inconsistencies as arising from the incorrect prediction of the unequipped driver routes, time-dependent O-D demand, incident characteristics, and route compliance aspects of equipped users. However, these sources are not separated in the method, and aggregate link counts are the basis for the route proportion adjustments. Thereby, systematic inconsistencies in the associated models or their parameters are not addressed.

Initial efforts to calibrate DTA model parameters have concentrated mostly on the supply or demand aspects whereby the traffic flow modeling parameters or the origin-destination (O-D) demand are adjusted on-line based on unfolding system states. Mahmassani et al., (1998) calibrate a DTA model using a proportional-integral-derivative feedback control strategy that reacts to any observed on-line deviations in traffic conditions. The procedure uses a real-time module to adjust the parameters of the travel time function and the flow propagation equations using real-time data on traffic measures such as average speed, inflow, and outflow. Further, an off-line module is used to update the parameters using full information on past conditions so as to improve the real-time adjustments.

Recent efforts seek to develop generalized frameworks to simultaneously calibrate supply and demand parameters to capture the critical interactions between these elements. Balakrishna et al., (2007) and Antoniou (2004) propose state-space frameworks to address the off-line and on-line calibration of DTA models, respectively. In the experimental analysis, they consider parameters associated with speed-density relationships, segment capacities, and the time-dependent O-D demand.

Driver behavior is a fundamental determinant of the network state evolution. Hence, a natural next step towards the accurate prediction of traffic network states is the incorporation of behavioral aspects in the consistency-checking procedures. In this context, Peeta and Yu (2006) propose a behavior-based consistency-seeking approach that considers traffic flow dynamics in conjunction with a hybrid probabilistic-possibilistic driver behavior model (Peeta and Yu, 2005) to consistently address driver learning processes so as to predict the time-dependent network states. The associated consistency-seeking problem updates the driver class fractions in the ambient traffic stream based on link traffic counts to reduce the state consistency gap. However, it does not adjust the underlying driver behavior models or their parameters.

In this study, the on-line calibration problem seeks to update the behavior model parameters. It focuses on ensuring a meaningful prediction of driver behavior under information provision, and consequently, on evolving network states. A fuzzy on-line calibration model is proposed to address the problem where the controller seeks to minimize the difference between the actual and the estimated network states by updating the controller-estimated driver behavior model parameters using the link traffic counts unfolding over time. As discussed earlier, the calibration of the controller-estimated driver behavior model enhances system performance by enabling more effective information-based network control strategies.

The remainder of this paper is organized as follows. Section 2 summarizes the behavior-consistent approach and discusses the associated on-line calibration problem. Section 3 describes the controller-estimated driver behavior model. Section 4 presents the fuzzy on-line calibration model used to address the on-line calibration problem as part of the deployment framework for the behavior-consistent approach. Section 5 discusses experiments and analyzes their results. Section 6 provides some concluding comments.

2. Problem background and description

Section 2.1 defines some terms relevant to the behavior-consistent approach. Section 2.2 summarizes the solution framework for the behavior-consistent approach. It provides the background for the on-line calibration problem discussed in Section 2.3. The notion that the behavior-consistent routing problem and the associated on-line calibration problem are addressed in a single on-line deployment framework is conceptually illustrated in Fig. 1, where the shaded boxes correspond to the solution logic for the behavior-consistent approach alone (Paz and Peeta, 2007). The non-shaded boxes correspond to new components developed in this paper to enable the on-line calibration of the controller-estimated driver-behavior model.

2.1. Definition of terms

The behavior-consistent approach categorizes routes based on their relevance to the controller and the drivers. Three types of routes are defined as follows.

Controller-desired routes (*DK*): These are routes that the controller would like the drivers to

choose. Depending on the controller objective, they can, for example, be the time-dependent SO or UE DTA routes. They are obtained by solving a deterministic DTA problem using current network conditions and projected demand for the appropriate time duration (represented by the stage length in this paper).

Driver-preferred routes (*PK*): These routes correspond to the choice set of the drivers. They are preferred by the drivers and are likely to be accepted by them. The estimation of the driver-preferred route set is a key step for any route choice model. From a technological standpoint, these route sets could possibly be obtained by the controller through two-way communication with drivers equipped with personalized information/communication devices. More generally, they are estimated (Bekhor et al., 2006) based on historical data collected through travel surveys and/or technologies such as two-way communication systems and global position systems.

Controllable routes (*CK*): These routes belong to both the controller-desired and driver-preferred route sets. In the behavior-consistent approach discussed next, they represent the set of routes used by the controller to recommend routes to drivers so as to influence system performance.

2.2. Solution framework for the behavior-consistent approach

The solution framework for the behavior-consistent approach enables the real-time deployment of behavior-consistent information strategies. As stated earlier, it is represented by the shaded boxes in Fig. 1. Paz and Peeta (2007) provide a comprehensive description of this framework. Here, a relevant summary is provided as background for the associated on-line calibration problem.

The solution framework includes as components an iterative search procedure and a deterministic DTA model within a rolling horizon stage-based framework. The planning horizon of interest is divided into stages, and each stage consists of a roll period and a tail period as seen in Fig. 1. At some point towards the end of the current stage σ , the controller computes the behavior-consistent information strategies for the next stage. First, it uses the deterministic DTA model to project traffic conditions and determine the SO DTA solution for the next stage ($\sigma+1$) based on the field traffic conditions for the current roll period and the forecasts of the origin-destination (O-D) demand for the next stage. An iterative search based optimization procedure (shaded box with dashed borders in the middle of Fig. 1) is then used to determine the information strategies that minimize the difference between the SO proportions for controllable routes and the corresponding estimated proportion of drivers taking those routes in the next roll period (Paz and Peeta, 2008a). It consists of a fuzzy control model that determines the search direction and step size to update the information strategies, and a controller-estimated driver behavior model that predicts the likely driver route decisions in light of these updated information strategies. That is, the controller seeks to direct the traffic system as close as possible to the time-dependent SO system state in a behavior-consistent manner using SO routes that are also preferred by the drivers. Hence, at convergence, the iterative search procedure determines the behavior-consistent proportions of drivers that should be recommended to take specific routes in the next roll period so as to achieve close to SO proportions for those routes. At the end of the current roll period, the stage counter is incremented by one. In the next roll period, routes are recommended to a subset of drivers based on the behavior-consistent route proportions. The actual driver behavior and the network flow dynamics determine the field conditions for that roll period. If the end of this roll period does not represent the end of the

planning horizon, field network conditions are measured towards the end of it and the entire stage-based procedure is repeated. Otherwise, the rolling horizon framework is terminated.

In the above framework, the SO solution and the corresponding routes represent the ideal system states. However, without loss of generality, they can be replaced by other states based on objectives such as the UE solution. Elsewhere (Paz and Peeta, 2008b), the authors study the effects of directing the system towards different objectives. In addition, they develop different deployment paradigms to enable recommending driver-preferred routes that partially overlap with the controller-desired routes. The corresponding results illustrate the flexibility of the behavior-consistent approach in the deployment context. They show how the controller can enhance system performance using different objectives and sets of routes to provide information.

2.3. On-line calibration problem

Fig. 1 explicitly illustrates the on-line calibration problem in the context of the broader behavior-consistent traffic routing problem addressed by the controller. As stated earlier, the non-shaded boxes represent components to address the calibration problem. The controller-estimated driver behavior model is used to estimate the driver routing decisions in the current roll period in light of the controller-recommended routes. A traffic flow simulator is used to obtain the controller-estimated traffic network states for the current roll period based on the estimated driver decisions. In this study, the traffic flow simulator is assumed to be accurate, and the calibration is focused on the behavior parameters. Thereby, if gaps exist between the controller-estimated and actual network states in terms of link traffic counts, a fuzzy on-line calibration model is used to calibrate the controller-estimated driver behavior model parameters. If state consistency issues do not exist, the field conditions for the current roll period are used to repeat the behavior-consistent approach by calculating the SO DTA solution for the next roll period.

While this study addresses state inconsistency by adjusting behavior parameters alone, the proposed fuzzy on-line calibration methodology provides a generalized approach to handle multiple sources of inconsistency through the use of aggregate *if-then* rules. Thereby, for example, rules associated with traffic flow modeling inconsistency can be seamlessly incorporated along with rules for behavior model inconsistency without any change to the structure of the fuzzy methodology. Another advantage of the methodology is that aggregate level sensor data can be used for addressing the calibration problem. This circumvents the need for disaggregate data (such as individual driver level data), enhancing the ability to practically deploy the methodology.

3. Controller-estimated traffic network states

Section 3.1 describes the controller-estimated driver behavior model used in this study. The output from the model is the set of time-dependent driver routing decisions. Section 3.2 illustrates how these drivers are loaded onto the network in the traffic flow simulator to estimate the time-dependent network states for the calibration problem.

3.1. Controller-estimated driver behavior model

Over the past two decades, a body of literature has been developed for the on-line estimation

and prediction of driver route choice behavior under information provision. Existing models range from econometric (probabilistic) to hybrid (probabilistic-possibilistic) models (Peeta and Yu, 2005). To handle the uncertainty associated with driver behavior, econometric models assume well-defined probability distributions while possibilistic models used fuzzy frameworks that can handle linguistic/qualitative and/or difficult-to-measure variables (Peeta and Yu, 2004). The current study proposes a fuzzy multinomial logit model as the controller-estimated driver behavior model. It uses simple aggregate-level behavioral *if-then* rules to determine the systematic component of the utilities of the various routes (alternatives). Akin to standard discrete choice logit models, an i.i.d extreme value error component is added to each utility to account for the randomness in driver behavior. Akin to models proposed by Lotan and Koutsopoulos (1993, 1999) for route choice behavior, the decision process is modeled as a non-linear combination of behavioral rules where each rule deals with a different aspect of the overall choice process. The controller-estimated driver behavior model is described hereafter.

3.1.1. Behavioral *if-then* rules

While the route choice process is time-dependent, its associated time dimension is ignored in Section 3.1.1 without loss of generality to simplify the notation. The driver routing decisions are based on a set of behavioral *if-then* rules that relate the decisions to the route characteristics (of the associated driver-preferred routes), the driver attributes in terms of information availability, and level of responsiveness to the information strategies. It is reasonable to expect that drivers do not use very sophisticated rules and/or many factors to make on-line routing decisions due to the associated time constraints. Hence, simple and straightforward rules consisting of one-dimensional left hand side (LHS) and right hand side (RHS) components are proposed here. In our experiments, it is assumed that travel time, route complexity, and the controller-recommended routes, are the key factors that influence the route choice decision-making process. However, additional factors can easily be added by creating the corresponding behavioral *if-then* rules.

The LHS (antecedent) of the rules deal with travel time, route complexity, and the controller-recommended routes. The RHS (consequent) deals with the propensity to choose a route, but does not represent the route choice itself. Rather, it is used to model the attractiveness of a driver-preferred route based on the conditions described by the LHS. In general, the rules used here are defined as:

“If A_k^h , Then B_k^h ”, $h = 1, \dots, BR$ and $k \in PK_{ij}$

where A_k^h is the LHS component of the h^{th} rule that corresponds to a characteristic of route k connecting O-D pair ij , B_k^h is the RHS component of the h^{th} rule that deals with the attractiveness of route k , and BR is the total number of rules. Here, k belongs to the set of driver-preferred routes (PK_{ij}).

Table 1 summarizes rules grouped using their LHS. Different sets of rules are used to model different levels of responsiveness to the information strategies. The LHS of the rules associated with the controller-estimated expected route travel times TT is characterized by the following five fuzzy sets: “Very Low (VL)”, “Low (L)”, “Medium (M)”, “High (H)”, and “Very High (VH)” travel times. The controller can estimate these expected travel times using historical data.

The number of nodes NV for each route is used to estimate the effect of route complexity on the route choice decisions. Here, the LHS is characterized by the following five fuzzy sets: “Very Low (VL)”, “Low (L)”, “Medium (M)”, “High (H)”, and “Very High (VH)” number of nodes. For controller-recommended routes, the LHS corresponding to a route recommendation Y is characterized by the following three fuzzy sets: “the Route is Recommended (RR)”, “the Route is Not Recommended (RNR)”, and the “Route Was Recommended (RWR)” in the previous roll period.

The RHS of the rules characterizes the attractiveness V of a route in terms of the following five fuzzy sets: “the driver will not choose this route (N)”, “the driver will probably not choose this route (PN)”, “the driver is indifferent to choosing this route (I)”, “the driver will probably choose this route (PO)”, and “the driver will choose this route (O)”.

The rules used in this study to capture driver behavior are based on the findings from previous studies and field observations. For example, routes with short travel time are preferred over those with higher travel times. Consistent with fuzzy logic, it is important to note that the inputs for the rules may not necessarily coincide with one of the LHS fuzzy sets described above. Rather, each input belongs to these fuzzy sets with different degrees of membership, and consequently will likely trigger the firing of more than one rule. The degree of membership is determined using membership functions.

3.1.2. Membership functions

Triangular membership functions are used to define the fuzzy sets associated with the behavioral *if-then* rules. The triangular shape is motivated by mathematical convenience. Only three parameters are required to deal with triangular membership functions. The controller’s expectation of driver route perception can be modeled through the shape, range, and amount of overlap between adjacent sets of the membership functions. For example, if the controller has poor knowledge (high ambiguity) or the expected route travel times vary widely among the drivers, wide membership functions can be used to represent that aspect. By contrast, narrow membership functions imply that the controller has good estimates or that the expected travel times are similar among the drivers. Although the shape of the membership function and its parameter values contribute to prediction accuracy, the behavioral *if-then* rules used and their associated weights in the fuzzy aggregation process are more critical for accurate route choice estimation. This is because the behavioral *if-then* rules define which membership functions are used, and their weights affect their contribution to the route attractiveness. Hence, this study uses simple membership functions and focuses on calibrating the weights of the *if-then* rules.

The membership functions are used to capture the expected degree of mapping μ between the controller’s expectation for an attribute and the LHS fuzzy sets. For controller-estimated expected travel times, the controller estimates that drivers have a range ($MinTT_k, MaxTT_k$) of possible travel times for each preferred route k . The degree of mapping for TT_k is represented by $\mu_T(TT_k)$. Five membership functions are defined to cover the range of the controller-estimated travel time. Given the range for each driver-preferred route, a super range covering all routes is defined ($Min TT = \min_{k \in PK_{ij}} MinTT_k, Max TT = \max_{k \in PK_{ij}} MaxTT_k$). This super range is covered evenly using five membership functions, as shown in Fig. 2. The same approach is used for number of nodes (route complexity) as well.

Fig. 2 also shows the three functions used to represent the membership functions for the LHS of the rules associated with route recommendations. There is no overlap among them because a

route is either recommended or not recommended. Hence, the membership functions associated with the RHS of these rules are directly used by the procedure; the degree of membership is either 1 or 0 based on whether that route is recommended or not recommended.

For the RHS of the behavioral rules, five membership functions corresponding to five fuzzy sets are used to characterize the route attractiveness. A range (-1, 1) is used to model the relative attractiveness of the routes, and the fuzzy logic decision process discussed hereafter uses only the relative difference in attractiveness over the set of driver-preferred routes to generate the controller-estimated driver route choice.

3.1.3. The fuzzy logic decision process

Fig. 3 summarizes the fuzzy logic decision process used to obtain the controller-estimated driver route choice. The inputs, TT_k , NN_k , and Y_k , are matched against the BR behavioral *if-then* rules to determine the activated (fired) rules and their corresponding fuzzy consequents V_{hk}^* . The membership functions μ of the consequents of the behavioral *if-then* rules are multiplied by their weights W . A fuzzy inference and aggregation mechanism is used to combine the consequences of all rules that are fired, and a defuzzification scheme is used to determine the controller-estimated attractiveness of each route. In this study, the max-min composition operator and Larsen product implication operator are used for fuzzy inference, and the center of gravity method (CGM) is used for defuzzification (Tsoukalas and Uhrig, 1997). The CGM is given by:

$$V_k = \frac{\sum_{h=1}^{BR} w_h \cdot V_{hk} \cdot S(\mu_{V_{hk}^*})}{\sum_{h=1}^{BR} w_h \cdot S(\mu_{V_{hk}^*})} \quad \forall k \in PK_{ij} \quad (1)$$

where $S(\cdot)$ determines the area of the fuzzy sets V_{hk}^* whose centroids are defined by V_{hk} , and V_k represents the attractiveness of route k . This process is repeated for all driver-preferred routes to generate the route attractiveness vector V . Since the controller-estimated driver behavior model needs to identify a discrete route for each driver, a mechanism is developed to select a route based on the vector V in which the attractiveness of an alternative V_k is treated as the systematic component of a random utility model. The utility of alternative k for driver r is given by:

$$U_k^r = V_k^r + \varepsilon_k^r \quad \forall r, k \in PK_{ij} \quad (2)$$

where ε_k^r is assumed to be an i.i.d. extreme value random component. Thereby, alternative k is chosen by driver r using the resultant fuzzy multinomial logit model with probability:

$$P^r(k) = P(V_k^r + \varepsilon_k^r \geq V_l^r + \varepsilon_l^r, \forall l \neq k) \quad \forall r, k \in PK_{ij} \quad (3)$$

The route choice probabilities are converted to discrete route choices using the following approach. First, a uniform random number generator is used to generate values between 0 and 1. Second, the probability range between 0 and 1 is demarcated into smaller ranges according to the

controller-estimated probabilities of a driver choosing driver-preferred routes. For example, if there are three routes with estimated choice probabilities 0.2, 0.3, and 0.5, the ranges associated with them are 0.0-0.2, 0.2-0.5, and 0.5-1.0, respectively. Third, if the random number generated falls in the range of choosing a specific route, the controller assumes that the driver chooses that route.

3.2. Network loading mechanism

Fig. 4 illustrates the network loading mechanism for roll period $\rho(\sigma)$ of stage σ using the controller-estimated driver behavior model to determine the initial and en-route controller-estimated driver route choices. The roll period is divided into discrete time intervals of length Δ , denoted by t . The initial routes for the new drivers in interval t are determined based on the dynamic inputs for t in terms of information provision and the current route characteristics. Driver en-route route choices in interval t are considered for those drivers who did not reach their destination in interval $t-1$ and who are located at an intermediate node (on their existing route) at the beginning of interval t . Intermediate nodes are viewed as potential decision nodes. These drivers are loaded onto the network at the beginning of t based on their en-route route choices using the dynamic inputs. If a driver who did not reach his/her destination at the end of the previous roll period $\rho(\sigma-1)$ is located on a link rather than at an intermediate node at the beginning of the first interval t of $\rho(\sigma)$, he/she is loaded onto the network at the beginning of t using his/her existing route.

A traffic flow simulator is used to generate the controller-estimated network state for interval t using the controller-estimated route choices. If t represents the last interval of $\rho(\sigma)$, the network loading for this roll period is terminated. Otherwise, the procedure is repeated until the end of $\rho(\sigma)$.

4. On-line parameter calibration

Section 4.1 discusses the formulation for the on-line calibration of the behavioral parameters. Section 4.2 describes the fuzzy on-line calibration model to calibrate the weights of the *if-then* rules in the controller-estimated driver behavior model.

4.1. Calibration of behavioral parameters

Several factors can contribute to the inconsistency between the controller-estimated network states and the actual conditions unfolding in real-time. Peeta and Bulusu (1999) list the following factors: (i) incorrect estimation of the time-dependent O-D demand, (ii) unexpected traffic incidents, (iii) incorrect traffic flow modeling, (iv) incorrect driver behavior modeling, (v) incorrect assumptions on system-related parameters, (vi) noise/sparsity in measurements, and (vii) failure of advance traveler information systems (ATIS) components.

The fuzzy on-line calibration model proposed in this paper can handle inconsistencies due to modeling errors (related to O-D demand, traffic flow, and behavior). As stated earlier, this study focuses on state inconsistency arising due to inaccurate values for the parameters of the controller-estimated driver behavior model. It assumes that the traffic flow modeling, the O-D demand predictions, and the data used here are accurate. This is done to derive insights on the controller-estimated driver behavior modeling aspects by isolating its effects.

The calibration problem seeks to update the weights of behavioral *if-then* rules of the controller-estimated driver behavior model so as to minimize the difference between the controller-estimated and actual (observed) network states. The associated formulation for roll period $\rho(\sigma)$ of stage σ is as follows.

$$\text{Minimize: } [\hat{\omega}^{\rho(\sigma)} - \bar{\omega}^{\rho(\sigma)}]^2 \quad (4)$$

Subject to:

$$\hat{\omega}^{\rho(\sigma)} = \hat{\omega}_*^{\rho(\sigma)} + \sum_t (MK \times C^t \times \hat{\delta}^t) \quad (5)$$

$$\hat{\delta}^t = \sum_{r \in R^t} g[\hat{F}(X^{rt}, Y^{rt}(\theta^{\rho(\sigma)}); W^{\rho(\sigma)})] \quad (6)$$

where,

- $\hat{\omega}^{\rho(\sigma)}$ is the estimated vector of link traffic counts for roll period $\rho(\sigma)$ of stage σ
- $\bar{\omega}^{\rho(\sigma)}$ is the observed vector of link traffic counts for roll period $\rho(\sigma)$ of stage σ
- $\hat{\omega}_*^{\rho(\sigma)}$ is the estimated vector of link traffic counts for roll period $\rho(\sigma)$ of stage σ for drivers who do not reach their destinations during roll period $\rho(\sigma-1)$ of stage $\sigma-1$
- MK is the estimated link-route incidence matrix for the driver-preferred routes
- C^t is the estimated link-route incidence adjustment matrix for time interval t
- $\hat{\delta}^t$ is the estimated vector of the number of new O-D desires for interval t taking driver-preferred routes
- \hat{F} is the controller-estimated driver behavior model which is used to estimate driver route choices
- R^t is the vector of O-D desires in time interval t
- X^{rt} is the estimated vector of route characteristics excluding information that influences the route choice decision of driver r in time interval t
- Y^{rt} is the route recommended by the controller to driver r in time interval t
- $\theta^{\rho(\sigma)}$ is the prescriptive information defined as the proportion of drivers that must be recommended to take specific routes in roll period $\rho(\sigma)$ of stage σ
- $W^{\rho(\sigma)}$ is the vector of rule weights (parameters) of the controller-estimated driver behavior model for roll period $\rho(\sigma)$ of stage σ

The controller objective (4) is to minimize the square of the difference between the estimated and observed vectors of link traffic counts for roll period $\rho(\sigma)$. The estimated vector of link traffic counts is determined using the network loading mechanism discussed in Section 3.2. It is expressed here by Equation (5) as the summation of the vectors of link counts for existing drivers who did not reach their destinations at the end of $\rho(\sigma-1)$ and the new O-D desires. $MK \times C^t \times \hat{\delta}^t$ represents the estimated vector of link count contributions from the new O-D desires entering the network in time interval t . The link-route incidence matrix MK is defined by the driver-preferred route sets. This matrix is used here only to generate the initial set of route alternatives for interval t . Unlike for DTA models, it does not define the entire driver route trajectory using a time-dependent link-path incidence matrix. Drivers make pre-trip route choices, and can change these choices en-route at decision nodes based on the ambient driving conditions and the information

provided to them. C^t denotes the adjustment to MK to ensure consistency between the observed and estimated link count contributions due to the new O-D desires entering the network in time interval t . Equation (6) defines the estimated vector of the number of new O-D desires for interval t taking driver-preferred routes in terms of the number of times that these routes are chosen by the O-D desires. Here, function g counts the number of times that a driver-preferred route is estimated to be chosen by new drivers. In the study experiments, X consists of the controller-estimated expected travel times and the number of nodes for the driver-preferred route sets.

The on-line calibration is done towards the end of $\rho(\sigma)$, at which time the observed link counts for all time intervals in this roll period are available. However, it needs to be done before the computation begins (Fig. 1) for the behavior-consistent strategies for the next roll-period. The unknown variables in the formulation (4)-(6) are the weights $W^{\rho(\sigma)}$ of the controller-estimated driver behavior model \hat{F} . Link traffic counts averaged across all time intervals in the roll period $\rho(\sigma)$ for stage σ (up to the point where behavior-consistent strategy computations begin for the next stage) serve as the network state data points to estimate the weights using the fuzzy calibration model described in the next section.

4.2. Fuzzy on-line calibration model

Fig. 5 illustrates the fuzzy on-line calibration model which consists of an input step (non-shaded box with dotted borders), a decision-processing step (non-shaded boxes with solid borders), and an output step (non-shaded box with dashed borders).

4.2.1. Input

The inputs are the vectors of error $e_{\overline{w}}^{\rho(\sigma)}$ and change in error $\Delta e_{\overline{w}}^{\rho(\sigma)}$ defined by:

$$e_{\overline{w}}^{\rho(\sigma)} = \overline{w}^{\rho(\sigma)} - \hat{w}^{\rho(\sigma)} \quad \text{and} \quad \Delta e_{\overline{w}}^{\rho(\sigma)} = e_{\overline{w}}^{\rho(\sigma)} - e_{\overline{w}}^{\rho(\sigma-1)} \quad (7)$$

where $\Delta e_{\overline{w}}^{\rho(\sigma)}$ is the difference between the current error $e_{\overline{w}}^{\rho(\sigma)}$ and the error in the previous stage $e_{\overline{w}}^{\rho(\sigma-1)}$.

4.2.2. Decision-processing component

Akin to the fuzzy logic decision process summarized in Section 3.1.3, the decision processing step uses calibration control rules, their associated membership functions, and a fuzzy aggregation, inference, and defuzzification scheme to determine the adjustment to the weights of the behavioral *if-then* rules.

4.2.2.1. Calibration control rules

The control *if-then* rules used by the fuzzy calibration model are two-dimensional rules (two inputs) obtained from observed patterns and problem characteristics. Three sets of control *if-then* rules are used to calibrate the behavioral *if-then* rules, one each for the behavioral rule consequent (RHS) implying an increase, decrease or neutrality related to route attractiveness.

Hence, for the same route, the weights of some behavioral rules may need to be increased while those of others may need to be decreased. For example, if the error associated with a route is positive, the number of drivers taking this route should be increased. This implies an increase in the weights associated with the behavioral rules that correspond to an increase in the attractiveness of this route. Hence, the calibration model determines how to calibrate the behavioral rule weights using a set of control rules such as:

If [e is NL and Δe is PL], Then [Δw is NS]

In this example, if the error e is negative large (NL) and the change in error Δe is positive large (PL), then the weight w is decreased by a negative small (NS) quantity Δw . Here, the consequent implies a decrease in the attractiveness of a route.

The LHS and RHS of the control rules are characterized by the following five fuzzy sets: “Negative Large (NL)”, “Negative Small (NS)”, “Zero (ZR)”, “Positive Small (PS)”, and “Positive Large (PL)” for error and change in error. Hence, e , Δe , and $\Delta w \in [\text{NL}, \text{NS}, \text{Z}, \text{PS}, \text{PL}]$. Table 2 shows the set of control *if-then* rules used by the fuzzy calibration model based on these five fuzzy sets. The total number of control rules is denoted as CR .

4.2.2.2. Membership functions

Corresponding to the five fuzzy sets, there are five triangular membership functions each for e , Δe , and Δw . Three membership functions, one for each of the two inputs and one for the output, are associated with each control *if-then* rule. The membership functions evenly cover the range of the domains for the inputs and output. The membership function parameters require off-line calibration to enable consistent on-line calibration of the parameters of the controller-estimated driver behavior model. In the study experiments, the off-line calibration of the membership function parameters was conducted using several iterations of the solution framework shown in Fig. 1.

4.2.2.3. Decision process

The max-min composition operator and Larsen product implication operator are used for fuzzy inference to determine the membership function μ of the RHS of control rule l of the weight $w_h^{\rho(\sigma)l*}$ for all behavioral rules h for roll period $\rho(\sigma)$ of stage σ . The center of gravity method is then used for defuzzification to determine the adjustments to the weights:

$$\Delta w_h^{\rho(\sigma)} = \frac{\sum_{l=1}^{CR} \bar{w}_h \cdot S(\mu_{w_h^{\rho(\sigma)l*}})}{\sum_{l=1}^{CR} S(\mu_{w_h^{\rho(\sigma)l*})}} \quad \forall h = 1, \dots, BR \quad (8)$$

where $S(\cdot)$ determines the area of the fuzzy sets $w_h^{\rho(\sigma)l*}$ whose centroids are defined by \bar{w}_h . $\Delta w_h^{\rho(\sigma)}$ represents the adjustment to the weight of behavioral rule h . The process needs to be repeated for all behavioral rules and data points (link traffic counts) resulting in a vector of

weight adjustments $\Delta w^{\rho(\sigma)}$.

4.2.3. Output

The calibrated weights used to determine the information strategies for the roll period of the next stage ($\sigma+1$) are defined as:

$$w^{\rho(\sigma+1)} = w^{\rho(\sigma)} + \Delta w^{\rho(\sigma)} \quad (9)$$

5. Experiments

Simulation experiments are conducted using the framework of Fig. 1 to evaluate the performance of the fuzzy on-line calibration model in the context of the broader problem of the on-line deployment of the behavior-consistent information-based network control strategies.

5.1. Experimental setup

5.1.1. Network characteristics

Simulation experiments are conducted using the Borman expressway corridor network shown in Fig. 6. It is located in northwest Indiana and consists of a sixteen-mile section of the Borman expressway (I-80/94), I-90 toll freeway, I-65, and the surrounding arterials and streets. It has 197 nodes, 460 links, and 43 zones (with centroids that represent origins/destinations). Depending on the destination, different numbers of alternative routes exist to divert traffic. Fig. 6 also shows 4 driver-preferred routes and 4 controller-desired routes associated with an O-D pair. The calibration of the weights of the 4 preferred routes and the behavioral *if-then* rule 11a (in Table 1) is illustrated in the experiments.

5.1.2. Driver-preferred routes and their controller-estimated expected travel times

A two-step off-line approach is used to estimate the driver-preferred route sets and their corresponding time-dependent controller-estimated expected travel times. The first step involves the solution to a UE DTA problem for the entire planning horizon using an average time-dependent demand. This solution provides an initial set of UE routes as input for the next step. The second step involves conducting several simulation runs using the controller-estimated driver behavior model to determine up to five routes for an O-D pair. These routes and their corresponding time-dependent travel times represent, the driver-preferred route sets and the time-dependent controller-estimated expected travel times, respectively.

5.1.3. Actual driver behavior model

In the absence of field data, the actual behavior of the drivers is represented here by a random coefficients path-size multinomial logit model. This model includes travel time, number of nodes, a path-size component used to capture the overlap between routes, and the route recommendation provided by the controller, as the explanatory variables. Details of this model are provided in Paz and Peeta (2007).

It should be noted here that the actual behavior at the individual driver level is currently an inferred quantity in the real-world, though technologies such as global positioning systems can substantially aid in modeling it. That is, in the future, when these technologies are adequately deployed and privacy-related policies are developed, the ability to track individual drivers can provide robust models of actual behavior as well as controller-estimated behavior. In this study, the actual behavior model is deliberately assumed to have a different structure compared to the controller-estimated model. This is to ensure that the study insights are based on conservative analyses and to imply that the actual behavior model is unknown to the controller. However, the controller can estimate the linkages between various factors and aggregate level behavior using past studies and historical data.

5.1.4. Level of responsiveness

The performance of the behavior-consistent approach and the fuzzy on-line calibration model is analyzed under two levels of responsiveness to the information strategies. The first level corresponds to the “less responsive” drivers who are influenced moderately by the provided information. They rely more on past experience and behavioral tendencies to make route choice decisions than on the traffic information. The second level corresponds to the “more responsive” drivers who are significantly influenced by the information. They are more likely to accept the route recommended by the controller. The details of these driver types for the controller-estimated and actual behavior models are provided in Paz and Peeta (2007).

5.1.5. Calibration cases

The 1st day case: this is the case where the controller does not have any information to determine the weight to assign to each behavioral rule used by the controller-estimated driver behavior model. It is the situation faced by the controller on the first day of deploying the behavior-consistent information-based network control strategies. Hence, the controller initially assigns the same weight to all rules. Therefore, the results obtained under this case are conservative and are affected by the initial values adopted for the weights.

The 2nd day case: this represents the case where there is information on the prior values for the weights associated with the controller-estimated driver behavior model. The information on the values of the weights is available after the first day. Here, the controller initially assigns the values computed at the end of the previous day using the on-line calibration model for the behavioral rule weights. This aids computational efficiency as the relative values that drivers assign to the different choice attributes are likely to be the same under normal conditions.

5.1.6. Scenarios

Two scenarios each are evaluated for each level of responsiveness and day case. In the “BC-info-CS” scenario, the system controller uses the full framework of Fig. 1 to determine the information strategies. That is, the information strategies are determined using the behavior-consistent approach, and the calibration model is used to calibrate the parameters of the controller-estimated driver behavior model during each stage. In the “BC-info” scenario, the system controller uses only the shaded boxes in Fig. 1 to determine the information strategies. That is, the calibration model is not used to update the parameters of the controller-estimated

driver behavior model. Hence, the two scenarios differ only in terms of whether the calibration is performed or not.

5.1.7. Performance measures

The effectiveness of the calibration model is measured in terms of its ability to accurately estimate the traffic pattern unfolding over time. The performance measure used here is the average percentage difference between the observed and estimated link traffic counts:

$$\left[\sum_{a \in \Gamma} \frac{|\chi_a^{\rho(\sigma)} - y_a^{\rho(\sigma)}|}{\chi_a^{\rho(\sigma)}} \right] \cdot \frac{100}{|\Gamma|} \quad (10)$$

where $\chi_a^{\rho(\sigma)}$ is the observed count on link a in roll period $\rho(\sigma)$, $y_a^{\rho(\sigma)}$ is the estimated count on link a in roll period $\rho(\sigma)$, and Γ is the set of links for which real-time measurements are obtained.

In Fig. 1, Eq. (10) is also used to determine whether the calibration model needs to be activated in each roll period. In the study experiments, the calibration model is activated if a threshold value of 5% is exceeded for this performance measure.

A second performance measure, the difference between the observed and corresponding SO states, can be computed by replacing $y_a^{\rho(\sigma)}$ in Eq. (10) with the counts obtained from the SO DTA traffic assignment.

5.1.8. Network States

In the study experiments, the “observed” network states are assumed to be the outcome of the actual driver behavior model (Section 5.1.3) in conjunction with the traffic flow simulator. The “estimated” network states are obtained using network loading mechanism described in Section 3.2.

5.1.9. Assumptions

In all scenarios, it is assumed that: (i) the estimated and actual demand are the same, (ii) all drivers with the same O-D pair have the same set of driver-preferred routes, and (iii) the controller-estimated and the actual driver-preferred route sets are the same. Consistent with the study objectives, this is designed to isolate and analyze the effects of the information strategies and the calibration model.

5.2. Results and analysis

5.2.1. The 1st day case

Fig. 7 shows the average percentage differences between observed and estimated traffic counts for the first day on which the calibration model is implemented. Initially, these differences are significant as the behavioral rules are arbitrarily assigned equal weights. However, as the calibration model starts using information from more stages, it is able to significantly reduce these differences. This suggests that the fuzzy calibration model can

calibrate the controller-estimated driver behavior model by reducing the estimation errors. As seen in the figure, the information strategies determined using the calibrated parameters are also able to move the system closer to the SO states. This is because the information strategies based on the calibrated parameters are able to generate better field behavior consistency, implying improved observed system performance. However, the calibration lowers the gap more for the behavior-consistent approach as it focuses on ensuring behavioral realism unlike the idealized SO states by updating the weights for all behavioral rules and all routes.

Fig. 7 also shows that better state consistency is achieved for the “more responsive” drivers when the observed states are compared to the SO DTA states. However, as illustrated in Fig. 8, the system travel time savings are larger for the “less responsive” drivers. Since the behavioral parameters are not calibrated before the first day, the calibration model needs a few stages during the first day to adapt the controller-estimated driver behavior model to the unfolding actual behavior. During these initial stages, the system performance may deteriorate. Since, drivers are more likely to accept route recommendations under the “more responsive” scenario, the negative performance effects can be amplified for the “more responsive” drivers. Consequently, though the calibration is enhanced over time, the initial negative effects cannot be compensated for adequately by the “more responsive” behavior. However, if the initially allocated values for the behavioral parameters are more representative of the actual behavior, the “more responsive” scenario could perform better than the “less responsive” one. Fig. 8 also shows that the BC-info scenario performs worse than the BC-info-CS scenario for both levels of responsiveness. This highlights the importance of calibrating the behavioral parameters. It also reinforces the notion that the determination of information strategies requires a meaningful estimation of driver behavior (Paz and Peeta, 2007).

Fig. 9 shows the weights of the information behavioral *if-then* rule 11a (in Table 1) over time for the O-D pair illustrated in Fig. 6. It indicates that the weights are continuously updated without reaching convergence except for few routes. However, as illustrated in Fig. 7, the calibration model is able to reduce the estimation error though the weights do not converge. Figure 7 also indicates that the percentage difference has lower variability after about stage 15, suggesting the likelihood of multiple solutions for the weights of the behavioral *if-then* rules in Fig. 9. This result is intuitive as only the relative differences in attractiveness (utility) of the alternative routes matters for the controller-estimated driver behavior model. That is, different combination of values for the weights can result in the same route choice probabilities in Eq. (3).

The rule 11a in Fig. 9 increases the attractiveness of a route if that route is recommended. The figure shows that route 1 has higher weights compared to other routes. This implies that a large number of drivers are choosing this route based on the behavior-consistent information. Fig. 10 indicates that route 1 is significantly recommended to drivers by the controller under the behavior-consistent approach. This suggests that the calibration model consistently adapts the controller-estimated driver behavior model to the observed network states.

5.2.2. The 2nd day case

Fig. 11 shows the percentage differences for the second day on which the calibration model is implemented. Their initial values are lower than those for the first day as the controller uses the calibrated parameters from the 1st day at the beginning of the second day. As before, the calibration model can reduce these differences over time. However, the relative improvement is not as significant as on the first day because the initial differences on the second day are smaller.

Fig. 12 shows that the initial negative effects that existed for the first day are no longer present because calibrated parameters from the first day are used. Consequently, as expected, it is observed that the “more responsive” case performs better.

A comparison of Figs. 10 and 13 indicates that the weights are more stable for the 2nd day. This implies that the initial values for the weights can influence their variability. However, our experiments assume that the actual driver behavior is not modified from day 1 to day 2. But, such modification is possible under driver learning processes. Hence, it is possible that there is variability in the values of weights on the second day if that effect were captured. Fig. 10 also illustrates that fewer drivers are recommended to take route 1 for the second day according to the behavior-consistent approach compared to the first day. This is because using calibrated parameters (obtained at the end of the first day) allows the controller to determine more consistent proportions to recommend routes, without the need to over-recommend or under-recommend significantly. For example, the initial negative effects in Fig. 8 are due to over- and under-recommendations of various routes. This is illustrated further in Fig. 10 where route 1 is recommended consistently more for the first day compared to the second day.

6. Concluding comments

This paper develops a fuzzy on-line calibration model to calibrate the parameters of a controller-estimated driver behavior model to enhance system state consistency in an operational context. The controller-estimated driver behavior model is a key component in the determination of behavior-consistent information-based network control strategies. The proposed calibration model fits seamlessly within a rolling horizon framework to deploy the behavior-consistent approach. Thereby, the framework determines the information strategies and updates the parameters associated to the controller-estimated driver behavior model.

The calibration model minimizes the difference between the observed and estimated network states in terms of link traffic counts. The practical deployment of the associated calibration model is aided by the structure of the controller-estimated driver behavior model which uses aggregate level *if-then* rules. This circumvents the need for data at the individual driver level, and the calibration can be based on measurable traffic data. In the context of broader route guidance related calibration problem, the proposed model provides a generalized approach to seamlessly incorporate modeling parameters associated with several components such as O-D demand, traffic flow, and driver response behavior. It implies adding calibration control rules for each parameter type and various data sources.

The study results are based on conservative analyses performed by deliberately having different structures for the controller-estimated driver behavior model and the actual driver behavior model. They suggest that the fuzzy on-line calibration model can effectively update the parameters of the controller-estimated driver behavior model, resulting in significant improvements in terms of the accuracy of the controller-estimated network states. Further, there are substantial benefits in terms of system travel time savings with respect to the no-information scenario when the calibration is performed in conjunction with the behavior-consistent approach. The calibration problem is important for the effective deployment of the behavior-consistent information-based network control strategies. If calibration is not performed, negative effects due to inconsistent driver behavior estimation can be magnified under high levels of driver responsiveness.

In general, the behavior-consistent approach in conjunction with the fuzzy on-line calibration model provides an alternative methodological perspective to address the complex deployment problem associated with the real-time information-based control of vehicular traffic systems. While the study experiment address only behavior model related inconsistencies, it is useful to analyze simultaneously the state inconsistency effects related to traffic flow and O-D demand parameters.

References

- Abdel-Aty, M.A., 1998. Modeling incident-related routing decisions by using a nested logit structure. *Transportation Research Record* 1645, TRB, National Research Council, Washington, DC, 103-110.
- Antoniou, C., 2004. On-line calibration for dynamic traffic assignment. PhD thesis. Department of Civil and Environmental Engineering, MIT.
- Balakrishna, R., Ben-Akiva, M.E., Koutsopoulos, H.N. 2007. Off-line calibration of dynamic traffic assignment: simultaneous demand-supply estimation. In: *Proceedings of the 86th Annual Meeting of the Transportation Research Board, Washington, DC [CD-ROM]*.
- Bekhor, S., Ben-Akiva, M.E., Ramming, M.S., 2006. Evaluation of choice set generation algorithms for route choice models. *Annals of Operations Research* 144 (1), 235-247.
- Bottom, J.A., 2000. Consistent anticipatory route guidance. PhD thesis. Department of Civil and Environmental Engineering, MIT.
- Lotan, T., Koutsopoulos, H. N., 1993. Models for route choice behavior in the presence of information using concepts from fuzzy set theory and approximate reasoning. *Transportation* 20 (2), 129-155.
- Lotan, T., Koutsopoulos, H. N., 1999. Modeling default behavior in the presence of information and its application to the route choice problem. *International Journal of Intelligent Systems* 14 (5), 501-533.
- Mahmassani H. S., 2001. Dynamic network traffic assignment and simulation methodology for advanced system management applications. *Networks and Spatial Economics* 1 (3-4), 267-292.
- Mahmassani, H. S., Hawas, Y.E., Abdelghany, K., Abdelfatah, A., Chiu, Y.-C., Kang, Y., Chang, G.L., Peeta, S., Taylor, R., Ziliaskopoulos, A. 1998. DYNASMART-X; Volume II: analytical and algorithmic aspects. Technical Report ST067-85-Volume II, Center for Transportation Research, University of Texas at Austin.
- Mahmassani, H. S., Jayakrishnan, R., 1991. System performance and user response under real-time information in a congested traffic corridor. *Transportation Research Part A* 25 (5) 293-308.
- Paz, A., Peeta, S., 2007. Behavior-consistent real-time traffic routing under information provision. In: *Proceedings of the 86th Annual Meeting of the Transportation Research Board, Washington, DC. [CD-ROM]*.
- Paz, A., Peeta, S., 2008a. Information-based network control strategies consistent with estimated driver behavior. *Transportation Research Part B*. In press.
- Paz, A., Peeta, S., 2008b. Deployment paradigms for behavior-consistent real-time traffic routing under information provision. In: *Proceedings of the 87th Annual Meeting of the Transportation Research Board, Washington, DC. [CD-ROM]*.
- Peeta, S., Bulusu, S., 1999. Generalized singular value decomposition approach for consistent

- on-line dynamic traffic assignment. Transportation Research Record 1667, TRB, National Research Council, Washington D.C., 77-87.
- Peeta, S., Yu, J.W., 2004. Adaptability of a hybrid route choice model to incorporating driver behavior dynamics under information provision. IEEE Transactions on Systems, Man, and Cybernetics Part A: Systems and Humans 34, 243-256.
- Peeta, S., Yu, J.W., 2005. A hybrid model for driver route choices incorporating en-route attributes and real-time information effects. Networks and Spatial Economics 5 (1), 21-40.
- Peeta, S., Yu, J.W., 2006. Behavior-based consistency-seeking models as deployment alternatives to dynamic traffic assignment models. Transportation Research Part C 14 (2), 114-138.
- Srinivasan, K.K., Mahmassani, H. S., 2000. Modeling inertia and compliance mechanisms in route choice behavior under real-time information. Transportation Research Record 1725, TRB, National Research Council, Washington D.C., 45-53.
- Tsoukalas, L., Uhrig, R., 1997. Fuzzy and Neural Approaches in Engineering. John Wiley and Sons, Inc., New York, NY.

Table 1
Behavioral *if-then* rules for the controller-estimated driver behavior model

Category	Rule #	LHS	RHS
Controller-estimated driver-expected travel time (<i>TT</i>)	1	If <i>TT</i> is Very Low (VL)	Driver will choose the alternative (O)
	2	If <i>TT</i> is Low (L)	Driver will probably choose the alternative (PO)
	3	If <i>TT</i> is Medium (M)	Driver is indifferent to the alternative (I)
	4	If <i>TT</i> is High (H)	Driver will probably not choose the alternative (PN)
	5	If <i>TT</i> is Very High (VH)	Driver will not choose the alternative (N)
Route complexity (<i>NN</i>)	6	If <i>NN</i> is Very Low (VL)	Driver will choose the alternative (O)
	7	If <i>NN</i> is Low (L)	Driver will probably choose the alternative (PO)
	8	If <i>NN</i> is Medium (M)	Driver is indifferent to the alternative (I)
	9	If <i>NN</i> is High (H)	Driver will probably not choose the alternative (PN)
	10	If <i>NN</i> is Very High (VH)	Driver will not choose the alternative (N)
Prescriptive information (<i>Y</i>) for more responsive drivers	11a	If <i>Y</i> is "Route is Recommended" (RR)	Driver will choose the alternative (O)
	12a	If <i>Y</i> is "Route Was Recommended" (RWR)	Driver will probably choose the alternative (PO)
	13a	If <i>Y</i> is "Route is Not Recommended" (RNR)	Driver will not choose the alternative (N)
Prescriptive information (<i>Y</i>) for less responsive drivers	11b	If <i>Y</i> is "Route is Recommended" (RR)	Driver will probably choose the alternative (PO)
	12b	If <i>Y</i> is "Route Was Recommended" (RWR)	Driver is indifferent to the alternative (I)
	13b	If <i>Y</i> is "Route is Not Recommended" (RNR)	Driver will probably not choose the alternative (PN)

Table 2
Calibration control *if-then* rules

Rules for weights 1, 2, 6, 7, 11a, 11b, 12a (in Table 1)		Error (e)				
The consequent (RHS) is O or PO		NL	NS	ZR	PS	PL
	NL	NL	NS	NS	ZR	PS
Change in Error (Δe)	NS	NL	NS	ZR	ZR	PS
	ZR	NS	NS	ZR	PS	PS
	PS	NS	NS	ZR	PS	PL
	PL	NS	ZR	ZR	PS	PL
Rules for weights 3, 8, 12b (in Table 1)		Error (e)				
The consequent (RHS) is I		NL	NS	ZR	PS	PL
	NL	NS	NS	ZR	NS	NS
Change in Error (Δe)	NS	NS	NS	ZR	NS	NS
	ZR	NS	NS	ZR	NS	NS
	PS	NS	NS	ZR	NS	NS
	PL	NS	NS	ZR	NS	NS
Rules for weight 4, 5, 9, 10, 13a, 13b (in Table 1)		Error (e)				
The consequent (RHS) is N or PN		NL	NS	ZR	PS	PL
	NL	PS	PS	ZR	NS	NS
Change in Error (Δe)	NS	PS	PS	ZR	NS	NS
	ZR	PS	PS	ZR	ZR	NS
	PS	PS	ZR	ZR	NS	NS
	PL	PS	ZR	NS	NS	NS

where:

- NL = Negative large
- NS = Negative small
- ZR = Zero
- PS = Positive small
- PL = Positive large

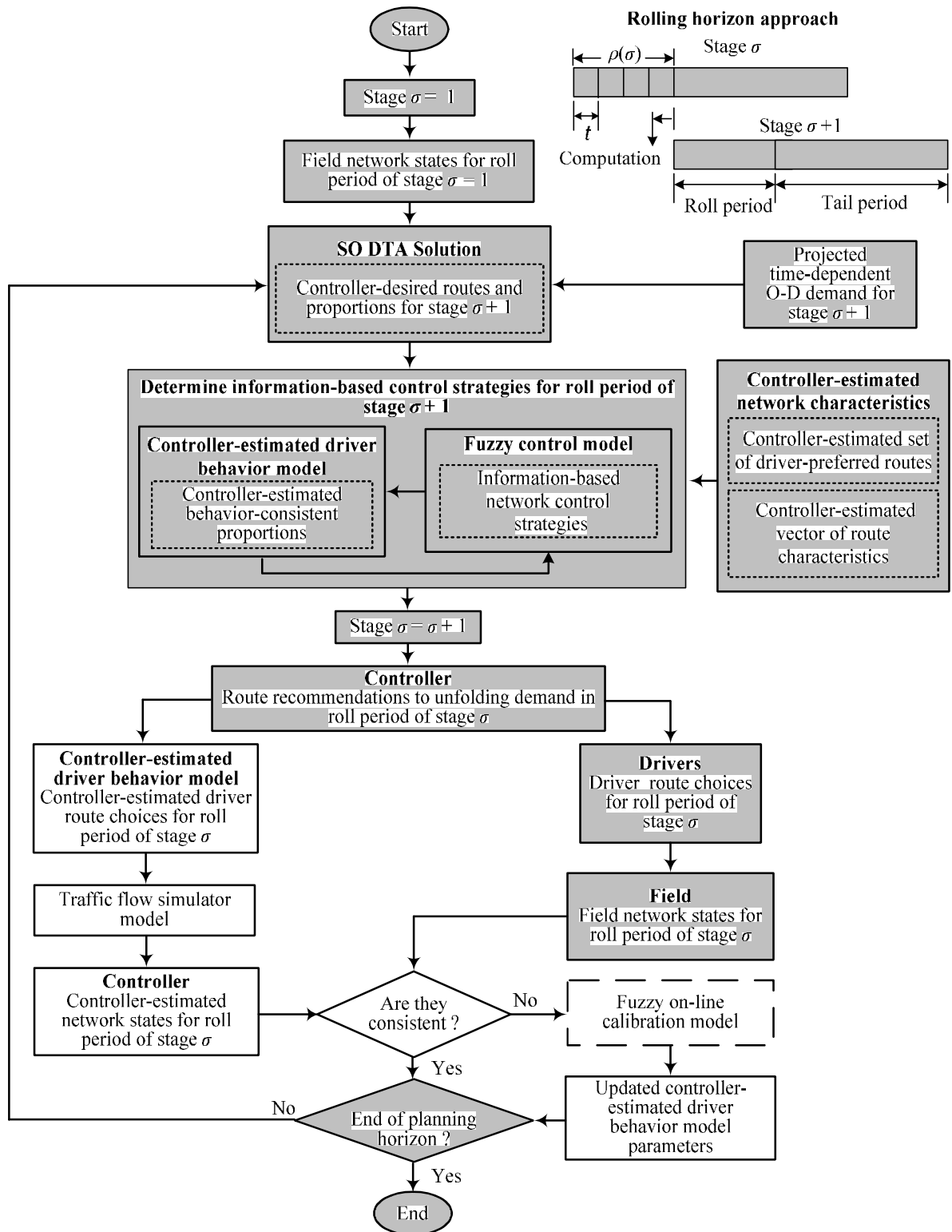


Fig. 1. Conceptual framework for the behavior-consistent real-time traffic routing and calibration problem.

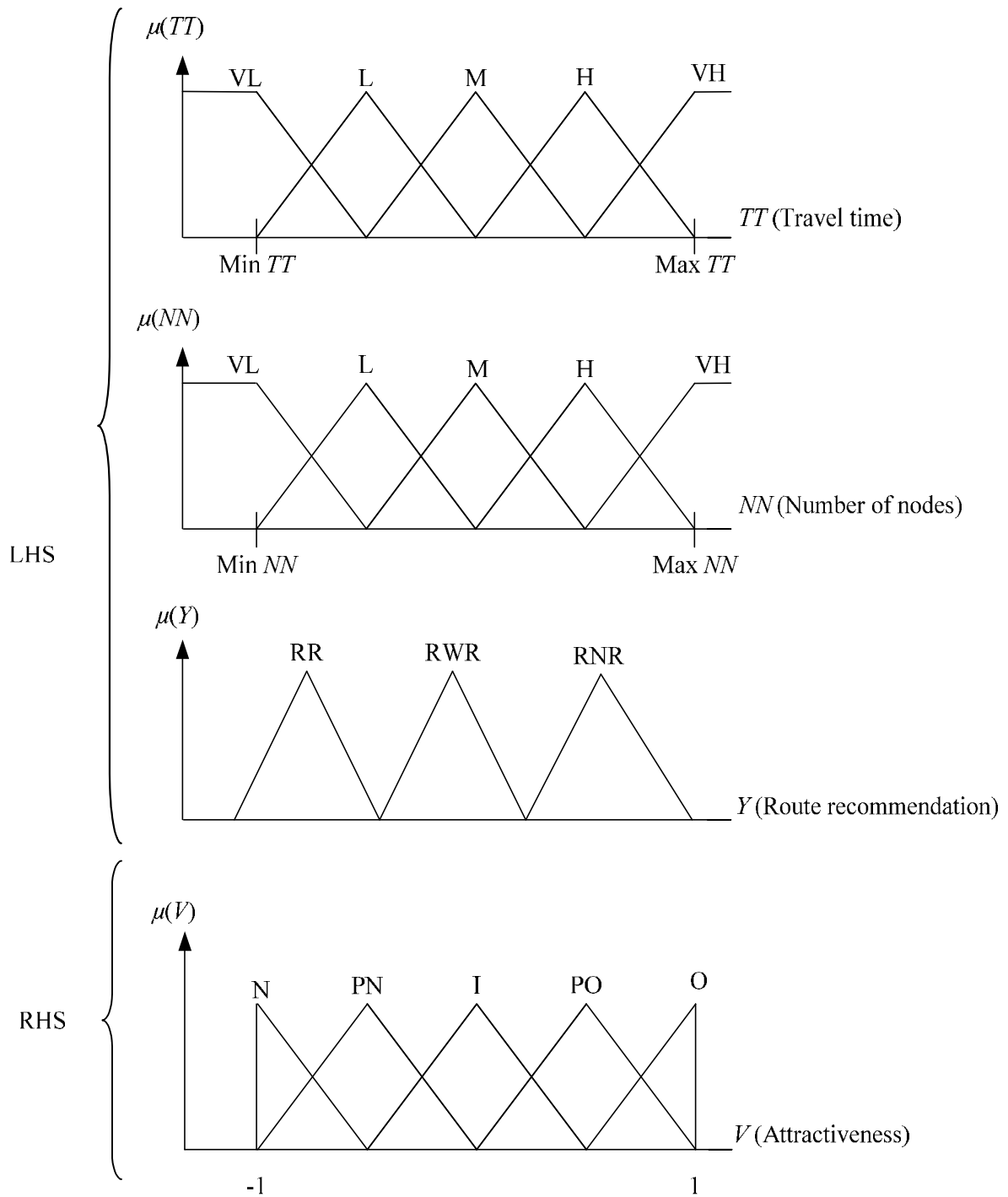


Fig. 2. Membership functions for the controller-estimated driver behavior model.

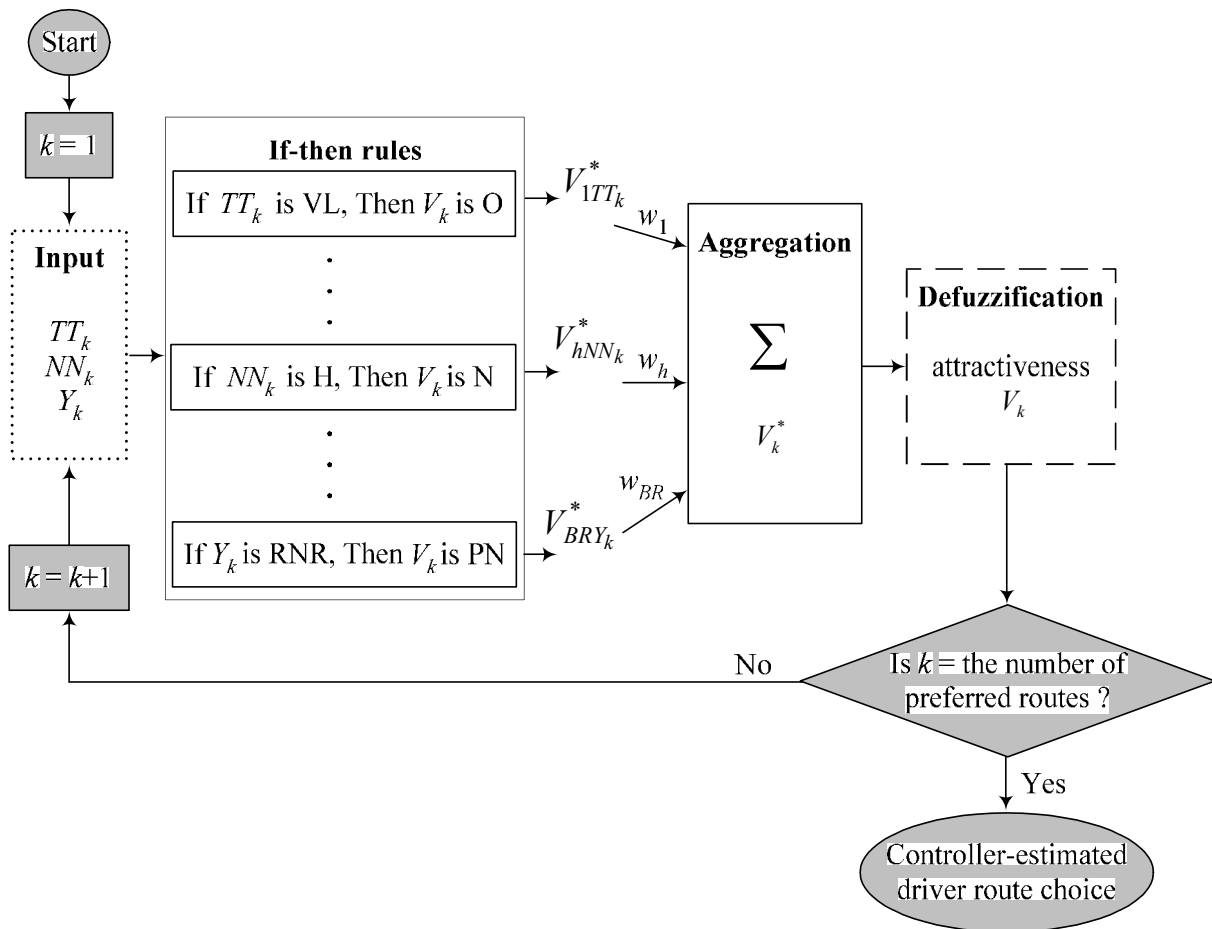


Fig. 3. Conceptual framework for the controller-estimated driver behavior model.

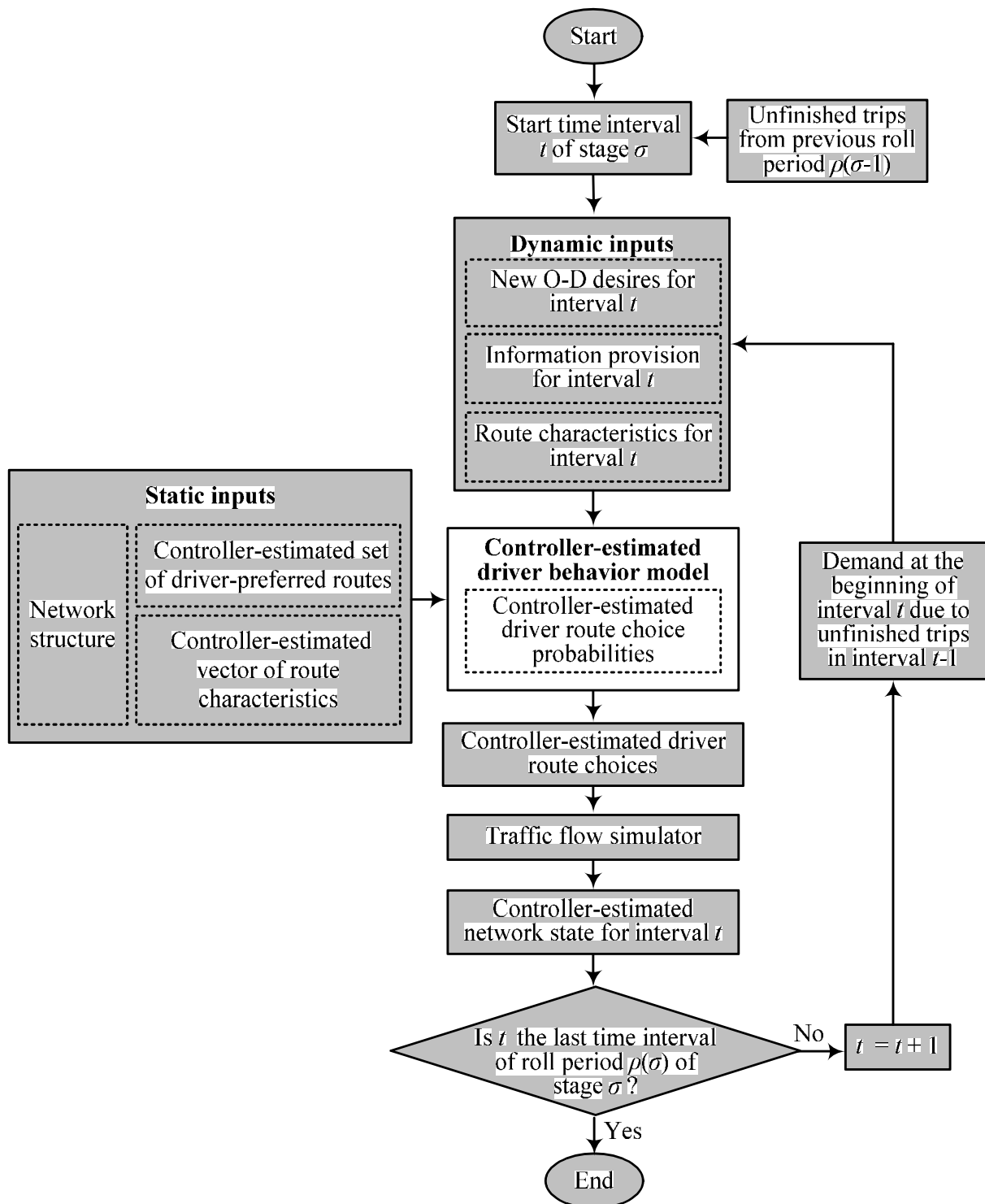


Fig. 4. Network loading for roll period of stage σ using the controller-estimated driver behavior model.

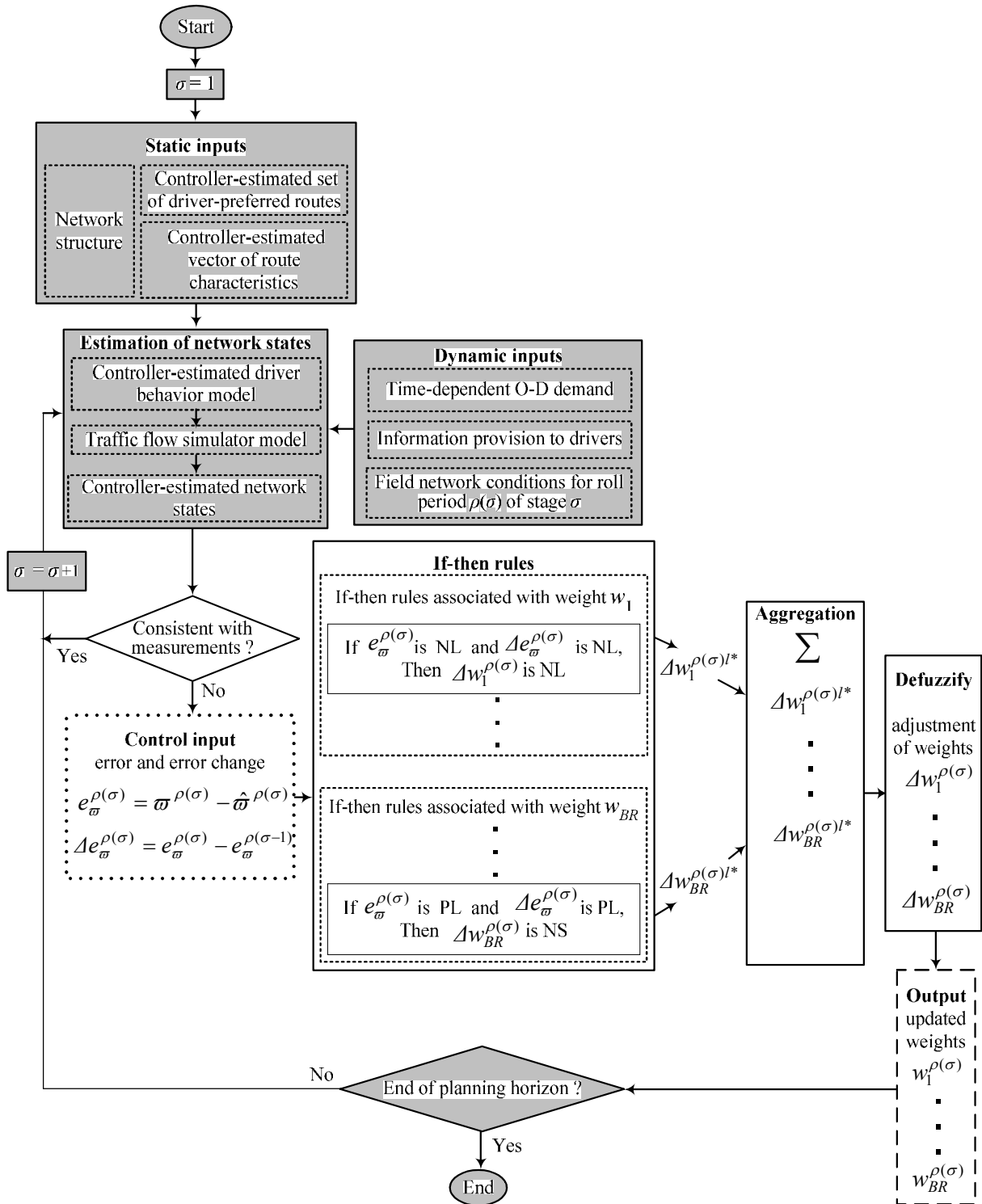


Fig. 5. Conceptual framework for the fuzzy on-line calibration model.

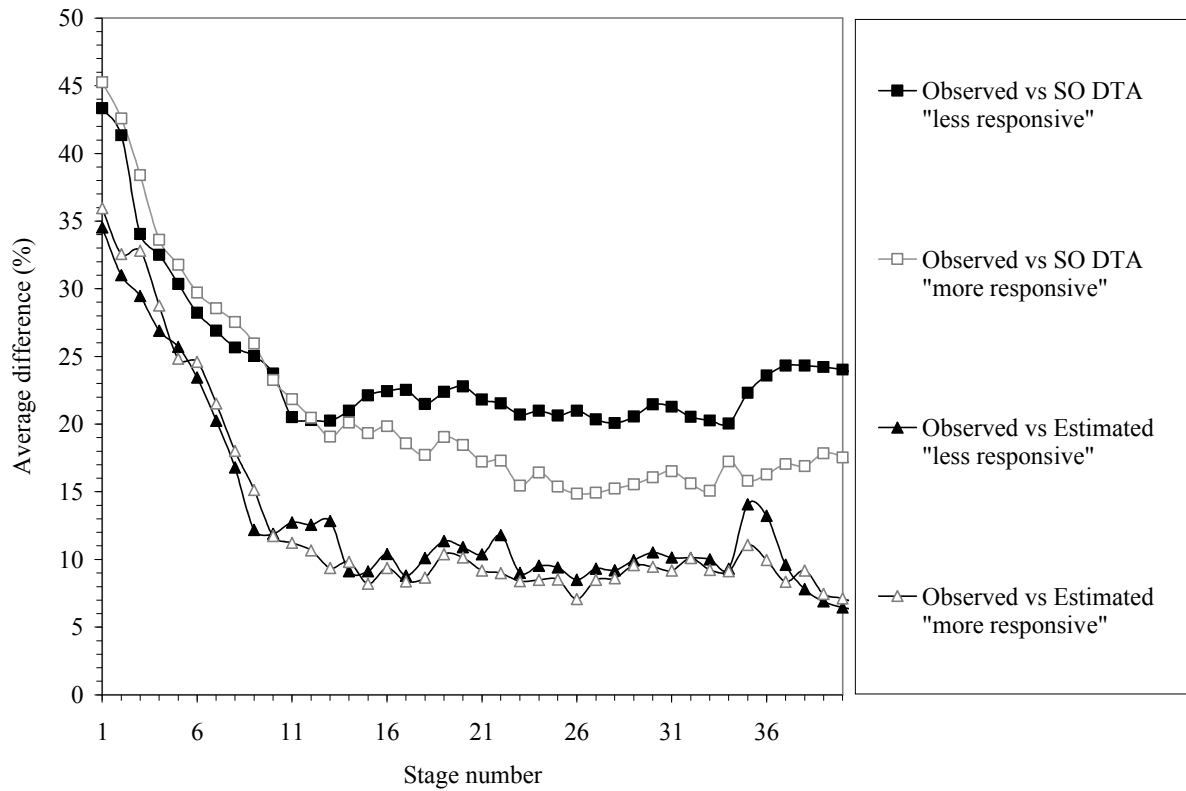


Fig. 7. Average percentage difference between the observed and estimated/SO traffic counts for the 1st day for the BC-info-CS scenario.

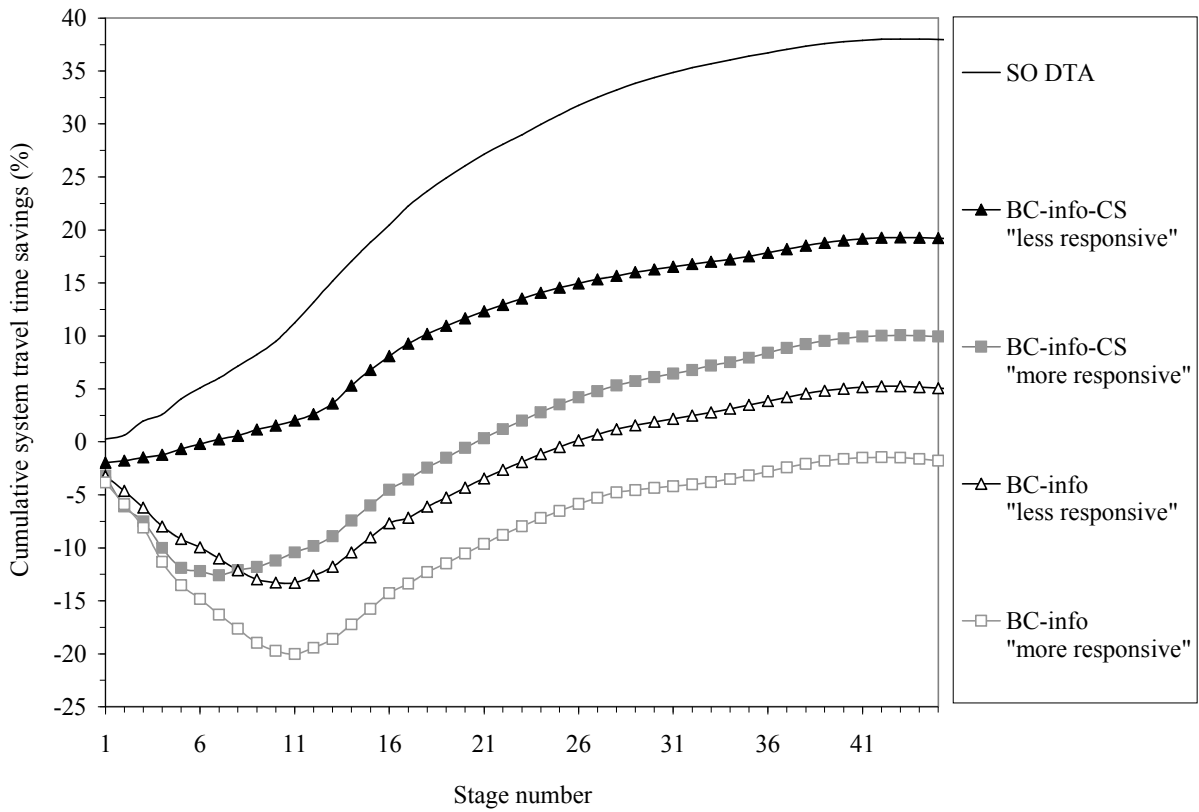


Fig. 8. Cumulative system travel time savings benchmarked against the no-information case (base-case) for the 1st day.

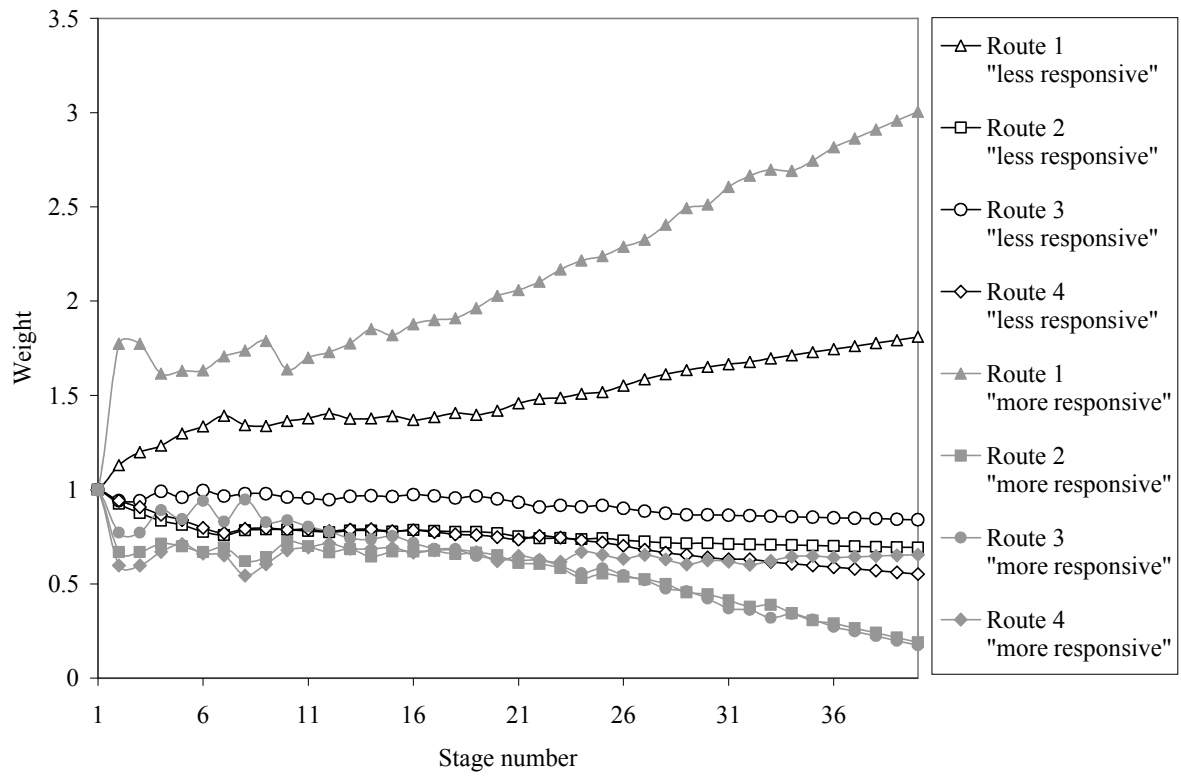


Fig. 9. Weights for behavioral *if-then* rule 11a for the 1st day.

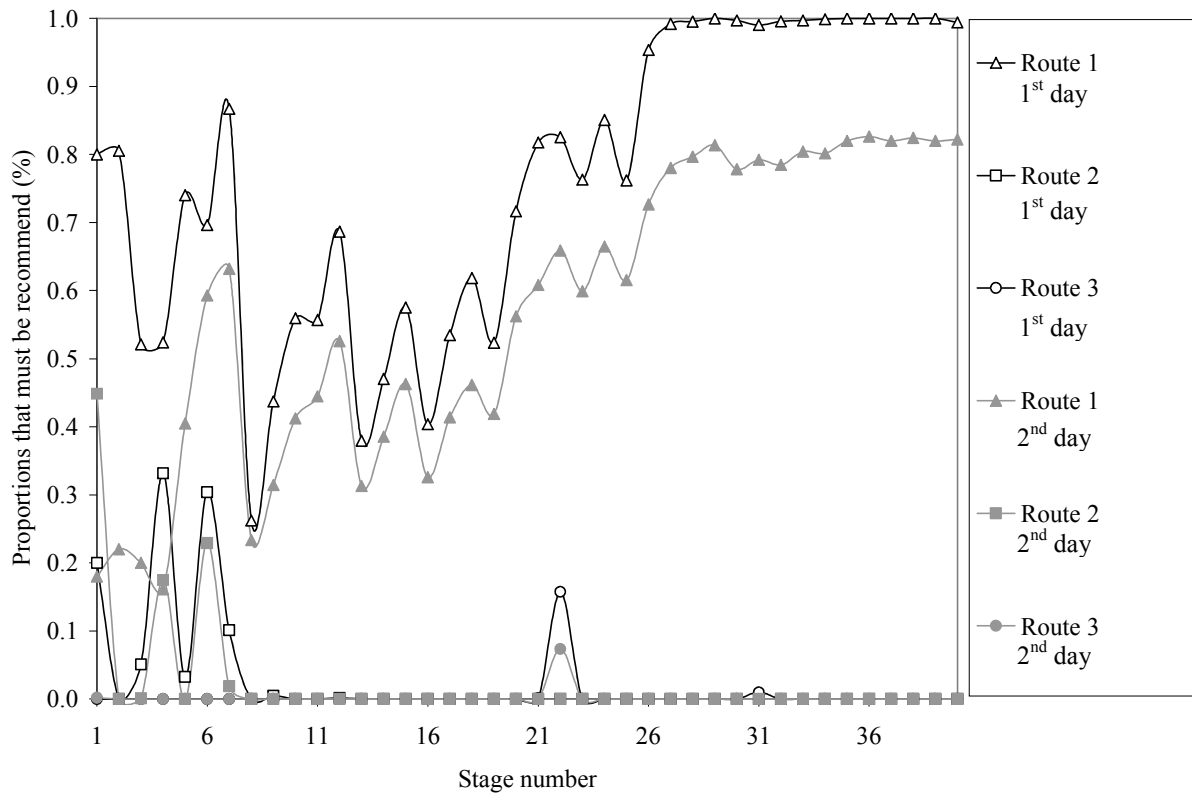


Fig. 10. Proportion of drivers that must be recommended to take specific routes for "less responsive" drivers.

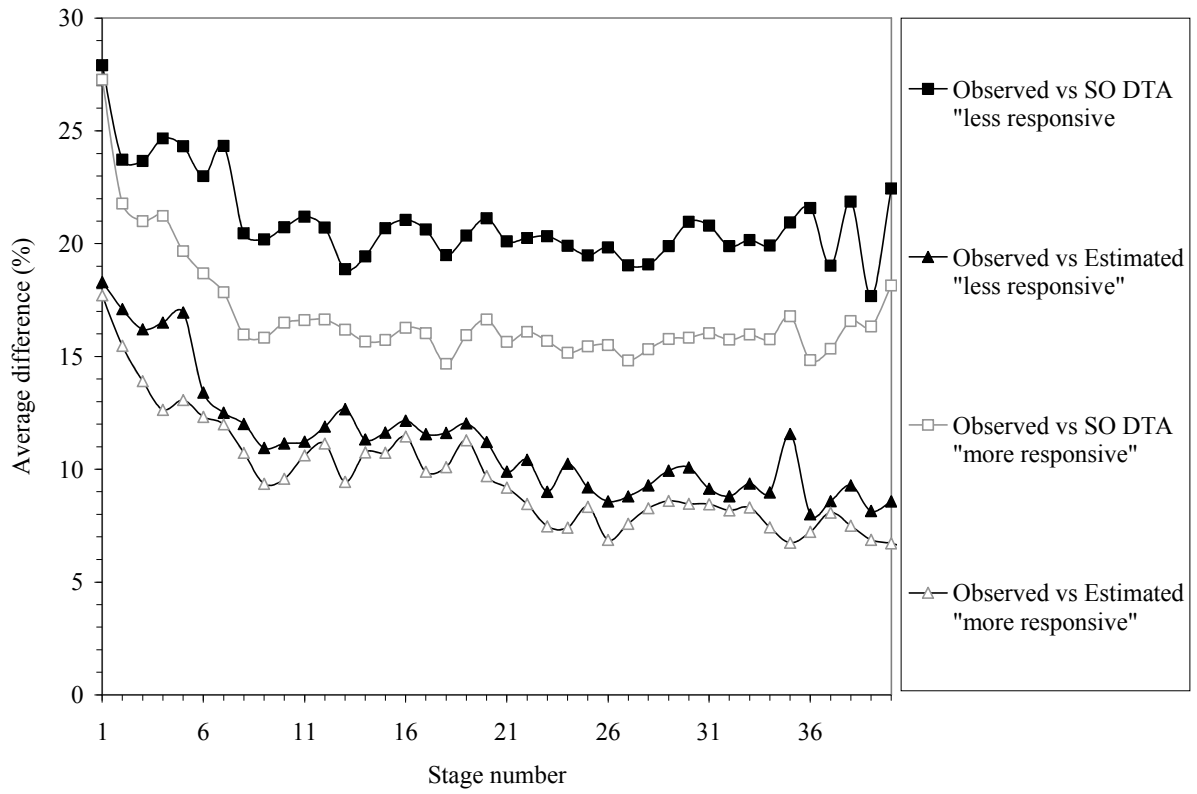


Fig. 11. Average percentage difference between the observed and estimated/SO traffic counts for the 2nd day for the BC-info-CS scenario.

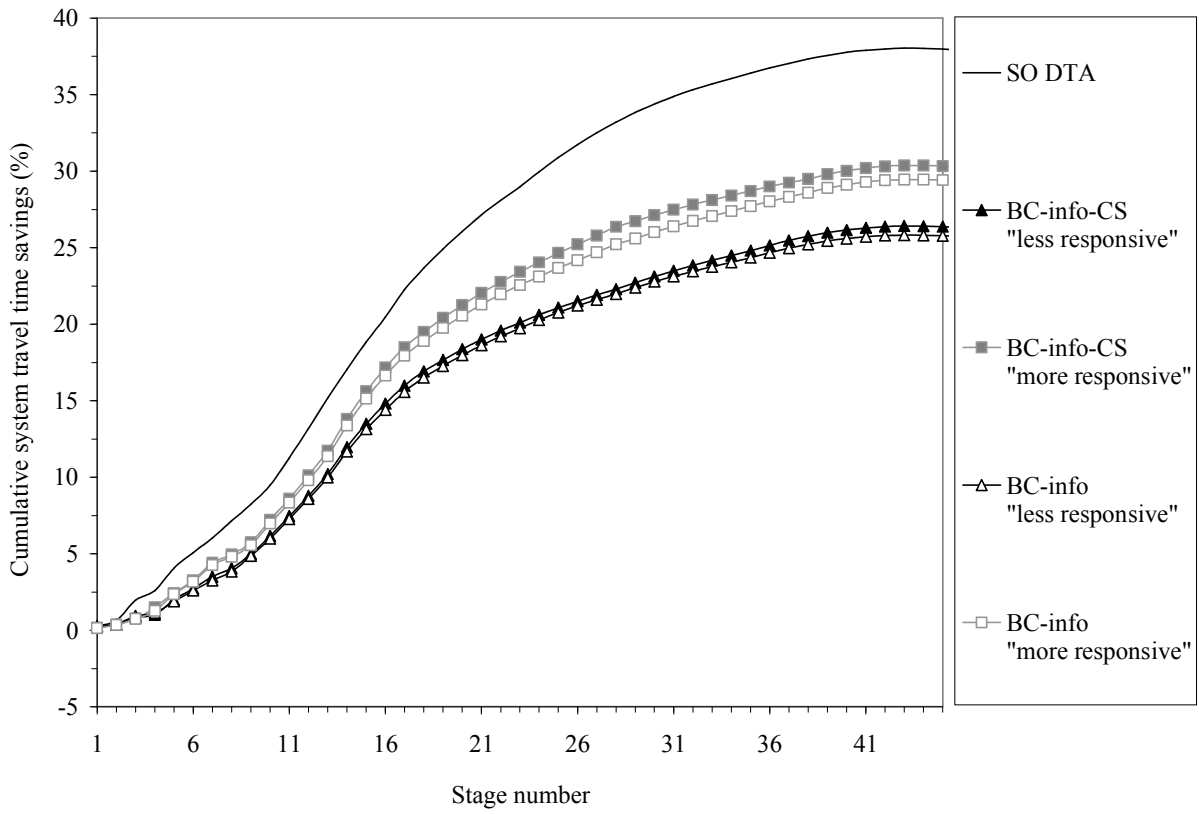


Fig. 12. Cumulative system travel time savings benchmarked against the no-information case (base-case) for the 2nd day.

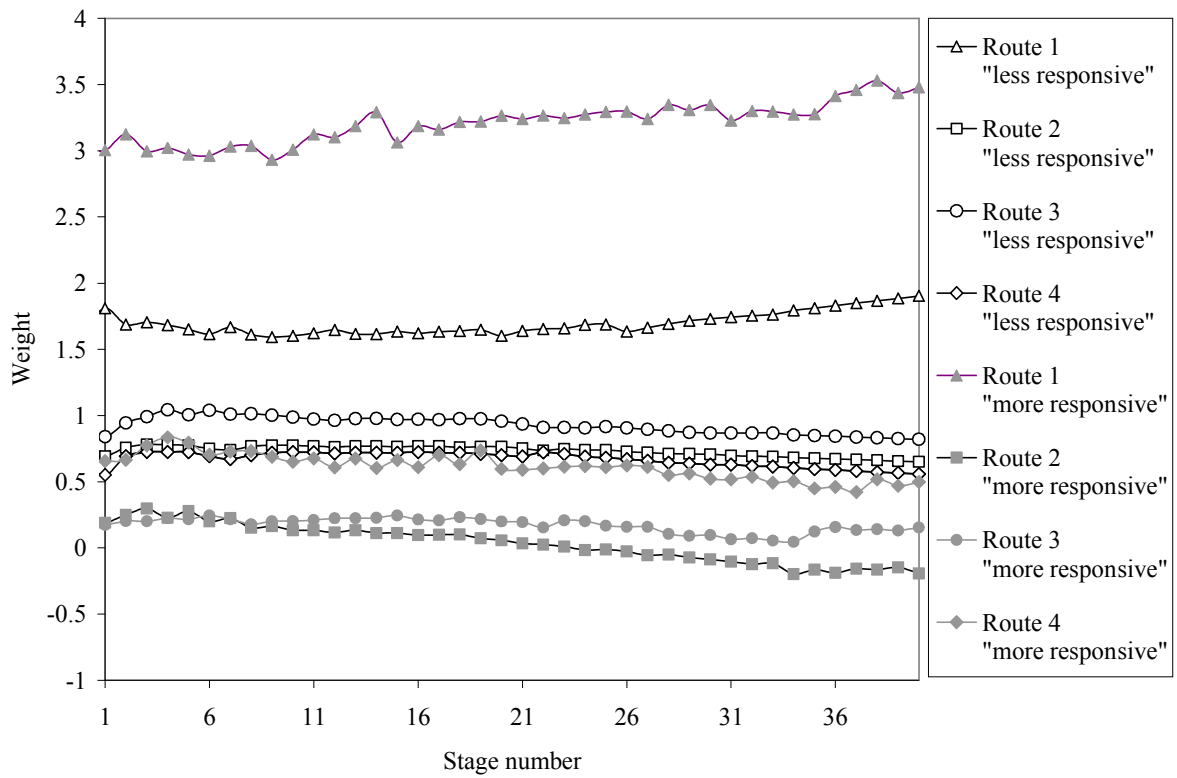


Fig. 13. Weights for behavioral *if-then* rule 11a for the 2nd day.