An intersection-movement-based stochastic dynamic user optimal route choice model for assessing network performance

Jiancheng Long
Professor
School of Transportation Engineering
Hefei University of Technology
Hefei 230009, China
Tel: 86-551-63831101
E-mail: jianchenglong@hfut.edu.cn

W.Y. Szeto[*]
Assistant Professor
Department of Civil Engineering
The University of Hong Kong
Pokfulam Road
Hong Kong
Tel: +852-28578552
Email: ceszeto@hku.hk

Hai-Jun Huang
Professor
School of Economics and Management
Beijing University of Aeronautics and Astronautics
Beijing 100191, China
Tel: 86-10-8231-6608
E-mail: haijunhuang@buaa.edu.cn

Ziyou Gao
Professor
School of Traffic and Transportation
Beijing Jiaotong University
Beijing 100044, China
Tel: 86-10-5168-8193
Fax: 86-10-5168-7127
E-mail: zygao@bjtu.edu.cn

---

[*] Corresponding author

Abstract: Different from traditional methods, this paper formulates the logit-based stochastic dynamic user optimal (SDUO) route choice problem as a fixed point (FP) problem in terms of intersection movement choice probabilities, which contain travelers' route information so that the realistic effects of physical queues can be captured in the formulation when a physical-queue traffic flow model is adopted, and that route enumeration and column generation heuristics can be avoided in the solution procedure when efficient path sets are used. The choice probability can be either destination specific or origin-destination specific, resulting into two formulations. To capture the effect of physical queues in these FP formulations, the link transmission model is modified for the network loading and travel time determination. The self-regulated averaging method (SRAM) was adopted to solve the FP formulations. Numerical examples were developed to illustrate the properties of the problem and the effectiveness of the solution method. The proposed models were further used to evaluate the effect of information quality and road network improvement on the network performance in terms of total system travel time (TSTT) and the cost of total vehicle emissions (CTVE). Numerical results show that providing better information quality, enhancing link outflow capacity, or constructing a new road can lead to poor network performance.

Keywords: Dynamic traffic assignment; Stochastic dynamic user optimal; Intersection movement; Fixed point problem; Vehicle emissions.

1. Introduction

Traffic congestion and emissions are major problems in Hong Kong and many other urban cities. These problems can be handled by appropriate transportation planning and traffic management, with the use of Dynamic Traffic Assignment (DTA) models.

DTA models can be developed by either the simulation-based approach (e.g., Yagar, 1971; Mahmassani, 2001; Florian et al., 2008; Tian and Chiu, 2014) or the analytical approach (see Peeta and Ziliaskopoulos, 2001 and Jiang et al., 2011 for comprehensive reviews). The simulation approach focuses on enabling practical deployment of the DTA models for realistic highway networks, the applicability of the models in real-life highway networks, and the ability of the models to adequately capture traffic dynamics and microscopic driver behavior such as lane changing. However, the solution properties of the models, such as solution existence and uniqueness, are not guaranteed and cannot be determined in advance. The analytical approach often formulates DTA problems in terms of certain well-known mathematical problems, such as mathematical programming problems (e.g., Merchant and Nemhauser, 1978a,b; Carey, 1987; Carey and Subrahmanian, 2000; Ziliaskopoulos, 2000; Waller and Ziliaskopoulos, 2006b; Nie, 2011; Waller et al., 2013), optimal control problems (e.g., Friesz et al., 1989; Ran et al., 1993), variational inequality (VI) problems (e.g., Friesz et al., 1993; Ran and Boyce, 1996; Chen and Feng, 2000; Huang and Lam, 2002; Han, 2003), nonlinear complementarity problems (NCP) (e.g., Wie et al., 2002; Ban et al., 2008), fixed-point problems (e.g., Smith, 1993; Lim and Heydecker, 2005; Szeto et al., 2011), differential complementarity systems (e.g., Ban et al., 2012), and differential VI problems (e.g., Friesz et al., 2001; Han et al., 2013c). Different from the simulation-based DTA models, the solution properties can be determined beforehand.

DTA models have two fundamental components: the travel choice principle and the traffic flow component (Szeto and Lo, 2006). The travel choice principle depicts travelers' propensity to travel, e.g., how they select their routes, departure times, modes, or destinations. Travel time is one important element in such choices. Commonly adopted travel choice principles include the dynamic user optimal (DUO) principle (e.g., Friesz et al., 1993, 2013b; Ran and Boyce, 1996; Yang and Meng, 1998; Huang and Lam, 2002; Lo and Szeto, 2002a,b; Waller and Ziliaskopoulos, 2001, 2006a; Golani and Waller, 2004; Han, 2007; Ng and Waller, 2012; Ukkusuri et al., 2012; Iryo, 2013; Blumberg-Nitzani and Bar-Gera), the SDUO principle (e.g., Ran and Boyce, 1996; Chen and Feng, 2000; Han, 2003; Lim and Heydecker, 2005; Szeto et al., 2011) and the dynamic system optimal (DSO) principle (e.g., Merchant and Nemhauser, 1978a,b; Carey, 1987, 2001; Li et al., 1999; Carey and Subrahmanian, 2000; Ziliaskopoulos, 2000; Nie, 2011; Doan and Ukkusuri, 2012; Carey and Watling, 2012; Qian et al., 2012; Han et al., 2013a, b; Ma et al., 2014; Mesa-Arango and Ukkusuri, 2014; Shen and Zhang, 2014). The DUO/SDUO/DSO principle assumes that travelers select their routes and/or departure times to minimize their individual actual/perceived/marginal travel cost, such as travel time.

The traffic flow component depicts how traffic propagates inside a traffic network and hence governs the network performance in terms of travel time (see e.g., Szeto, 2008; Sumalee et al., 2012;

Ngoduy, 2013; Zhang et al., 2013; Zhong et al., 2013; Zhu et al., 2013; Balijepalli et al., 2014; Chen et al., 2014; Chiabaut et al., 2014). This is sometimes referred to as a dynamic network loading (DNL) model. The existing approaches for developing DNL models can be broadly classified into two categories: non-physical queue models and physical queue models. Exit functions (e.g., Merchant and Nemhauser, 1978a,b; Carey, 1987, 1990; Carey and Srinivasan, 1993), link performance functions (e.g., Ran and Boyce, 1996; Chen and Hsueh, 1998; Ban et al., 2008) and the point queue models (e.g., Huang and Lam, 2002; Nie and Zhang, 2005) can be put under the first category. These models have a simpler calculation but fail to capture some fundamental traffic dynamics such as queue spillback. The second category includes advanced exit flow models which are developed based on either Daganzo's (1994, 1995) solution scheme (i.e., Cell Transmission Model (CTM)) or Newell's (1993) solution scheme to the Lighthill and Whitham (1955) and Richards (1956) (LWR) hydrodynamic model of traffic flow (see Kuwahara and Akamatsu, 2001; Lo and Szeto, 2002a,b for example). They can describe traffic dynamics on a road network, including shock waves and queue spillback over multiple links, and are popularly applied to calculate travel times for DTA models in the past decade (e.g., Kuwahara and Akamatsu, 2001; Lo and Szeto, 2002a,b; Szeto and Lo, 2004; Szeto et al., 2011). Recently, Yperman (2007) proposed the link transmission model (LTM) which can be considered as a combination of Daganzo's (1994, 1995) CTM with a triangular fundamental diagram and Newell's (1993) solution scheme. Since each whole link is treated as one cell, the LTM's computational efficiency is much higher than that of classic numerical solution schemes for the LWR model, whilst retaining the same accuracy when the true fundamental diagram is triangular.

The success of the resultant analytical DTA model in capturing the effect of queue spillback depends on not only the choice of traffic flow models but also the choice of decision variables used in the formulation. Traditionally, either link or path inflow variables are used in the model and the corresponding models are referred to as link-based models (e.g., Friesz et al., 1989; Ran and Boyce, 1996; Wie et al., 2002; Ban et al., 2008) and path-based models (e.g., Friesz et al., 1993; Chen and Feng, 2000; Huang and Lam, 2002; Lo and Szeto, 2002a,b; Szeto and Lo, 2004, 2006; Lim and Heydecker, 2005; Szeto et al., 2011; Meng and Khoo, 2012). Link-based models do not require having the path set information in advance, in which the path set can be large even for a medium-scale highway network. Hence, these models can avoid path enumeration and path set generation heuristic in the solution procedure and have the potential to be applied to large-scale highway networks. Link-based DTA models cannot, however, capture queue spillback since the link flow variables cannot model traffic at intersections.

In contrast, path-based models have important information, such as path inflows and the path set, to model traffic at diverges and merges. Path-based models can, therefore, track spillback queues when a realistic traffic flow or DNL model is used. Path-based models also have the advantage that stochastic assignment using logit-type models can be applied easily, ensuring much faster convergence to an equilibrium than a deterministic approach. Nevertheless, the main disadvantage of such models is that they require explicit enumeration of the path choice set, which can be very time consuming, even

for medium networks. Hence, for large-scale network applications, path enumeration has not been used to obtain the path set. Instead, path-set generation (e.g., Bliemer et al., 2004) has been used in these applications, which have generated paths when needed. Also, unused paths can be deleted. However, this path set generation and deletion procedure is a heuristic and solution convergence may not be guaranteed. Some smarter ways of using route sets, such as the concept of subpaths, were developed to improve the efficiency of commuting resources (e.g., Chabini, 2001; Raadsen, et al. 2010). In spite of these, it is unclear whether they can be applied to SDUO problems. For logit-based SDUO models, the dynamic stochastic network loading (DYNASTOCH) method, which is an extension of Dial's (1971) stochastic network loading (STOCH) method for the static stochastic user equilibrium (SUE) assignment problem, was usually adopted in the solution procedure to avoid explicit enumeration of the path choice set (Ran and Boyce, 1996; Han, 2003). Nonetheless, the optimal solutions obtained by this method are link inflow rates, which do not contain the travelers' path information, and hence traffic dynamics, including queue spillback and junction blockage, cannot be captured in link-based models unless that inflow rates are augmented by other variables (such as exit flow, queue lengths) and auxiliary variables to model physical queues.

To retain the advantages of both the link-based and the path-based models, Long et al. (2013b) proposed an intersection-movement-based DTA model. They formulated the DUO traffic assignment problem in terms of approach proportions, i.e., the proportion of traffic on the current link or node that select a downstream link when leaving an intersection (or a node). This definition requires either two adjacent links or one origin and one of its outgoing links to define an intersection movement. This is different from the classical definition that only downstream link is used to define the proportion. An approach proportion implicitly contains the travelers' path information since a path can be deduced by checking the downstream links involved in defining the approach proportions from the origin to the destination. As a result, this type of models can retain the advantages of both the link-based models and the path-based models: 1) Path enumeration and path generation heuristics can be avoided in the solution procedure, and; 2) the realistic effects of physical queues can be captured when a physical queue DNL model is adopted. Therefore, the intersection-movement-based DTA formulation is superior with the other two types of formulations when the effect of physical queues is considered. However, Long et al. (2013b) only considered the DUO principle. Whether this intersection-based approach can be applied to DTA with a more general principle such as the SDUO principle is questionable.

In this paper, we formulate the SDUO problem as a FP problem in terms of intersection movement choice probabilities, which can also be interpreted as approach proportions. An intersection movement contains travelers' path information so that queue spillback can be captured when a realistic DNL model is used. The choice probabilities can be destination specific or origin-destination specific, resulting into two models. The decision variables of the two proposed DTA models are the time-dependent intersection movement choice probabilities, which are used as the inputs of a DNL model. The corresponding link flow and link travel time can be obtained by a DNL model. The proposed FP

models are general in the sense that it can incorporate any types of DNL models to depict traffic flow propagation inside the traffic network and determine travel times as long as the model can produce a unique mapping from approach proportion to link travel time. Because of the computation efficiency of the LTM and its ability to capture queue spillback and shockwave phenomena, we develop a DNL model based on the LTM. However, the LTM requires path flow rates as the inputs. Therefore, we propose a modified LTM (MLTM) in which the inputs are intersection movement choice probabilities. The MLTM is used to generate the cumulative inflows into and outflows from each link over time and derive dynamic link travel time functions.

Both intersection-movement-based SDUO models are solved by the self-regulated averaging method (SRAM) proposed by Liu et al. (2009). The SRAM was initially developed to solve the deterministic network SUE assignment problem (e.g., Liu et al. 2009; Long et al., 2010, 2014), and further used to solve a multi-class doubly SDUO problem (Szeto et al., 2011). The SRAM is faster than the method of successive averages (MSA), which is a widely used method in solving traffic assignment problems (e.g., Sheffi, 1985; Ran and Boyce, 1996; Nielsen et al., 1998; Han, 2003). For DNL, other than using the MLTM to estimate link travel times, we avoid path enumeration by using a method modified from DYNASTOCH.

This paper also provides numerical examples to illustrate the properties of the problem and the effectiveness of the solution method. The proposed models were further used to evaluate the effect of information quality and road network improvement on the network performance in terms of total system travel time (TSTT) and the cost of total vehicle emissions (CTVE). Numerical results show that providing better information quality, enhancing link outflow capacity or constructing a new road may lead to poor network performance.

The main contribution of this paper includes the following:

1) This paper proposes a novel methodology which adopts intersection-movement-based decision variables and Dial's (1971) stochastic network loading (SNL) concept to formulate logit-based SDUO route choice problems as fixed point (FP) problems; to the best of our knowledge, this paper is the first one to formulate the SDUO problems using both the intersection movement and Dial's concepts; moreover, FP formulations define the equilibrium flow pattern through local conditions, unlike variational inequality formulations, and provide an effective way to include equilibrium conditions within network design or O-D matrix update from traffic counts (Cantarella, 1997);

2) This paper analyzes the mathematical properties of the problems rigorously. In particular, this paper proves that the proposed intersection-movement-based SDUO route choice models follows logit-based SDUO route choice principle, and are equivalent to the link-based model in the literature;

3) This paper proposes intersection-movement-based SNL methods, examines its properties, and develops a solution approach based on these methods to solve the SDUO problems; this

solution approach does not relying on path generation heuristics, path enumeration techniques and subpath concepts to solve the problems and perform SNL;

4) This paper illustrates the properties of the SDUO problems using simple numerical examples. Specifically, the paper shows that providing better information quality, enhancing link outflow capacity or constructing a new road can lead to poor network performance in terms of TSTT and CTVE, and that a higher travel demand may result in lower CTVE.

The remainder of this paper is organized as follows. In the next section, each of the two intersection-movement-based SDUO problems is formulated as a FP problem. In Section 3, the MLTM, a discretized travel time model and the measure of network performance are introduced. In Section 4, the SRAM is presented to solve the proposed SDUO models. Numerical examples are given in Section 5, and finally, conclusions are provided in Section 6.

2. Model formulation

In this section, we firstly review the path-based SDUO problems and two definitions of an efficient path set. Based on these definitions, this section proposed two intersection-movement-based SDUO models with no restriction on the choice of the traffic flow model adopted.

2.1. Problem setting and notations

We consider a road network $G(N, A)$, where N and A denote the sets of nodes and links, respectively. Link $a = (l_a, h_a)$ is the link with the tail node l_a and the head node h_a, $a \in A$. $A(i)$ is the set of links whose tail node is i; $B(i)$ is the set of links whose head node is i. R and S denote the sets of origin nodes and destination nodes, respectively. K_{rs} denotes the path set of origin-destination (OD) pair (r, s). We discretize the time period T of interest and the departure time period T_D into finite sets of time intervals $K = \{k = 1, 2, \cdots, \underline{K}\}$ and $K_D = \{k = 1, 2, \cdots, \underline{K}_D\}$, respectively. Let δ be the interval length such that $\delta \underline{K} = T$ and $\delta \underline{K}_D = T_D$. Without loss of generality, the unit of time used in this paper is in time interval, and we assume that each traveler travel from his/her origin to his/her destination by his/her vehicle. This means that the occupancy of each vehicle is assumed to equal one. Following most of DTA works (e.g., Ran et al., 1993; Wie et al., 2002; Ban et al., 2008; Friesz et al., 2013; Waller et al., 2013), the network is assumed to be empty initially. We note that if the network was not empty initially, the procedures of dynamic network loading and travel time determination were required slight modifications to cater the traffic initially inside the network, and the travel times of flows entering the network were increased if the initial traffic contributed to congestion to the flows and remained unchanged otherwise. However, the assumption of the presence of initial traffic in the network does not affect the proofs of all the propositions in general in this paper because the mapping between travel time and traffic flow is still unique. The following notations are adopted to formulate the SDUO problem:

$q^{rs}(k)$ traffic demand generated at origin r during interval k and traveling to destination s.

$u_a(k)$ inflow into link a during interval k.

$u_a^s(k)$ inflow into link a during interval k and to destination s.

$u_{ab}^s(k)$ inflow into link a during interval k and through link $b \in A(h_a)$ to destination s.

$u_a^{rs}(k)$ inflow that departs from origin r, enters link a during interval k and travels to destination s.

$u_{ab}^{rs}(k)$ inflow that departs from origin r, enters link a during interval k and passes through link $b \in A(h_a)$ to destination s.

$U_a(k)$ cumulative number of vehicles that enter link a by the end of interval k.

$U_a^{rs}(k)$ cumulative number of vehicles that depart from origin r, enter link a by the end of interval k and travels to destination s.

$U_{ab}(k)$ cumulative number of vehicles that enter link a by the end of interval k and pass through link $b \in A(h_a)$.

$U_{ab}^s(k)$ cumulative number of vehicles that enter link a by the end of interval k and pass through link $b \in A(h_a)$ to destination s.

$U_{ab}^{rs}(k)$ cumulative number of vehicles that depart from origin r, enter link a by the end of interval k and pass through link $b \in A(h_a)$ to destination s.

$v_a(k)$ outflow from link a during interval k.

$v_a^s(k)$ outflow from link a during interval k to destination s.

$v_{ab}^s(k)$ outflow from link a during interval k and through link $b \in A(h_a)$ to destination s.

$v_a^{rs}(k)$ outflow that departs from origin r, exits link a during interval k and travels to destination s.

$v_{ab}^{rs}(k)$ outflow that departs from origin r, exits link a during interval k and passes through link $b \in A(h_a)$ to destination s.

$V_a(k)$ cumulative number of vehicles that leave link a by the end of interval k.

$V_a^{rs}(k)$ cumulative number of vehicles that departs from origin r, enters link a by the end of interval k and travels to destination s.

$V_{ab}(k)$ cumulative number of vehicles that leave link a and enter link $b \in A(h_a)$ by the end of interval k.

$V_{ab}^s(k)$ cumulative number of vehicles that leave link a and enter link $b \in A(h_a)$ by the end of interval k and travel to destination s.

$V_{ab}^{rs}(k)$ cumulative number of vehicles that depart from origin r, leave link a and enter link $b \in A(h_a)$ by the end of interval k and travel to destination s.

$g_i^{rs}(k)$ flow from origin r to destination s arriving at node i during interval k.

$\tau_a(k)$ average link travel time for travelers entering link a during interval k.

$\alpha_a^{rs}(k)$ link choice probability of travelers who depart from origin r, enter link a during interval k and travel to destination s.

$\alpha_{ab}^{rs}(k)$ intersection movement choice probability of travelers who depart from origin r, enter link a during interval k and pass through link $b \in A(h_a)$ to destination s.

$\alpha_{ab}^s(k)$ intersection movement choice probability of travelers who enter link a during interval k and pass through link $b \in A(h_a)$ to destination s.

$f_p^{rs}(k)$ inflow that departs from origin r, enters path p during interval k and travels to destination s.

$\eta_p^{rs}(k)$ actual travel time for travelers who enter the network during interval k and travel from r to s through path p.

$\hat{\eta}_p^{rs}(k)$ perceived actual travel time for travelers who enter the network during interval k and travel from r to s through path p.

$P_p^{rs}(k)$ probability of travelers entering path p between r and s during interval k.

$L_a^s(k)$ likelihood of travelers who enter link a during interval k and travel to destination s.

$L_{ab}^s(k)$ likelihood of travelers who enter link a during interval k and pass through link $b \in A(h_a)$ to destination s.

$L_a^{rs}(k)$ likelihood of travelers who depart from origin r, enter link a during interval k and travel to destination s.

$L_{ab}^{rs}(k)$ likelihood of travelers who depart from origin r, enter link a during interval k and pass through link $b \in A(h_a)$ to destination s.

$W_a^s(k)$ weight for travelers who enter link a during interval k and travel to destination s.

$W_{ab}^s(k)$ weight for travelers who enter link a during interval k and pass through link $b \in A(h_a)$ to destination s.

$W_a^{rs}(k)$ weight for travelers who depart from origin r, enter link a during interval k and travel to destination s.

$W_{ab}^{rs}(k)$ weight for travelers who depart from origin r, enter link a during interval k and pass through link $b \in A(h_a)$ to destination s.

$\pi^{is}(k)$ minimum travel time for travelers departing from node i during interval k to destination s

$\pi_a^{is}(k)$ minimum travel time for travelers from node i to destination s entering link $a \in A(i)$ during interval k.

$\pi_{ab}^s(k)$ minimum travel time for travelers entering link a during interval k and traveling from node l_a (i.e., the tail node of link a) to destination s through link $b \in A(h_a)$.

$\boldsymbol{\pi}$ vector of minimum travel times to destinations via specific links $\boldsymbol{\pi} = [\pi_a^{is}(t), \pi_{ab}^s(t)]$.

By definition, the minimum travel times $\pi_a^{is}(k)$ and $\pi_{ab}^s(k)$ can be calculated as follows:

$$\pi_a^{is}(k) = \tau_a(k) + \pi^{h_a s}(k + \tau_a(k)), \forall i \in N, s \in S, a \in A(i), k \in K \text{ , and} \tag{1}$$

$$\pi_{ab}^s(k) = \tau_a(k) + \pi_b^{h_a s}(k + \tau_a(k)), \forall s \in S, a \in A, b \in A(h_a), k \in K \text{ ,} \tag{2}$$

where $k + \tau_a(k)$ may not always be an integer, and a linear interpolation of minimum travel times is adopted to handle the non-integer cases. (Similarly, linear interpolation will also be applied to other variables and functions with the argument $k + \tau_a(k)$ throughout this paper.) Eq. (1) states that the minimum travel time from origin node i to destination s via link a is equal to the sum of travel time over link a and the minimum travel time from the head node of link a to destination s. Eq. (2) states that the minimum travel time from the tail node of link a to destination s via link b is equal to the sum of travel time over link a and the minimum travel time from the head node of link a to destination s via link b.

2.2. Stochastic dynamic user optimal route choice problem

In the SDUO route choice problem, travelers' departure times are assumed to be fixed and hence travelers' departure time choice behavior is not be modeled. Moreover, it is assumed that actual route travel times are perceived differently by travelers due to variations in their perception and exogenous factors, and for each departure time interval and each OD pair, each of the perceived route travel times is assumed to be a random variable (Ran and Boyce, 1996). The random variable is equal to the sum of two components: a systematic term and an error term:

$$\hat{\eta}_p^{rs}(k) = \eta_p^{rs}(k) + \varepsilon_p^{rs}(k), \tag{3}$$

where $\varepsilon_p^{rs}(k)$ is a random error term, and represents the perception error of the travel time over route p between OD pair (r, s) for travelers departing during interval k. It should be pointed out that the randomness of perceived travel times also depends on modelling approximation, such as zoning, as well as modeling errors, such as missing attributes.

With the definition of the perceived route travel time for each departure time interval, the static SUE principle stated by Daganzo and Sheffi (1977) can be directly extended to describe the optimality condition of the SDUO problem in the discrete time setting, which can be expressed as follows (Ran and Boyce, 1996):

If, for each OD pair during each time interval, the travel times perceived by travelers departing during the same time interval are equal and minimal, then the dynamic traffic flow over the network is in a travel-time-based SDUO state.

Mathematically, this condition can be formulated as follows:

$$f_p^{rs}(k) = q^{rs}(k)P_p^{rs}(k), \forall r \in R, s \in S, p \in K_{rs}, k \in K_D, \tag{4}$$

where

$$P_p^{rs}(k) = \Pr\{\hat{\eta}_p^{rs}(k) \le \hat{\eta}_h^{rs}(k), \forall h \in K_{rs}\}, \forall r \in R, s \in S, p \in K_{rs}, k \in K_D. \tag{5}$$

Equation (5) defines the route choice probability of travelers entering path p between OD pair (r, s) during interval k as the probabilities that the perceived travel time over path p between OD pair (r, s) when they depart from r during interval k is not greater than the perceived travel time over other paths between (r, s). This equation implies that the sum of route choice probabilities of all routes connecting the same OD pair for travelers departing during the same time interval equals one. Moreover, by definition, the route choice probabilities and demands are non-negative. Hence, Eq. (4) implies

$$f_p^{rs}(k) \ge 0, \forall r \in R, s \in S, p \in K_{rs}, k \in K_D, \text{ and} \tag{6}$$

$$\sum_p f_p^{rs}(k) = q^{rs}(k), \forall r \in R, s \in S, k \in K_D. \tag{7}$$

If, for each OD pair and departure time interval, the perception errors of route travel times follow the identical and independent Gumbel distributions with mean zero and identical standard deviation, then Eq. (4) immediately becomes a multinomial logit-based dynamic stochastic route choice model, and the route choice probabilities can be formulated as follows:

$$P_p^{rs}(k) = \frac{\exp[-\theta\eta_p^{rs}(k)]}{\sum_{p'}\exp[-\theta\eta_{p'}^{rs}(k)]}, \forall r \in R, s \in S, p \in K_{rs}, k \in K_D, \tag{8}$$

where θ is a positive dispersion parameter that reflects an aggregate measure of drivers' perception of travel times (Sheffi, 1985). A higher value of θ indicates that drivers have more accurate perception of travel times or higher information quality on the traffic network condition (Lo and Szeto, 2004).

2.3 Efficient path set

In general, path choice set generation is required for solving both the static SUE and SDUO problems. Several algorithms (e.g., Dial's (1971) STOCH algorithm and k-shortest path algorithms) can be applied for this purpose. Since the STOCH algorithm and its dynamic extension (i.e., the DYNASTOCH algorithm) do not need to enumerate the path set and allows performing a logit-based stochastic network loading efficiently, they are popularly used for solving stochastic traffic assignment. For example, Huang (1995) adopted the STOCH algorithm to generate the path set for solving static SUE models. Ran and Boyce (1996) extended Dial's STOCH algorithm and developed the DYNASTOCH algorithm to address the SDUO route choice problem. The DYNASTOCH algorithm was later used by Han (2003). Lim and Heydecker (2005) used the STOCH algorithm to generate the path set for solving the SDUO problem with simultaneous departure time and route choices.

Both the STOCH and DYNASTOCH algorithms rely on the concept of efficient or reasonable paths to generate a path choice set for the stochastic traffic assignment problems. In the literature, various definitions on efficient or reasonable paths were given. Dial's (1971) definition on an efficient path can be stated as follows:

Definition D1. *A path between an OD pair is reasonable if it includes only links that take travelers further away from the origin and closer to the destination.*

Ran and Boyce's (1996) definition on an efficient path can be stated as follows:

Definition D2. *A path between an OD pair is reasonable if it includes only links that take travelers closer to the destination.*

Based on these two definitions, we have the following.

Proposition 1. The path set under Definition D1 is a subset of the path set under Definition D2.

The proof is given in Appendix A.1.

These two reasonable path definitions depend on a measure of travel cost associated with or travel distance of each link, such as free-flow travel time. Although the measure can be defined as a function of time, the measure should be constant so that the set of reasonable paths remain unchanged from iteration to iteration during solving SDUO assignment and can be fixed in advance before solving SDUO assignment; otherwise, changes in this path set from iteration to iteration happen and cause the flow patterns to fluctuate over iterations, and no convergent solution can be produced eventually when solving SDUO assignment (Han, 2003; Lim and Heydecker, 2005). Hence, the efficient path concepts in this paper are presented for static networks and are based on free flow travel time (e.g., Han, 2003). Moreover, the reasonable path set is not a function of departure time.

All reasonable paths between an OD pair form a path set for that OD and also a sub-network. Let $G^{rs} = (N^{rs}, A^{rs})$ be the sub-network with respect to OD pair (r, s), where N^{rs} and A^{rs} are the sets of nodes and links in the sub-network G^{rs}, respectively. Then, we have the following proposition.

Proposition 2. (Acyclicity of sub-network). Under Definition D1 or D2, the sub-network G^{rs} is a directed acyclic graph.

The proof is given in Appendix A.2.

Definition D3. (Topological distance). Let the length of every link in the sub-network G^{rs} be 1. Then, the topological distance of a node is defined as the minimum distance from that node to destination s. In other words, the topological distance of a node in the sub-network G^{rs} is defined as the minimum number of links required to pass through from that node to destination s.

Definition D4. (Ascending (descending) pass). An ascending (descending) pass is a sequential visit to each node of a sub-network G^{rs} following the increasing (decreasing) order of topological distance.

Let $A^{rs}(i)$ and $B^{rs}(i)$ be, respectively, the set of links whose tail and head nodes are i in the sub-network G^{rs}. Then, we can prove the following.

Proposition 3. Under Definition D2, for any common node in two sub-networks $G^{r_1 s}$ and $G^{r_2 s}$, where $r_1, r_2 \in R, s \in S$, the topological distance of that node in the sub-network $G^{r_1 s}$ is equal to that in the sub-network $G^{r_2 s}$.

The proof is given in Appendix A.3.

2.4. Dynamic network constraints

Dynamic network constraints are generally used to formulate the feasible domain of DTA problems. Those constraints can be classified into five categories (Ran and Boyce, 1996, Ban et al., 2008): mass balance constraints, first-in-first-out (FIFO) constraints, flow conservation constraints, flow propagation constraints, and definitional constraints. The mass balance constraints define the relationship among link flows (i.e., the number of vehicles on the link), inflow rates, and outflow rates: the rate change in the link flow at each time instant is equal to the difference between the inflow rate and the outflow rate at that time. The FIFO constraints require that vehicles that enter the link earlier leave it sooner. Most of the DUO-based traffic assignment models tend to implicitly guarantee FIFO by using a proper link travel time model which satisfies this property. This is because explicitly imposing this category of constraints may increase the problem complexity and computation burden. In this paper, we do not present the first two categories of constraints, which are implicitly satisfied in the implementation of DNL. The last three categories of constraints are discussed in detail as follows.

The flow propagation constraints represent the consistent evolvement of traffic flows in both temporal and spatial domains. In continuous-time space, they describe the relationship between cumulative link inflow, cumulative link outflow, and time-dependent link travel times:

$$\tilde{V}_a^{rs}(t + \tilde{\tau}_a(t)) = \tilde{U}_a^{rs}(t), \qquad \forall r \in R, s \in S, a \in A^{rs}, t \in T. \tag{9}$$

where $\tilde{U}_a^{rs}(t)$ ($\tilde{V}_a^{rs}(t)$) is the cumulative number of vehicles that depart from origin r, enter (leave) link a by the time t, and travels to destination s, and $\tilde{\tau}_a(t)$ is travel time for travelers entering link a at time t. Following Ban et al. (2008), we use an "inverse link travel time function" $\tilde{p}_a(t) = \tilde{\tau}_a(t - \tilde{p}_a(t))$, where $\tilde{p}_a(t)$ denotes the travel time over link a for vehicles exiting the link at time t. Eq. (9) can be expressed as follows:

$$\tilde{V}_a^{rs}(t) = \tilde{U}_a^{rs}(t - \tilde{p}_a(t)), \qquad \forall r \in R, s \in S, a \in A^{rs}, k \in K . \tag{10}$$

When time is discretized, $t = k\delta$. Then, it follows

$$U_a^{rs}(k) = \tilde{U}_a^{rs}(k\delta), \text{ and } V_a^{rs}(k) = \tilde{V}_a^{rs}(k\delta), \qquad \forall r \in R, s \in S, a \in A^{rs}, k \in K . \tag{11}$$

By definition, we have

$$u_a^{rs}(k) = U_a^{rs}(k) - U_a^{rs}(k-1), \qquad \forall r \in R, s \in S, a \in A^{rs}, k \in K , \text{ and} \tag{12}$$

$$v_a^{rs}(k) = V_a^{rs}(k) - V_a^{rs}(k-1), \qquad \forall r \in R, s \in S, a \in A^{rs}, k \in K . \tag{13}$$

Definition D5 (Critical inflow interval) (Long et al., 2011). A critical inflow interval with respect to interval k is defined as follows:

$$n_k = \max\{\ell \,|\, U_a(\ell) \le V_a(k), \ell \le k - \tau_a^0\} . \tag{14}$$

where τ_a^0 is the free flow travel time on link a.

With the assumption of a constant flow rate during each time interval, substituting Eqs. (10) and (11) into Eq. (13), and using linear combinations for the cumulative inflow curves, we can obtain a linear combination of link inflows for link outflows, given as follows

$$v_a^{rs}(k) = \sum_{\ell \le k - \tau_a^0} \zeta_{a,k}^\ell u_a^{rs}(\ell), \qquad \forall r \in R, s \in S, a \in A^{rs}, k \in K, \tag{15}$$

where $\zeta_{a,k}^\ell$ is the proportion of inflow that enters link a during interval ℓ and leaves the link during interval k. If $n_k = n_{k-1}$, all vehicles that leave link a during interval k enter the link during the same interval n_k, and hence $\zeta_{a,k}^{n_k} = 1 + [\tilde{p}_a((k-1)\delta) - \tilde{p}_a(k\delta)]/\delta$, and $\zeta_{a,k}^\ell = 0, \forall \ell \ne n_k$; otherwise,

$$\zeta_{a,k}^\ell = \begin{cases} n_{k-1} + 1 - [k - 1 - \tilde{p}_a((k-1)\delta)/\delta], & \text{if } \ell = n_{k-1}, \\ 1, & \text{if } n_{k-1} < \ell < n_k, \\ k - \tilde{p}_a(k\delta)/\delta - n_k, & \text{if } \ell = n_k, \\ 0, & \text{otherwise.} \end{cases}$$

The flow conservation constraints require that all flows entering any node (except the destination node) together with the demand generated at this node must exit from this node, and can be formulated as follows:

$$g_i^{rs}(k) = \sum_{a \in A^{rs}(i)} u_a^{rs}(k) = \begin{cases} \delta_i^r q^{rs}(k) + \sum_{b \in B^{rs}(i)} v_b^{rs}(k), \forall r \in R, s \in S, i \in N^{rs}, k \in K_D; \\ \sum_{b \in B^{rs}(i)} v_b^{rs}(k), \forall r \in R, s \in S, i \in N^{rs}, k \in K \setminus K_D, \end{cases} \tag{16}$$

where $\delta_i^r = 1$ if $i = r$, and $\delta_i^r = 0$ otherwise.

By definition, link inflow must be nonnegative:

$$u_a^{rs}(k) \ge 0, \qquad \forall r \in R, s \in S, a \in A^{rs}, k \in K . \tag{17}$$

2.4. Link-based model

To our best knowledge, almost all models for the SDUO problems are path-based. However, the logit-based SDUO problem can be solved without finding path flows. Hence, we believe that there should be a corresponding link-based model. In this paper, we show that the link-based model for the logit-based SDUO problem can be developed using the theory of the DYNASTOCH algorithm, which is an extension of Dial's (1971) STOCH algorithm for the static SUE problem and was firstly proposed by Ran and Boyce (1996). The DYNASTOCH algorithm assumes that link travel times are

functions of link inflow, and link travel times are used to update link inflow during a dynamic logit-based stochastic network loading process. Therefore, a FP problem for the logit-based SDUO problem can be developed using the theory of the DYNASTOCH algorithm.

According to the DYNASTOCH algorithm of Ran and Boyce (1996), the likelihood of a link to be entered by vehicles during interval k in the sub-network G^{rs} with respect to OD pair (r, s) is expressed as follows:

$$L_a^{rs}(k) = \exp\{\theta[\pi^{l_a s}(k) - \pi_a^{l_a s}(k)]\}, \forall r \in R, s \in S, a \in A^{rs}, k \in K . \tag{18}$$

A backward pass method is used to calculate the weight of each link:

$$W_a^{rs}(k) = L_a^{rs}(k)\left\{\delta_{h_a}^s + \sum_{b \in A^{rs}(h_a)} W_b^{rs}(k + \tau_a(k))\right\}, \forall r \in R, s \in S, a \in A^{rs}, k \in K , \tag{19}$$

where $\delta_{h_a}^s = 1$ if $h_a = s$ and $\delta_{h_a}^s = 0$ otherwise.

The choice probability of a link in the sub-network G^{rs} is given as follows:

$$\alpha_a^{rs}(k) = \frac{W_a^{rs}(k)}{\sum_{b \in A^{rs}(l_a)} W_b^{rs}(k)}, \forall r \in R, s \in S, a \in A^{rs}, k \in K . \tag{20}$$

Note that the likelihoods, the weights and the choice probabilities of all unused links are equal to zero. That is, $L_a^{rs}(k) = 0$, $W_a^{rs}(k) = 0$, and $\alpha_a^{rs}(k) = 0$ are satisfied for all $a \in A \setminus A^{rs}$ and $\forall r, s, k$.

Proposition 4. Under Definition D2, $\alpha_a^{r_1 s}(k) = \alpha_a^{r_2 s}(k)$ is satisfied for all $a \in A^{r_1 s} \bigcap A^{r_2 s}$ and $\forall r_1, r_2, s, k$.

The proof is given in Appendix A.4.

The link inflow into a link equals the choice probability of this link multiplied by the total flow arriving at its entrance node, given by

$$u_a^{rs}(k) = \begin{cases} \alpha_a^{rs}(k)[q^{rs}(k)\delta_{l_a}^r + \sum_{b \in B^{rs}(l_a)} \sum_{\ell \leq k - \tau_b^0} \zeta_{b,k}^\ell u_b^{rs}(\ell)], \forall r \in R, s \in S, a \in A^{rs}, k \in K_D, \\ \alpha_a^{rs}(k) \sum_{b \in B^{rs}(l_a)} \sum_{\ell \leq k - \tau_b^0} \zeta_{b,k}^\ell u_b^{rs}(\ell), \forall r \in R, s \in S, a \in A^{rs}, k \in K \setminus K_D. \end{cases} \tag{21}$$

Substituting Eq. (15) into Eq. (21), we have

$$u_a^{rs}(k) = \begin{cases} \alpha_a^{rs}(k)\left\{q^{rs}(k)\delta_{l_a}^r + \sum_{b \in B^{rs}(l_a)} \sum_{\ell \leq k - \tau_b^0} \zeta_{b,k}^\ell u_b^{rs}(\ell)\right\}, \forall r \in R, s \in S, a \in A^{rs}, k \in K_D, \\ \alpha_a^{rs}(k) \sum_{b \in B^{rs}(l_a)} \sum_{\ell \leq k - \tau_b^0} \zeta_{b,k}^\ell u_b^{rs}(\ell), \forall r \in R, s \in S, a \in A^{rs}, k \in K \setminus K_D. \end{cases} \tag{22}$$

Link choice probabilities are functions of link travel times, which in turns are functions of link inflows. Hence, link choice probabilities are also functions of link inflows. Therefore, the link-based SDUO problem can be formulated as a FP problem: to find $\mathbf{u} = [\alpha_a^{rs}(k)g_{l_a}^{rs}(k)]$ such that

$$\mathbf{u} = \mathbf{f(u)}, \tag{23}$$

where $\alpha_a^{rs}(k)$ is defined by Eqs. (18)-(20), and $g_{l_a}^{rs}(k)$ is defined by Eq. (16).

2.5. Intersection-movement-based model

2.5.1. Model I

The output of the link-based model for the SDUO problem is the vector of link inflow $\mathbf{u} = [u_a^{rs}(k)]$, which do not contain any path information. Hence, it cannot be directly used as an input to the DNL

model with physical queue consideration. One method to overcome this problem is to use intersection movement choice probabilities instead of link inflows to formulate the problem.

For a node that is neither an origin nor a destination in G^{rs}, there are at least one incoming link into the node and one outgoing link from it. Then, an intersection movement can be described by a pair of incoming and outgoing links. For example, the intersection movement for flows coming from l_b at node h_a in Fig. 1 can be described by two links a and b. The intersection movement represents flows making through and turning movements at intersections.

For an origin, the movement of travelers entering into the network is also described by two numbers, namely the node and link numbers. A traveler's departure from an origin can also be viewed as an intersection movement, as the origin can be considered as an intersection and the traveler can select links for entering the network. For instance, if h_a is also an origin for the concerned OD pair, then the intersection movement at h_a is described by the tail node of the outgoing link (i.e., the origin node itself) and the outgoing link b.

An intersection movement implicitly includes the path related information, i.e., how many vehicles on a link selecting a turning or though movement at the next intersection or how many vehicles departing from an origin selecting a particular downstream link to enter the network. Indeed, each intersection movement includes sub-path information where a subpath is formed by two adjacent links or by an origin node and an outgoing link from that node. Each intersection movement describes how traffic moves either from one link to another or from an origin node to a link. Hence, by checking intersection movements from one intersection to the other, a path can be traced from an origin to a destination. To illustrate this further, we consider a path from l_a to h_b in Fig. 1. Traditionally, it can be described by either a sequence of nodes ($l_a \rightarrow h_a \rightarrow h_b$) or a sequence of links ($a \rightarrow b$). Alternatively, the path can also be viewed as a sequence of intersection movements ($l_a \rightarrow a$, $a \rightarrow b$). The path can be deduced by checking the intersection movements at node l_a and h_b.

As path information is implicitly included in this formulation approach, queue spillback can be modeled if a physical-queue traffic model is adopted. Modeling queue spillback in this formulation only relies on defining the immediate downstream link for the flow on each link. Flows cannot enter the downstream link if there is insufficient space and, instead, they form a queue on the upstream link. However, as a path set is not required to be known, path enumeration or column generation can be avoided during the solution process.

To incorporate the concept of intersection movements into a formulation, we need to disaggregate link flows by intersection movements. As shown in Fig. 1, the inflow into link a during interval k is $u_a^{rs}(k)$, which is disaggregated by intersection movements at the next intersection h_a. Hence, by definition, we have

$$u_a^{rs}(k) = \sum_{b \in A^{rs}(h_a)} u_{ab}^{rs}(k), \forall r \in R, s \in S, a \in A^{rs}, k \in K. \tag{24}$$

Note that k is the time interval for flows entering link a, not the time interval passing through the intersection h_a or leaving link a.

The disaggregated link flows should satisfy the following three categories of constraints:

The flow propagation constraints:
$$v_{ab}^{rs}(k) = \sum_{\ell \le k-\tau_a^0} \zeta_{a,k}^{\ell} u_{ab}^{rs}(\ell), \forall r \in R, s \in S, a \in A^{rs}, b \in A^{rs}(h_a), k \in K. \tag{25}$$

The flow conservation constraints:
$$\sum_{b \in A(h_a)} u_{ab}^{rs}(k) = u_a^{rs}(k)\delta_{l_a}^r + \sum_{b \in B(l_a)} \sum_{\ell \le k-\tau_b^0} \zeta_{b,k}^{\ell} u_{ba}^{rs}(\ell), \forall r \in R, s \in S, a \in A^{rs}, k \in K. \tag{26}$$

The nonnegativity constraints:
$$u_{ab}^{rs}(k) \ge 0, \quad \forall r \in R, s \in S, a \in A^{rs}, b \in A^{rs}(h_a), k \in K. \tag{27}$$

Similar to the DYNASTOCH algorithm of Ran and Boyce (1996), the likelihood of an intersection movement in the sub-network G^{rs} can be expressed as follows:
$$L_{ab}^{rs}(k) = \exp\{\theta[\pi_a^s(k) - \pi_{ab}^s(k)]\}, \forall r \in R, s \in S, a \in A^{rs}, b \in A^{rs}(h_a), k \in K, \text{ and} \tag{28}$$
$$L_a^{rs}(k) = \exp\{\theta[\pi^{l_a s}(k) - \pi_a^{l_a s}(k)]\}, \forall r \in R, s \in S, a \in A^{rs}(r), k \in K_D. \tag{29}$$

Note that Eq. (29) is different from Eq. (18); the set to define link a is different.

A backward pass method is used to calculate the weight of each intersection movement:
$$W_{ab}^{rs}(k) = L_{ab}^{rs}(k)\left\{\delta_{h_b}^s + \sum_{c \in A^{rs}(h_b)} W_{bc}^{rs}(k + \tau_a(k))\right\}, \forall r \in R, s \in S, a \in A^{rs}, b \in A^{rs}(h_a), k \in K, \text{ and} \tag{30}$$
$$W_a^{rs}(k) = L_a^{rs}(k)\left\{\delta_{h_a}^s + \sum_{b \in A^{rs}(h_a)} W_{ab}^{is}(k)\right\}, \forall r \in R, s \in S, a \in A^{rs}(r), k \in K_D. \tag{31}$$

in the sub-network G^{rs} is given as follows:
$$\alpha_{ab}^{rs}(k) = \frac{W_{ab}^{rs}(k)}{\sum\limits_{b' \in A^{rs}(h_a)} W_{ab'}^{rs}(k)}, \forall r \in R, s \in S, a \in A^{rs}, b \in A^{rs}(h_a), k \in K, \text{ and} \tag{32}$$
$$\alpha_a^{rs}(k) = \frac{W_a^{rs}(k)}{\sum\limits_{a' \in A^{rs}(r)} W_{a'}^{rs}(k)}, \forall r \in R, s \in S, a \in A^{rs}(r), k \in K_D. \tag{33}$$

If $\boldsymbol{\alpha} = [\alpha_a^{rs}(k), \alpha_{ab}^{rs}(k)]$ is known, then the intersection movements $\mathbf{u} = [u_a^{rs}(k), u_{ab}^{rs}(k)]$ can be calculated by
$$u_a^{rs}(k) = \alpha_a^{rs}(k)q^{rs}(k), \forall r \in R, s \in S, a \in A^{rs}(r), k \in K_D, \text{ and} \tag{34}$$
$$u_{ab}^{rs}(k) = \begin{cases} \alpha_{ab}^{rs}(k)\left[\alpha_a^{rs}(k)q^{rs}(k)\delta_{l_a}^r + \sum\limits_{b' \in B^{rs}(l_a)} \sum\limits_{\ell \le k-\tau_{b'}^0} \zeta_{b',k}^{\ell} u_{b'a}^{rs}(\ell)\right], \forall r \in R, s \in S, a \in A^{rs}, b \in A^{rs}(h_a), k \in K_D, \\ \alpha_{ab}^{rs}(k) \sum\limits_{b' \in B^{rs}(l_a)} \sum\limits_{\ell \le k-\tau_{b'}^0} \zeta_{b',k}^{\ell} u_{b'a}^{rs}(\ell), \forall r \in R, s \in S, a \in A^{rs}, b \in A^{rs}(h_a), k \in K \setminus K_D. \end{cases} \tag{35}$$

Definition D6. (Causality of traffic flow). The link travel times for traffic entering during interval k depend on traffic entering during or earlier than interval k, but not on traffic entering later than k.

Causality is an actual traffic behavior. It means that the speed and travel time of a vehicle on a link is only affected by the speed of vehicles ahead. It may or may not be captured by traffic flow models. According to Definition D6, if the traffic flow modeled by traffic flow model satisfies causality, the link travel time $\tau_a(k)$ depends on the traffic entering during interval k or earlier than k, but not on traffic entering later than k.

Proposition 5. If the network is empty initially and the network traffic modeled satisfies causality, then the intersection movement vector $\mathbf{u} = [u_a^{rs}(k), u_{ab}^{rs}(k)]$ which satisfies constraints (25)-(27) is a function of the intersection movement choice probability vector $\mathbf{\alpha} = [\alpha_a^{rs}(k), \alpha_{ab}^{rs}(k)]$, and the function $\mathbf{u}(\mathbf{\alpha})$ is surjection. In other words, $\mathbf{u}$ can be unilaterally determined by the intersection movement choice probability vector $\mathbf{\alpha}$.

The proof is given in Appendix A.5.

Based on the preceding discussion, we can have the following:

● Intersection movements are functions of intersection movement choice probabilities according to Proposition 5,

● Link inflows are functions of intersection movements as shown in Eq. (24),

● Link travel times are functions of link inflows according to the assumption of the DYNASTOCH algorithm,

● Minimum travel times to destinations are functions of link travel times according to Eqs. (1) and (2), and

● Intersection movement choice probabilities are functions of minimum travel times to destinations according to Eqs. (28)-(33).

Therefore, intersection movement choice probabilities are functions of themselves, and the intersection-movement-based SDUO problem can be reformulated as a FP in terms of the vector of intersection movement choice probabilities: to find $\mathbf{\alpha} = [\alpha_a^{rs}(k), \alpha_{ab}^{rs}(k)]$

$$\mathbf{\alpha} = \mathbf{h}(\mathbf{\alpha}) \, , \tag{36}$$

where $\mathbf{h}(\mathbf{\alpha})$ is defined by Eqs. (28)-(33).

Long et al. (2013a) proved that discretised link travel times based on cumulative curves, which will be used for our models, converge to the corresponding continuous-time link travel times if the latter is continuous with respect to time instant t. Hence, we have the following assumption for general link travel time functions:

Assumption A1. The link travel time $\tilde{\tau}_a(t)$ is continuous with respect to time instant t, and link travel time $\tau_a(k)$ converges to link travel times $\tilde{\tau}_a(k\delta)$ when the interval length δ approaches to zero.

Proposition 6. Under Assumption A1, a solution of the intersection-movement-based model (36) is an optimal solution of the logit-based SDUO route choice problem when the interval length approaches to zero.

The proof is given in Appendix A.6.

Since a route travel choice probability is the multiplication of choice probabilities of intersection movements on the route, we can retrieve route travel choice probabilities from intersection movement choice probabilities, given as follows:

$$\hat{P}_p^{rs}(k) = \alpha_{a_1}^{rs}(k) \prod_{i=1}^{m_l-1} \alpha_{a_i a_{i+1}}^{rs}(t_{a_i} / \delta) \, , \tag{37}$$

where route p of OD pair (r, s) consists of links $\{a_1, a_2, \cdots, a_{m_l}\}$, and m_l is the number of links on that route, t_{a_i} is the arrival time at node l_{a_i}. According to Proposition 6, the route travel choice

probabilities retrieved by Eq. (37) is equal to the logit-based SDUO route choice probabilities in Eq. (8) when the interval length approaches to zero. In general, time discretization introduces errors for the retrieved route travel choice probabilities.

Proposition 7. Link choice probabilities and intersection movement choice probabilities satisfy the following relationship: $\alpha_{ab}^{rs}(k) = \alpha_b^{rs}(k + \tau_a(k)), \forall r \in R, s \in S, a \in A^{rs}, b \in A^{rs}(h_a), k \in K$.

The proof is given in Appendix A.7.

Proposition 8. Under Assumption A1, if $\boldsymbol{\alpha}^* = [\alpha_a^{rs*}(k), \alpha_{ab}^{rs*}(k)]$ is a solution of the FP problem (36), then the corresponding link inflow vector $[u_a^{rs*}(k)]$ is also a solution of the FP problem (23) when the interval length approaches to zero.

The proof is given in Appendix A.8.

Definition D7 (Common intersection movement). If $a, b \in A^{r_1s} \bigcap A^{r_2s}$, then the intersection movement (a, b) is defined as a common intersection movement with respect to the sub-networks G^{r_1s} and G^{r_2s} .

Proposition 9. Under Definition D2, all the common intersection choice probabilities of travelers who depart from any origins, enter a link during the same time interval and pass through the same next link to the identical destination are equal. Equivalently, $\alpha_{ab}^{r_1s}(k) = \alpha_{ab}^{r_2s}(k)$ is satisfied for all $a, b \in A^{r_1s} \bigcap A^{r_2s}$ and $\forall r_1 \in R, r_2 \in R, s \in S, k \in K$ under Definition D2.

The proof is given in Appendix A.9.

2.5.2. Model II

According to Proposition 9, for each common intersection movement during each time interval, the choice probabilities of the common intersection movement are irrespective of origin under Definition D2. Therefore, the SDUO problem can be formulated in terms of destination-based variables only. In other words, we distinguish flows according to their destinations instead of OD pairs. According to Definition D2, all used links for each destination form a sub-network. Let $G^s = (N^s, A^s)$ be the sub-network with respect to destination s, where N^s and A^s are the sets of nodes and links in the sub-network G^s, respectively. $A^s(i)$ and $B^s(i)$ are, respectively, the sets of links whose tail and head nodes are i in the sub-network G^s. Then, we have $N^s = \bigcup_r N^{rs}$ and $A^s = \bigcup_r A^{rs}$.

Proposition 10. The sub-network G^s is a directed acyclic graph.

The proof is similar to that of Proposition 1 and hence is omitted.

Proposition 11. Under Definition D2, $A^{rs}(i) = A^s(i)$ if $i \in N^{rs}$.

The proof is given in Appendix A.10.

Similar to Model I, if the SDUO problem is formulated in terms of intersection movements, then the corresponding model should satisfy the following three categories of constraints:

The flow propagation constraints:
$$v_{ab}^s(k) = \sum_{\ell \le k - \tau_a^0} \lambda_{a,k,\ell} u_{ab}^s(\ell), \forall s \in S, a \in A^s, b \in A^s(h_a), k \in K. \tag{38}$$

The flow conservation constraints:

$$\sum_{b \in A^s(h_a)} u_{ab}^s(k) = \begin{cases} \sum_{r \in R} u_a^{rs}(k)\delta_{l_a}^r + \sum_{b \in B^s(l_a)} \sum_{\ell \le k - \tau_b^0} \zeta_{b,k}^\ell u_{ba}^s(\ell), \forall s \in S, a \in A^s, k \in K_D, \\ \sum_{b \in B^s(l_a)} \sum_{\ell \le k - \tau_b^0} \zeta_{b,k}^\ell u_{ba}^s(\ell), \forall s \in S, a \in A^s, k \in K \setminus K_D. \end{cases} \tag{39}$$

The nonnegativity constraints:

$$u_{ab}^s(k) \ge 0, \qquad \forall s \in S, \ a \in A^s, \ b \in A(h_a), k \in K. \tag{40}$$

In Model I, the calculation of likelihoods, weights, and intersection movement choice probabilities with respect to each OD pair are independent of origins and can be directly used to calculate likelihoods, weights, and choice probabilities with respect to each destination. Herein, we directly give the formulae as follows:

$$L_{ab}^s(k) = \exp\{\theta[\pi_a^s(k) - \pi_{ab}^s(k)]\}, \forall s \in S, a \in A^s, b \in A^s(h_a), k \in K, \tag{41}$$

$$L_a^s(k) = \exp\{\theta[\pi^{l_a s}(k) - \pi_a^{l_a s}(k)]\}, \forall r \in R, s \in S, a \in A^s(r), k \in K_D, \tag{42}$$

$$W_{ab}^s(k) = L_{ab}^s(k)\left\{\delta_{h_b}^s + \sum_{c \in A^s(h_b)} W_{bc}^s(k + \tau_a(k))\right\}, \forall s \in S, a \in A^s, b \in A^s(h_a), k \in K, \tag{43}$$

$$W_a^s(k) = L_a^s(k)\left\{\delta_{h_a}^s + \sum_{b \in A^s(h_a)} W_{ab}^s(k)\right\}, \forall r \in R, s \in S, a \in A^s(r), k \in K_D, \tag{44}$$

$$\alpha_{ab}^s(k) = \frac{W_{ab}^s(k)}{\sum_{b' \in A^s(h_a)} W_{ab'}^s(k)}, \forall s \in S, a \in A^s, b \in A^s(h_a), k \in K, \text{ and} \tag{45}$$

$$\alpha_a^{rs}(k) = \frac{W_a^s(k)}{\sum_{a' \in A^s(r)} W_{a'}^s(k)}, \forall r \in R, s \in S, a \in A^s(r), k \in K_D, \tag{46}$$

where Eq. (46) defines the choice probability of the movements corresponding to flows departing from origin r during time interval k to destination s. Note that in Eq. (46), $a \in A^s(r)$, not $a \in A^{rs}(r)$ as in Eq. (33).

Given $\bar{\boldsymbol{\alpha}} = [\alpha_a^{rs}(k), \alpha_{ab}^s(k)]$, the inflows $u_a^{rs}(k)$ and $u_{ab}^s(k)$ can be determined by:

$$u_a^{rs}(k) = \alpha_a^{rs}(k)q^{rs}(k), \forall r \in R, s \in S, a \in A^s(r), k \in K_D, \text{ and} \tag{47}$$

$$u_{ab}^s(k) = \begin{cases} \alpha_{ab}^s(k)\left[\sum_{r \in R} \alpha_a^{rs}(k)q^{rs}(k)\delta_{l_a}^r + \sum_{b' \in B^s(l_a)} \sum_{\ell \le k - \tau_{b'}^0} \zeta_{b',k}^\ell u_{b'a}^s(\ell)\right], \forall s \in S, a \in A^s, b \in A^s(h_a), k \in K_D, \\ \alpha_{ab}^s(k) \sum_{b' \in B^{rs}(l_a)} \sum_{\ell \le k - \tau_{b'}^0} \zeta_{b',k}^\ell u_{b'a}^s(\ell), \forall s \in S, a \in A^s, b \in A^s(h_a), k \in K \setminus K_D. \end{cases} \tag{48}$$

Proposition 12. If the network is empty initially and the network traffic modeled satisfies causality, then the intersection movement vector $\bar{\mathbf{u}} = [u_a^{rs}(k), u_{ab}^s(k)]$ which satisfies constraints (38)-(40) is a function of the intersection movement choice probability vector $\bar{\boldsymbol{\alpha}} = [\alpha_a^{rs}(k), \alpha_{ab}^s(k)]$, and the function $\bar{\mathbf{u}}(\bar{\boldsymbol{\alpha}})$ is surjection. In other words, $\bar{\mathbf{u}}$ can be unilaterally determined by the intersection movement choice probability vector $\bar{\boldsymbol{\alpha}}$.

The proof is similar to that of Proposition 5.

Similar to Model I, the intersection-movement-based SDUO problem can be alternatively formulated in terms of the vector of destination-based intersection movement choice probabilities, given as follows:

$$\bar{\boldsymbol{\alpha}} = \bar{\mathbf{h}}(\bar{\boldsymbol{\alpha}}) \,, \tag{49}$$

where $\bar{\boldsymbol{\alpha}} = [\alpha_a^{rs}(k), \alpha_{ab}^s(k)]$ is the intersection movement choice probability vector, and $\bar{\mathbf{h}}(\bar{\boldsymbol{\alpha}})$ is defined by Eqs. (41)-(46).

Definition D8 (Transfer function). A transfer function from the vector of destination-based intersection movement choice probabilities to the vector of OD-based intersection movement choice probabilities is defined as follows:

$$\boldsymbol{\alpha} = \mathbf{y}(\bar{\boldsymbol{\alpha}}) \,, \tag{50}$$

where $\boldsymbol{\alpha} = [\alpha_a^{rs}(k), \alpha_{ab}^{rs}(k)]$, $\bar{\boldsymbol{\alpha}} = [\alpha_a^{rs}(k), \alpha_{ab}^s(k)]$, and $\alpha_{ab}^{rs}(k) = \alpha_{ab}^s(k), \forall r \in R, s \in S, a \in A^{rs}, b \in A^{rs}(h_a)$, $k \in K$. Note that the function is unique or a point-to-point mapping.

Assumption A2. Both $\mathbf{y}(\bar{\boldsymbol{\alpha}})$ and $\bar{\boldsymbol{\alpha}}$ are used as inputs of DNL models and output the same link cumulative flows and link travel times.

Based on A1, D2 and D8, we have Propositions 13-16.

Proposition 13. Under Definition D2 and Assumption A2, we have $\mathbf{h}(\mathbf{y}(\bar{\boldsymbol{\alpha}})) = \mathbf{y}(\bar{\mathbf{h}}(\bar{\boldsymbol{\alpha}}))$.

The proof is given in Appendix A.11.

Proposition 14. Under Definition D2 and Assumption A2, if the vector $\bar{\boldsymbol{\alpha}}^*$ is a solution of the FP problem (49), then $\boldsymbol{\alpha}^* = \mathbf{y}(\bar{\boldsymbol{\alpha}}^*)$ is a solution of the FP problem (36).

Proposition 15. Under Definition D2 and Assumptions A1 and A2, if the vector $\bar{\boldsymbol{\alpha}}^*$ is a solution of the FP problem (49), then the corresponding link inflow vector $[u_a^{rs*}(t)]$ is also a solution of the FP problem (23) when the interval length approaches to zero.

Proposition 16. Under Definition D2 and Assumption A2, we have $\left\| \bar{\mathbf{h}}(\bar{\boldsymbol{\alpha}})\text{-}\bar{\boldsymbol{\alpha}} \right\|_1 \le \left\| \mathbf{h}(\mathbf{y}(\bar{\boldsymbol{\alpha}}))\text{-}\mathbf{y}(\bar{\boldsymbol{\alpha}}) \right\|_1$ and $\left\| \bar{\mathbf{h}}(\bar{\boldsymbol{\alpha}})\text{-}\bar{\boldsymbol{\alpha}} \right\|_\infty = \left\| \mathbf{h}(\mathbf{y}(\bar{\boldsymbol{\alpha}}))\text{-}\mathbf{y}(\bar{\boldsymbol{\alpha}}) \right\|_\infty$.

The proof of propositions 13, 14, and 16 are given in Appendices 11, 12, 13. Proposition 15 follows directly from Propositions 9 and 14.

3. Network traffic flow model and network performance assessment

3.1. The modified link transmission model

To capture queue spillback in the FP problems (36) and (49), we developed a MLTM (see Appendix B for details) as a network flow propagation model and used it in the FP models, although other traffic flow models that can produce a unique mapping from intersection movement choice probabilities to link travel times can be incorporated into these FP models. Different from the original LTM (Yperman, 2007), the MLTM takes the demands and the choice of probabilities of intersection movements as inputs. The MLTM is adopted to implement DNL and generate cumulative link flows, which can be further used to compute link travel times.

3.2. Discretized travel time model

The proposed intersection-movement-based DTA model is formulated in the discrete time setting and thus a discretized link travel time model is required to compute the travel times. In discrete time DTA models, the travel time over a link with respect to a particular time interval can be defined as the

average travel time of vehicles that enter that link during that interval (Lo and Szeto, 2002a). It can be formulated as follows:

$$\tau_a(k) = \frac{\int_{U_a(k-1)}^{U_a(k)} \left[V_a^{-1}(v) - U_a^{-1}(v) \right] dv}{U_a(k) - U_a(k-1)} , \tag{51}$$

where $U_a^{-1}(\cdot)$ and $V_a^{-1}(\cdot)$ are the inverse functions of $U_a(\cdot)$ and $V_a(\cdot)$, respectively. The numerator and the denominator of the right hand side of Eq. (51) are total travel time and the number of vehicles that enter link a during this interval, respectively.

In general, it is not possible to obtain the inverse functions $U_a^{-1}(\cdot)$ and $V_a^{-1}(\cdot)$ in close forms. Therefore, piecewise linear functions, such as the step function (SF) (e.g., Lo and Szeto, 2002a) and linear interpolation (LI) (e.g., Yperman, 2007), are usually applied to approximate the profiles of cumulative flows. With the two types of piecewise linear approximations, Long et al. (2011) developed formulations for SF-type and LI-type link travel time functions, and also proposed a modified LI-type (MLI-type) link travel time model. The three types of travel time functions are proved to satisfy some desirable properties such as FIFO, causality, monotonicity, and continuity. This paper uses the MLI-type link travel time model to calculate the link travel times, since it has better properties (e.g., with a higher precision) than other two functions (see Long et al. 2011 for the discussion of accuracy).

Given link travel times, the travel time required by vehicles departing from their origin during interval k to traverse a path $p = \{a_1, a_2, \cdots, a_m\}$ can be computed by using the following nested function (Ran and Boyce, 1996; Huang and Lam, 2002):

$$\eta_p(k) = \tau_{a_1}(k) + \tau_{a_2}(k + \tau_{a_1}(k)) + \cdots + \tau_{a_m}(k + \tau_{a_1} + \cdots + \tau_{a_{m-1}}) . \tag{52}$$

where $\tau_{a_1} = \tau_{a_1}(k)$, $\tau_{a_2} = \tau_{a_2}(k + \tau_{a_1}(k))$,…, for short.

Assumption A3. The link flow satisfies link FIFO and causality, and the cumulative outflows are continuous with respect to the link inflow vector.

Note that the flow estimated by the MLTM satisfies the first part of A2 but may not satisfy the second part due to queue spillback. Under A2, we have Propositions 17-21.

Proposition 17. Under Assumption A3, link (route) travel times are continuous with respect to the intersection movement choice probability vector $\boldsymbol{\alpha}$ ($\bar{\boldsymbol{\alpha}}$).

Proposition 18. Under Assumption A3, $\mathbf{h}(\boldsymbol{\alpha})$ is a continuous function of the intersection movement choice probability vector $\boldsymbol{\alpha}$.

Proposition 19. Under Definition D2 and Assumption A3, $\bar{\mathbf{h}}(\bar{\boldsymbol{\alpha}})$ is a continuous function of the intersection movement choice probability vector $\bar{\boldsymbol{\alpha}}$.

Proposition 20. Under Assumption A3, the FP problem (36) has at least one solution.

Proposition 21. Under Assumption A3, the FP problem (49) has at least one solution.

The proof of proposition 17 follows directly from Proposition 6 in Long et al. (2013b). The proof of proposition 18 is given in Appendix A.14. The proof of proposition 19 is basically the same as that of Proposition 18. The proof of proposition 20 is given in Appendix A.15. The proof of proposition 21 is similar to that of Proposition 20.

According to Propositions 20 and 21, the proposed intersection-movement-based SDUO problems guarantee solution existence under Assumption A3. However, the link travel times (or route travel times) derived by a physical-queue traffic flow model may not be continuous, leading to the possibility of the non-existence of SDUO solutions (Szeto and Lo, 2006).

3.3. Measures of network performance

3.3.1. Total system travel time

TSTT is a widely used measure of network performance in various problems, such as signal control problems, network design problems, ramp control problems, etc. TSTT is defined as the sum of the travel times of all vehicles from their origins to their destinations. It equals the total link travel times of all vehicles. Since link inflows during each time interval can be derived by the cumulative link inflows, which are outputs of the MLTM, TSTT can be formulated as follows:

$$\text{TSTT} = \sum_{k \in K} \sum_{a \in A} u_a(k)\tau_a(k) = \sum_{k \in K} \sum_{a \in A} [U_a(k) - U_a(k-1)]\tau_a(k). \tag{53}$$

3.3.2. Total cost of vehicle emissions

Following Penic and Upchurch (1992), we assume that vehicles travel over a link with a constant speed, and the emission rate of a link is a function of its grade and the average speed of vehicles. Let M be the set of types of pollutants considered. For all $m \in M$, we adopt the following emission rate function:

$$\phi_m(s, g) = \frac{A_m(s) + [B_m(s) - A_m(s)]h(g)}{C_m s}, \tag{54}$$

where $A_m(s)$ and $B_m(s)$ are functions of the vehicle average speed s, $h(g)$ is a weight function of the grade g of roadway, and C_m is a constant. The functions and the constants are given in Table 1, where s is in ft/s, g is in percent, and $\phi_m(s, g)$ is in grams per vehicle foot.

According to the constant speed assumption, we can use link travel times to calculate average vehicle speeds, given as follows:

$$s_a(t) = \frac{d_a}{\tau_a(t)}, \tag{55}$$

where d_a is the length of link a.

Substituting Eq. (55) in the emission rate function (54), we can obtain the total emissions of the whole network, which can be further transferred into a monetary measure, namely CTVE. CTVE can be calculated by:

$$\text{CTVE} = \sum_{k \in K} \sum_{m \in M} \sum_{a \in A} \eta_m \phi_m(s_a(k), g_a) d_a [U_a(k) - U_a(k-1)], \tag{56}$$

where η_m is the emission costs of unit pollutant m. The emission costs of unit pollutant are given in Table 2.

4. Solution algorithm

4.1. The general scheme

The general scheme of the whole procedure for solving the intersection-movement-based SDUO problem is provided in Fig. 2. In this procedure, the sub-networks for all OD pairs or destinations are generated according to Definition D1 or D2. The network flow propagation model is the MLTM (see Appendix B for details), and intersection movement choice probabilities are inputs of the MLTM when implementing the DNL. The MLTM generates the cumulative inflows into and outflows from each link over time and derive dynamic link travel times. Link travel times are further used to implement stochastic network loading (as mentioned in Subsection 4.2 for details). After stochastic network loading, intersection movement choice probabilities are updated. Finally, if some convergence conditions are satisfied, the whole procedure is terminated; otherwise, the DNL is re-implemented according to current intersection movement choice probabilities, and repeat the consequent steps.

4.2. The algorithm of dynamic stochastic network loading

The algorithm of dynamic stochastic network loading (SNL) in a sub-network is similar to Ran and Boyce's (1996) DYNASTOCH algorithm, but the major differences are their outputs and the ways to define minimum travel times to destinations, the likelihoods, and the weights. The outputs of DYNASTOCH are link flows whereas the outputs of the algorithm herein are intersection movement choice probabilities. Moreover, intersection-movement-based SNL algorithm defines the likelihoods and weights based on the intersection movement concept whereas DYNASTOCH is not. The intersection-movement-based SNL for the intersection-movement-based SDUO problem is outlined as follows:

Step 0: Using Eqs. (1) and (2) to calculate the minimum travel times to destinations according to the current link travel times.

Step 1: Using Eqs. (28) and (29) for Model I or Eqs. (41) and (42) for Model II to calculate intersection movement likelihoods.

Step 2: Using Eqs. (30) and (31) for Model I or Eqs. (43) and (44) for Model II to calculate intersection movement weights in an ascending pass.

Step 3: Using Eqs. (32) and (33) for Model I or Eqs. (45) and (46) for Model II to calculate intersection movement choice probabilities.

The above algorithm can be used under both Definitions D1 and D2. If Definition D2 is adopted, then we only need to distinguish the intersection movements according to their destinations without considering their origins. In this case, the intersection-movement-based SNL algorithm can be implemented in a destination-based manner.

Because time is discretized, the weights of intersection movements cannot be calculated exactly, and we found that there is a small gap between the retrieved route travel choice probabilities obtained by (37) with the probabilities obtained by Eq. (8). In order to improve the accuracy of the weights of intersection movements, we adopt a smaller time step for the intersection-movement-based SNL

algorithm. We divide each time interval into e smaller time intervals. Let ϑ be the length of the small time interval such that $\delta = e\vartheta$. In the intersection-movement-based SNL algorithm, we obtain the minimum travel times to destinations, intersection movement likelihoods, and intersection movement weights at instant $k\delta + j\vartheta$, $\forall k \in K, j \in \{1, 2, \cdots e\}$. Their values at other time instants are estimated by linear interpolation. The effectiveness of this method will be demonstrated in the Section of Numerical Examples.

4.3. The self-regulated averaging method

The path-based SDUO problems can be solved by any general computational techniques developed for VI, such as the diagonalization method (e.g., Ran and Boyce, 1996; Han, 2003), the projection method (e.g., Chen and Feng, 2000; Lo and Szeto, 2004), and so on, provided that the convergent requirements are satisfied. The SDUO problems are usually solved by the techniques developed for FP problems including the method of successive averages (MSA) (e.g., Ran and Boyce, 1996; Han, 2003), which relies on a predetermined step size for guaranteeing convergence. However, the convergence speed of the MSA is slow due to improper step size (too large or too small for some iterations). Liu et al. (2009) proposed a SRAM to deal with the slow convergence problem and to solve the deterministic-network SUE problem. The SRAM was further used by Szeto et al. (2011) to solve their SDUO problem. The SRAM for solving Model I is outlined as follows:

Step 1: *Initialization*. Calculate the vector of initial intersection movement choice probabilities $\boldsymbol{\alpha}^1$ using free-flow travel times. Set $\kappa = 1$, $\tilde{\eta} \geq 1$, $0 < \gamma \leq 1$, $\beta^0 = 1$, and the convergence tolerance $\varepsilon > 0$.

Step 2: *Stochastic loading*. Compute $\mathbf{h}(\boldsymbol{\alpha}^\kappa)$ via the intersection-movement-based SNL algorithm.

Step 3: *Determination of the step size*. Obtain the step size $\tilde{\lambda}^\kappa = 1/\beta^\kappa$, where

$$\beta^\kappa = \begin{cases} \beta^{\kappa-1} + \tilde{\eta}, & \text{if } \left\| \mathbf{h}(\boldsymbol{\alpha}^\kappa)\text{-}\boldsymbol{\alpha}^\kappa \right\| \geq \left\| \mathbf{h}(\boldsymbol{\alpha}^{\kappa-1})\text{-}\boldsymbol{\alpha}^{\kappa-1} \right\|, \\ \beta^{\kappa-1} + \gamma, & \text{otherwise.} \end{cases} \tag{57}$$

Step 4: *Intersection movement choice probability update*. Let $\boldsymbol{\alpha}^{\kappa+1} = \boldsymbol{\alpha}^\kappa + \tilde{\lambda}^\kappa (\mathbf{h}(\boldsymbol{\alpha}^\kappa)\text{-}\boldsymbol{\alpha}^\kappa)$.

Step 5: *Convergence checking*. If $\left\| \mathbf{h}(\boldsymbol{\alpha}^\kappa)\text{-}\boldsymbol{\alpha}^\kappa \right\| \leq \varepsilon$, then stop the algorithm; otherwise, set $\kappa = \kappa + 1$, and go to Step 2.

The above solution algorithm can be directly used to solve Model II by replacing $\boldsymbol{\alpha}$ and $\mathbf{h}(\cdot)$ with $\bar{\boldsymbol{\alpha}}$ and $\bar{\mathbf{h}}(\cdot)$, respectively. Note that if $\tilde{\eta} = \gamma = 1$ is satisfied, then the SRAM immediately becomes the traditional MSA.

Both the Manhattan norm (1-norm) and the maximum norm (∞-norm) can be used in the SRAM. However, the ∞-norm operator is more suitable to evaluate the convergence of the SRAM than the 1-norm operator. The reason is as follows. Assume $\bar{\boldsymbol{\alpha}}$ is a feasible solution of Model II. Then, $\mathbf{y}(\bar{\boldsymbol{\alpha}})$ is a feasible solution of Model I. Under Definition D1, $\bar{\boldsymbol{\alpha}}$ and $\mathbf{y}(\bar{\boldsymbol{\alpha}})$ can be viewed as approximate solutions with the same quality. According to Proposition 16, we have $\left\| \bar{\mathbf{h}}(\bar{\boldsymbol{\alpha}})\text{-}\bar{\boldsymbol{\alpha}} \right\|_1 \leq \left\| \mathbf{h}(\mathbf{y}(\bar{\boldsymbol{\alpha}}))\text{-}\mathbf{y}(\bar{\boldsymbol{\alpha}}) \right\|_1$, which implies that the value of the convergence indicator of the feasible solution $\mathbf{y}(\bar{\boldsymbol{\alpha}})$ for Model I can be overestimated if the 1-norm operator is used. On the contrary, the precision of a solution can be

evaluated in a fair way by using the ∞-norm operator, since we have $\left\|\overline{\mathbf{h}}(\overline{\boldsymbol{\alpha}})\text{-}\overline{\boldsymbol{\alpha}}\right\|_{\infty} = \left\|\mathbf{h}(\mathbf{y}(\overline{\boldsymbol{\alpha}}))\text{-}\mathbf{y}(\overline{\boldsymbol{\alpha}})\right\|_{\infty}$, which confirms that $\overline{\boldsymbol{\alpha}}$ and $\mathbf{y}(\overline{\boldsymbol{\alpha}})$ are with the same quality. Therefore, we adopt the ∞-norm operator to check the convergence of the algorithm in Step 5. Note that the mapping function $\mathbf{h}(\boldsymbol{\alpha})$ may not be monotonic with respect to $\boldsymbol{\alpha}$, and hence the SRAM is a heuristic and may not guarantee convergence.

5 Numerical examples

In this section, six numerical experiments are presented to illustrate the properties of the proposed SDUO problem, the convergence of the solution algorithm, and the effect of information quality and road network improvement on network performance. All experiments were run on a computer with an Intel (R) Core(TM) 2 Quad Q9550 2.83GHz CPU and a 3.5GB RAM.

The OD demands of all examples lasted for 30 intervals with a trapezoidal sharp (see Fig. 3). For a particular OD pair (r, s), its OD demand increased from zero at the beginning of the 1st interval with a constant rate until the end of the 5th interval, maintained the value at a peak of $q_{\max}^{rs}$ (veh/s) until the end of the 15th interval, and then decreased at a constant rate to zero that occurred at the end of the 30th interval. The networks were empty initially. The input parameters of the MLTM for each link in all example networks were the same and are given as follows:

● Jam density: 133 vehicles/km (i.e., 7.5 m for every vehicle).
● Free-flow speed: 54 km/h (i.e., 15 m/s), and backward shock-wave speed: 18 km/h (i.e., 5 m/s).
● Flow capacity: 1800 vehicles/h/lane (i.e., 0.5 veh/s/lane).
● The length of each time interval δ: 10 s.

Example 1: Comparison of the results obtained under different definitions on efficient path sets.

A test network as shown in Fig. 4 was adopted to illustrate the differences in the results obtained under Definitions D1 and D2. This network consisted of 6 nodes, 8 links, and 2 OD pairs (from Nodes 1 and 2 to Node 3). The free-flow travel time (in s) of each link is also given in Fig. 4. The path sets of the two OD pairs under Definitions D1 and D2 are given in Table 3. We can observe that two more paths connect OD pair (1,3) under Definition D2 than under Definition D1. This is because links 2-4 and 6-5 do not take travelers further away from Origin 1, and hence all paths through the two links are not efficient for OD pair (1,3) under Definition D1. On the other hand, since links 2-4 and 6-5 take travelers closer to the Destination 3, the first and the third paths of this OD pair through the two links were efficient for OD pair (1,3) under Definition D2.

All the links were under the free-flow condition during the whole study period. Under this situation, all intersection movements had constant choice probabilities. In Table 4, the choice probabilities of two intersection movements with respect to the two OD pairs are given. We can observe that the choice probabilities of the intersection movement 2-6-3 with respect to the two OD pairs are positive and are not equal to each other under Definition D1. This result implies that all the choice probabilities of an intersection movement with the same destination may not be equal under Definition D1 even if the choice probabilities are positive. On the contrary, the choice probabilities of the intersection movements 2-6-3 and 4-6-5 with respect to the two OD pairs are positive and equal to each other

under Definition D2. This result agrees with Proposition 9. i.e., for each intersection movement during each time interval, the intersection movement choice probabilities of travelers with the same destination are equal under Definition D2 if their probabilities are positive.

Example 2: Performance of the SRAM

To illustrate the performance of the SRAM to solve the proposed intersection-movement-based SDUO models, we developed a numerical example using the Sioux Falls network as shown in Fig. 5. This network consisted of 24 nodes, 76 links, and 528 OD pairs. The travel demand of each OD pair equaled half of that in Leblanc et al. (1975). The free-flow travel time and the number of lanes on each link are given in Table 5. The grades of all links were zero. The values of the parameters for solution algorithms are as follows: $\tilde{\eta}=1.5$, $\gamma=0.01$, $\varepsilon=1.0\times10^{-4}$.

In Figs. 6 and 7, we provided the sub-networks of OD pair (11, 22) and Destination 22. One can observe that all the sub-networks are acyclic. We also can observe that the sub-network of OD pair (11, 22) under Definition D1 is within that of OD pair (11, 22) under Definition D2, and the latter is within that of Destination 22. We set $\theta=1.0$ and applied the SRAM to solve the intersection-movement-based SDUO models. We then determined the gaps between the retrieved route travel choice probabilities obtained by Eq. (37) with the probabilities obtained by Eq. (8), where the mean percentage error (MPE) and the maximum percentage error (MaxPE) are defined as follows:

$$MPE = \frac{\sum_{r\in R}\sum_{s\in S}\sum_{p\in K_{rs}}\sum_{k\in K_D}|P_p^{rs}(k)-\hat{P}_p^{rs}(k)|}{\sum_{r\in R}\sum_{s\in S}\sum_{p\in K_{rs}}\sum_{k\in K_D}P_p^{rs}(k)}\times100\% \text{, and}$$

$$MaxPE = \max_{r\in R, s\in S, p\in K_{rs}, k\in K_D}\left\{\frac{|P_p^{rs}(k)-\hat{P}_p^{rs}(k)|}{P_p^{rs}(k)}\times100\%\right\}.$$

The MPEs against various values of e are graphically shown in Fig. 8. One can observe that the MPE can be effectively reduced by enlarging the value of e. When $e = 5$, the MPE is reduced from 0.023% to 0.0022%, and the MaxPE is reduced from 5.55% to 0.91%. In the following experiments, we set $e = 5$.

Both the 1-norm operator and the ∞-norm operator were used to determine the step sizes in the SRAM. The values of the convergence indicators obtained during the implementation of the algorithm are presented in Fig. 9. We can observe that the value of each convergence indicator under the two efficient path set definitions gradually decreases as the number of iterations grows up and all of them can reach 1.0×10^{-6} when $\theta=1.0$. The results also show that the 1-norm operator leads to a faster convergence than the ∞-norm operator because using the ∞-norm operator can determine a better step size in Eq. (57) (i.e., in Step 3). In addition, we can observe that Model I and Model II under Definition D2 have the same convergence rate when the ∞-norm is used to determine the step size. This result agrees with both Propositions 13 and 16. Since the initial solutions of the two models were obtained under the free-flow condition and were equivalent, Proposition 16 ensures that their step sizes were equal to each other at the current iteration and the algorithm generated equivalent solutions in the

next iteration. Since the 1-norm operator outperformed the ∞-norm operator, only the 1-norm operator was used to determine the step sizes in the SRAM in the following experiments.

To further illustrate the efficiency of the algorithms, we graphically present the required number of iterations and CPU time for solving each model in Fig. 10. From this figure, we can observe that the solution algorithm for Model I under Definition D1 requires the smallest number of iterations on average, while the solution algorithm for Model II under Definition D2 requires the least CPU time on average. The solution algorithm for Model I under Definition D2 required both the largest number of iterations and the most CPU time. The average CPU times for each component and each iteration of the solution algorithm for each model are presented in Table 6. According to this table, most of the CPU times for the solution algorithm were spent on the DNL and the DYNASTOCH algorithm. Compared with the solution algorithm for Model I, the solution algorithm for Model II under Definition D2 was more efficient. The average CPU times spent in each component and in one iteration for Model II under Definition D2 were the smallest among the three cases.

Example 3: Improving the information quality may lead to a worse network performance.

We were also interested in the influence of the value of the dispersion parameter θ on network performance. We first considered the Sioux Falls network. Both TSTT and CTVE are presented in Fig. 11. We can observe that both TSTT and CTVE under Definition D1 are very close to those under Definition D2. The maximum differences in TSTT and CTVE are only 1.35% and 0.84%, respectively. This result implies that both Definitions D1 and D2 can be used in the SDUO model to evaluate network performance without significantly different results obtained. The result presented in Fig. 11 also shows that both Models I and II under Definition D2 lead to the same TSTT and CTVE. This result matches with Proposition 12. The results presented in Fig. 11 also show that CTVE may increase when the value of the dispersion parameter θ grows up. This implies that improving the information quality as represented by the value θ (as in Lo and Szeto, 2004) may lead to a worse network performance in terms of CTVE.

This example also adopted a test network as shown in Fig. 12 with two nodes, two parallel links, and one OD pair (from Node 1 to Node 2). The number of lanes, the outflow capacity (in veh/s), the grade (in percent), and the free-flow travel time (in s) of each link are also given in order inside round bracket next to the link in Fig. 12. We varied the dispersion parameter θ and solved the SDUO problem under different levels of OD demand. The results of TSTT and CTVE are presented in Fig. 13 when the maximum OD demand is 1.0 veh/s and 1.5 veh/s, respectively. As revealed in this figure, both TSTT and CTVE increase as the value of θ grows up. Again, it implies that improving the information quality may lead to a worse network performance. We noted that improved information causing higher TSTT was also previously observed in the static case (e.g., Unnikrishnan and Waller, 2009). However, we confirm this observation in the dynamic case, and also for CTVE.

Example 4: An improvement of road outflow capacity may result in an increase in total vehicle emissions.

The test network used in Example 3 was also adopted in this example. We improved the network by enlarging the outflow capacity of the short link (e.g., by adding more lanes). We can observe from Fig. 14(a) that TSTT can be reduced by enhancing the outflow capacity of the short link. However, this may lead to an increase in CTVE (see Fig. 14(b)). This is an emission paradox (Nagurney, 2000) that increasing road outflow capacity results in an increase in total vehicle emissions (and hence CTVE). The reason for the occurrence of this emission paradox is that the short link had a higher emission rate due to a higher road grade. More vehicles used the short link after its outflow capacity was improved. Hence, the total vehicle emission increased.

Example 5: The addition of a link may result in an increase in TSTT and total vehicle emissions

This example adopted two test networks as shown in Fig. 15. Fig. 15(a) shows the network before a new link was constructed. This network had three nodes, two links, and two OD pairs (i.e., OD pairs (1,2) and (1,3). The number of lanes, the outflow capacity (in veh/s), the grade (in percent), and the free-flow travel time (in s) of each link are also given in order inside round brackets next to the link in Fig. 15(a). Fig. 15(b) reveals the network after link 2-3 was constructed to connect Node 2 and Node 3. The parameters associated with the new link are also provided in Fig. 15(b). We varied the dispersion parameter θ for both the cases with and without link addition, solved the corresponding SDUO problem under different levels of OD demand, and computed the changes in network performance in Fig. 16. The increases in TSTT and CTVE are presented in Fig. 16. We can observe that the increases in TSTT and CTVE are positive for all the values of θ. This result implies that the addition of a link to a network may lead to a decrease in network performance in terms of TSTT and CTVE.

Example 6: An increase in travel demand may result in a decrease in total vehicle emissions.

The network in Fig. 15(b) was also adopted in this example. In this example, we fixed the travel demand of OD pair (1,3) to be 1.0 veh/s, but varied the travel demand of OD pair (1,2) from 0.6 veh/s to 1.4 veh/s. The CTVEs under different demand levels are graphically shown in Fig. 17. We can observe that each of the curves of CTVEs is not monotone as the travel demand of OD pair (1,2) grows up. In particular, when the demand increases from a low level to a medium level, CTVE decreases. This result implies that a decrease in the travel demand of OD pair (1,2) may result in an increase in total vehicle emissions. When the traffic demand of this OD pair increased, the congestion on link 1-2, the only path between OD pair (1,2), increased and hence the congestion on path 1-2-3 between OD pair (1,2) increased. As a result, fewer vehicles traveling between OD pair (1,3) used this route and hence link 2-3. The emission rate of link 2-3 was larger than other links due a higher grade. The reduction in vehicle emissions from this link was larger than the increment in vehicle emissions from other links. Therefore, fewer total vehicle emissions were generated.

6. Conclusion

This paper proposes an intersection-movement-based SDUO route choice problem in which we focus on the travelers' choice of intersection movements. Two existing definitions on the efficient path set, i.e., Definitions D1 and D2, are considered for the SDUO problem, and two FP models (i.e.,

Model I and Model II) are respectively developed to formulate the SDUO problem in terms of the OD-based intersection movement choice probabilities and the destination-based intersection movement choice probabilities. We can prove that the two models are equivalent under Definition D2. We also developed the MLTM, which is derived from the LTM, as a network flow propagation model which takes the demands and the choice of probabilities of intersection movements as inputs. A solution algorithm based on the SRAM was developed to solve the proposed SDUO problem.

Numerical examples were set up to test the performance of the proposed models and algorithm. The numerical results show that Model II is consistent with Model I under Definition D2, and Model II can be solved more efficiently. The proposed DTA models were further used to evaluate the effects of information quality and road network improvement on network performance. Through specific examples, we found four interesting phenomena that can occur in networks: (1) an improvement of information quality results in an increase in TSTT and total vehicle emissions, (2) an improvement of road outflow capacity results in an increase in total vehicle emissions, (3) the addition of a link results in an increase in TSTT and total vehicle emissions, and (4) a decrease in travel demand may result in an increase in total vehicle emissions. These examples demonstrated that the information quality, the transportation network topology, link capacity, as well as the travel demand must be taken into consideration in any policies aimed towards the reduction of TSTT and total vehicle emissions. The second and the third phenomena related to TSTT are usually referred to as Braess' paradox, i.e., improving transportation supply may worsen network performance. This kind of phenomena often occurs in highly non-linear systems.

The SDUO problem addressed in this paper is a logit-based route choice problem, which has long been known to have serious shortcomings in representing networks due to the assumption on the identical and independent Gumbel distributions of route travel time. In future research, we will be interested in extending the proposed approach to develop probit-based or the nested logit SDUO models to overcome the shortcomings. In addition, we also plan to use the proposed models for the offline transport planning and policy evaluation, such as advanced traveler information services, network design (e.g., Szeto et al., 2010, 2014; Miandoabchi et al., 2012a,b), signal control, intersection improvement, staggered work hours (e.g., Yushimito et al., 2014), incident detection (e.g., Ghosh and Smith, 2014), traffic flow/density forecasting (e.g., ; Szeto et al., 2009; Ye et al., 2012; Anand et al., 2014; Chiou et al., 2014), and so on. Furthermore, our current formulation and analysis is based on a discrete-time setting. Extending the formulation to a continuous-time one is more realistic because time is continuous by nature. However, this extension introduces difficulty in analysis and the analysis is not straightforward, because continuous-time DTA problems are infinite dimensional mathematical programming problems (e.g., Pang and Stewart, 2008), and fundamentally different from discrete-time problems that are usually finite dimensional mathematical programming problems (Wie et al., 2002). More up-to-date mathematical techniques such as differential variational inequality or differential complementarity system are required for analyzing the continuous-time DTA problems rigorously (e.g.,

Friesz et al., 2010; Ban et al., 2012; Friesz and Meimand, 2014). This extension deserves a full investigation and is left for future research.

Acknowledgements

Appendix A: Proof of Propositions

A.1. Proof of Proposition 1

Proof. Under Definition D1, all links on any reasonable paths take travelers closer to the destination. This is also one of the requirements of a link to be on any reasonable paths under Definition D2. Therefore, any path that belongs to the path set under Definition D1 also belongs to the path set under Definition D2. This completes the proof. □

A.2. Proof of Proposition 2

Proof. We assume that there exists a cycle in the sub-network G^{rs}, and the node sequence of the cycle is $\{i_1, i_2, \cdots, i_m, i_1\}$. According to Definition D1 or D2, all links on the efficient path take travelers closer to the destination, and hence we have $\pi^{i_1 s} > \pi^{i_2 s} > \cdots > \pi^{i_n s} > \pi^{i_1 s}$, where π^{is} is the minimum travel time from node i to destination s. This is a contradiction. Therefore, the sub-network G^{rs} is a directed acyclic graph. This completes the proof. □

A.3. Proof of Proposition 3

Proof. Under Definition D2, we have $A^{r_1 s}(i) = A^{r_2 s}(i)$ for all $i \in N^{r_1 s} \bigcap N^{r_2 s}$. This is because any link in $A^{r_1 s}(i)$ takes travelers closer to destination s under this definition, and so does any link in $A^{r_2 s}(i)$. Therefore, we can conclude $A^{r_1 s}(i)$ (or $A^{r_2 s}(i)$) is not affected by both origins r_1 and r_2, and hence

$$\bigcup_{i \in N^{r_1 s} \cap N^{r_2 s}} A^{r_1 s}(i) = \bigcup_{i \in N^{r_1 s} \cap N^{r_2 s}} A^{r_2 s}(i).$$ Because the two sub-networks are the same, the topological distances

of node i in the two sub-networks are the same. This completes the proof. □

A.4. Proof of Proposition 4

Proof. Since $a \in A^{r_1 s} \bigcap A^{r_2 s}$, we have $l_a \in N^{r_1 s} \bigcap N^{r_2 s}$ and $h_a \in N^{r_1 s} \bigcap N^{r_2 s}$. According to the proof of Proposition 3, we have $A^{r_1 s}(l_a) = A^{r_2 s}(l_a)$ and $A^{r_1 s}(h_a) = A^{r_2 s}(h_a)$ for all $a \in A^{r_1 s} \bigcap A^{r_2 s}$. Therefore, to prove $\alpha_a^{r_1 s}(k) = \alpha_a^{r_2 s}(k)$, we only need to prove that $W_a^{r_1 s}(k) = W_a^{r_2 s}(k)$ is satisfied for all $a \in A^{r_1 s} \bigcap A^{r_2 s}$. According to Eq. (18), we have

$$L_a^{r_1 s}(k) = L_a^{r_2 s}(k) = \exp\{\theta[\pi^{l_a s}(k) - \pi_a^{l_a s}(k)]\}, \forall a \in A^{r_1 s} \bigcap A^{r_2 s}, \text{ and } \forall r_1, r_2, s, k. \tag{58}$$

Consider the ascending passes of the sub-networks $G^{r_1 s}$ and $G^{r_2 s}$. According to Proposition 3, their common nodes have the same topological distance. Hence, we can set the topological order of any node $i \in N^{r_1 s} \bigcap N^{r_2 s}$ to be the same in the two ascending passes. Let $\{s, i_1, i_2, \cdots, i_m\}$ be the sequence of

the common nodes in the ascending passes with the superscript of node i representing the topological order of the node. The method of mathematical induction can then be used to prove that $W_a^{r_1s}(k) = W_a^{r_2s}(k)$ is satisfied for all $a \in A^{r_1s} \bigcap A^{r_2s}$.

We assume that $W_b^{r_1s}(k) = W_b^{r_2s}(k)$ for all $b \in A^{r_1s}(i_\iota), \iota < \kappa$, where κ is a positive integer. For a given link $a \in A^{r_1s}(i_\kappa)$, the topological order of node h_a is less than that of node l_a (i.e., i_κ), and hence we have $W_b^{r_1s}(k) = W_b^{r_2s}(k)$ for all $b \in A^{r_1s}(h_a)$. Equivalently, we have $W_b^{r_1s}(k + \tau_a(k)) = W_b^{r_2s}(k + \tau_a(k))$ for all $b \in A^{r_1s}(h_a)$. Substituting this and Eq. (58) into Eq. (19), we have $W_a^{r_1s}(k) = W_a^{r_2s}(k)$ for all $b \in A^{r_1s}(i_\kappa)$.

In addition, if $b \in A^{rs}(i_1)$, then we have $h_b = s$, since the destination is the only node with a smaller order than node i_1. Hence, $W_b^{r_1s}(k) = L_b^{r_1s}(k)$ and $W_b^{r_2s}(k) = L_b^{r_2s}(k)$ for all $b \in A^{r_1s}(i_1)$ (or $b \in A^{r_2s}(i_1)$). According to Eq. (58), we have $W_b^{r_1s}(k) = W_b^{r_2s}(k)$ for all $b \in A^{r_1s}(i_1)$. This completes the proof. □

A.5. Proof of Proposition 5

Proof. The method of mathematical induction will be used to prove this proposition. We assume that for all $\ell < k$, the link inflows $u_{ab}^{rs}(\ell)$ can be unilaterally determined by $\boldsymbol{\alpha}$.

In Eq. (35), $q^{rs}(k)$ is the traffic demand which is independent of $\boldsymbol{\alpha}$. Since the network traffic satisfies causality and $k - p_a(k) \le k - \tau_b^0$, $\zeta_{a,k}^\ell$ for all $\ell \le k - \tau_b^0$ can be determined by the traffic flows that enter into link b not later than $k - \tau_b^0$. The above assumption imply that both $\zeta_{a,k}^\ell$ and $u_{b'a}^{rs}(\ell)$ ($\ell \le k - \tau_b^0$) can be unilaterally determined by $\boldsymbol{\alpha}$. Therefore, if the link inflows $u_{ab}^{rs}(\ell)$ for all $\ell < k$ are unilaterally determined by $\boldsymbol{\alpha}$, then $u_{ab}^{rs}(k)$ will also be unilaterally determined by $\boldsymbol{\alpha}$. Since the network is empty initially, $v_{ba}^{rs}(0) = 0$. According to Eq. (35), we have $u_{ab}^{rs}(0) = \alpha_{ab}^{rs}(0)\alpha_a^{rs}(0)q^{rs}(0)\delta_{l_a}^r$. This implies that the link inflows $u_{ab}^{rs}(0)$ is unilaterally determined by $\boldsymbol{\alpha}$. This completes the proof. □

A.6. Proof of Proposition 6

Proof. Assume route p of OD pair (r, s) consists of links $\{a_1, a_2, \cdots, a_m\}$, where m is the number of links on that route and $a_1 \in A^{rs}(r)$. To simplify the notation, we add a wave line above the variable that is a function of time instant t. Under Assumption A1, $\tau_a(k) = \tilde{\tau}_a(t)$ when the interval length approaches to zero, where $t = k\delta$. The travel time of route p is summation of travel times of links on the route, we have $\tilde{\tau}_p(t) = \tau_p(k)$. The variables $\pi_a^{is}(k)$ and $\pi_{ab}^s(k)$ are minimum travel times of some routes, and hence they also converge to the continuous travel time $\tilde{\pi}_a^{is}(t)$ and $\tilde{\pi}_{ab}^s(t)$, respectively. Eqs. (28)-(33) indicate the likelihoods, the weights, and the choice probabilities of all intersection movements converge to their continuous forms.

Let $P_p^{rs}(k)$ be the route choice probability of vehicles departing r during interval k and using route p between origin r and destination s, and t_a be the arrival time at node l_a. Then, we have

$$P_p^{rs}(k) = G^{rs}(k)L_{a_1}^{rs}(k)\prod_{i=2}^{m-1}\{L_{a_ia_{i+1}}^{rs}(t_{a_i}/\delta)\} = G^{rs}(k)\tilde{L}_{a_1}^{rs}(t)\prod_{i=2}^{m-1}\{\tilde{L}_{a_ia_{i+1}}^{rs}(t_{a_i})\}, \tag{59}$$

where $G^{rs}(k)$ is the proportionality constant for flows between OD pair (r, s) departing during time interval k and is used to ensure that the sum of the right hand side of (59) over all routes between OD pair (r, s) equals 1. Substituting Eqs. (28) and (29) into Eq. (59), we have

$$P_p^{rs}(k) = G^{rs}(k)\exp\{\theta[\tilde{\pi}^{rs}(t) - \tilde{\pi}_{a_1}^{s}(t)]\}\prod_{i=1}^{m-1}\exp\{\theta[\tilde{\pi}_{a_i}^{s}(t_{a_i}) - \tilde{\pi}_{a_i a_{i+1}}^{s}(t_{a_i})]\}$$

$$= G^{rs}(k)\exp\{\theta[\tilde{\pi}^{rs}(t) - \tilde{\pi}_{a_1}^{s}(t)]\}\prod_{i=1}^{m-1}\exp\{\theta[\tilde{\pi}_{a_i}^{s}(t_{a_i}) - \tilde{\pi}_{a_{i+1}}^{s}(t_{a_i} + \tilde{\tau}_{a_i}(t_{a_i})) - \tilde{\tau}_{a_i}(t_{a_i})]\}$$

$$= G^{rs}(k)\exp\left\{\theta\left[\tilde{\pi}^{rs}(t) - \tilde{\pi}_{a_1}^{s}(t) + \sum_{i=1}^{m-1}\{[\tilde{\pi}_{a_i}^{s}(t_{a_i}) - \tilde{\pi}_{a_{i+1}}^{s}(t_{a_i} + \tilde{\tau}_{a_i}(t_{a_i})) - \tilde{\tau}_{a_i}(t_{a_i})]\}\right]\right\}$$

$$= G^{rs}(k)\exp\left\{\theta\left[\tilde{\pi}^{rs}(t) - \tilde{\pi}_{a_1}^{s}(t) + \sum_{i=1}^{m-1}\tilde{\pi}_{a_i}^{s}(t_{a_i}) - \sum_{i=1}^{m-1}\tilde{\pi}_{a_{i+1}}^{s}(t_{a_i} + \tilde{\tau}_{a_i}(t_{a_i})) - \sum_{i=1}^{m-1}\tilde{\tau}_{a_i}(t_{a_i})\right]\right\}$$

$$= G^{rs}(k)\exp\left\{\theta\left[\tilde{\pi}^{rs}(t) - \tilde{\pi}_{a_1}^{s}(t) + \sum_{i=1}^{m-1}\tilde{\pi}_{a_i}^{s}(t_{a_i}) - \sum_{i=1}^{m-1}\tilde{\pi}_{a_{i+1}}^{s}(t_{a_{i+1}}) - \sum_{i=1}^{m-1}\tilde{\tau}_{a_i}(t_{a_i})\right]\right\}$$

$$= G^{rs}(k)\exp\left\{\theta\left[\tilde{\pi}^{rs}(t) - \tilde{\pi}_{a_1}^{s}(t) + \sum_{i=1}^{m-1}\tilde{\pi}_{a_i}^{s}(t_{a_i}) - \sum_{i=2}^{m}\tilde{\pi}_{a_i}^{s}(t_{a_i}) - \sum_{i=1}^{m-1}\tilde{\tau}_{a_i}(t_{a_i})\right]\right\}$$

$$= G^{rs}(k)\exp\left\{\theta\left[\tilde{\pi}^{rs}(t) - \tilde{\pi}_{a_1}^{s}(t) + \left[\tilde{\pi}_{a_1}^{s}(t) + \sum_{i=2}^{m-1}\tilde{\pi}_{a_i}^{s}(t_{a_i})\right] - \left[\sum_{i=2}^{m-1}\tilde{\pi}_{a_i}^{s}(t_{a_i}) + \tilde{\pi}_{a_m}^{s}(t_{a_m})\right] - \sum_{i=1}^{m-1}\tilde{\tau}_{a_i}(t_{a_i})\right]\right\}$$

$$= G^{rs}(k)\exp\left\{\theta\left[\tilde{\pi}^{rs}(t) - \tilde{\pi}_{a_1}^{s}(t) + \tilde{\pi}_{a_1}^{s}(t) + \sum_{i=2}^{m-1}\tilde{\pi}_{a_i}^{s}(t_{a_i}) - \sum_{i=2}^{m-1}\tilde{\pi}_{a_i}^{s}(t_{a_i}) - \tilde{\tau}_{a_m}(t_{a_m}) - \sum_{i=1}^{m-1}\tilde{\tau}_{a_i}(t_{a_i})\right]\right\}$$

$$= G^{rs}(k)\exp\left\{\theta\left[\tilde{\pi}^{rs}(t) - \sum_{i=1}^{m}\tilde{\tau}_{a_i}(t_{a_i})\right]\right\}$$

$$= G^{rs}(k)\exp\left\{\theta\left[\tilde{\pi}^{rs}(t) - \tilde{\eta}_l^{rs}(t)\right]\right\}.$$

Note that by definitions, $\tilde{\pi}_{a_m}^{s}(t_{a_m}) = \tilde{\tau}_{a_m}(t_{a_m})$ and $\tilde{\eta}_p^{rs}(t) = \sum_{i=1}^{m}\tilde{\tau}_{a_i}(t_{a_i})$. Since $\sum_{p'}P_{p'}^{rs}(k) = 1$, summing over p' on both sides of equation leads to

$$\sum_{p'}P_{p'}^{rs}(k) = \sum_{p'}G^{rs}(k)\exp\left\{\theta\left[\tilde{\pi}^{rs}(t) - \tilde{\eta}_p^{rs}(t)\right]\right\} = \sum_{p'}G^{rs}(k)\exp\left\{\theta\left[\tilde{\pi}^{rs}(k) - \tilde{\eta}_p^{rs}(k)\right]\right\} = 1.$$

Rearranging the preceding equation, we have

$$G^{rs}(k) = \frac{1}{\sum_{p'}\exp\{\theta[\pi^{rs}(k) - \eta_{p'}^{rs}(k)]\}}.$$

Therefore, we have

$$P_p^{rs}(k) = \frac{\exp\{\theta[\pi^{rs}(k) - \eta_p^{rs}(k)]\}}{\sum_{p'}\exp\{\theta[\pi^{rs}(k) - \eta_{p'}^{rs}(k)]\}} = \frac{\exp\{-\theta\eta_p^{rs}(k)\}}{\sum_{p'}\exp\{-\theta\eta_{p'}^{rs}(k)\}}.$$

The rest of this proof is similar to that in Ran and Boyce (1996). This completes the proof. □

A.7. Proof of Proposition 7

Proof. Substituting Eqs. (1) and (2) into Eq. (28), we have

$$L_{ab}^{rs}(k) = \exp\{\theta[\pi_a^{s}(k) - \pi_{ab}^{s}(k)]\}$$

$$= \exp\{\theta[\tau_a(k) + \pi^{h_a s}(k + \tau_a(k)) - \tau_a(k) - \pi_b^{h_a s}(k + \tau_a(k))]\}$$

$$= \exp\{\theta[\pi^{h_a s}(k + \tau_a(k)) - \pi_b^{h_a s}(k + \tau_a(k))]\}$$

$$= L_b^{rs}(k + \tau_a(k)).$$

To prove $\alpha_{ab}^{rs}(k) = \alpha_b^{rs}(k + \tau_a(k)), \forall r,s,a,b,k$, we only need to prove that $W_{ab}^{rs}(k) = W_b^{rs}(k + \tau_a(k))$ is satisfied for all r, s, a, b, k. If so, we have

$$\alpha_{ab}^{rs}(k) = \frac{W_{ab}^{rs}(k)}{\sum_{b' \in A^{rs}(h_a)} W_{ab'}^{rs}(k)} = \frac{W_b^{rs}(k + \tau_a(k))}{\sum_{b' \in A^{rs}(h_a)} W_{b'}^{rs}(k + \tau_a(k))} = \alpha_b^{rs}(k + \tau_a(k)).$$

Let $\{s, i_1, i_2, \cdots, i_m, r\}$ be the node sequence in the ascending pass of the sub-network G^{rs}. The method of mathematical induction can also be used to prove that $W_{ab}^{rs}(k) = W_b^{rs}(k + \tau_a(k))$ is satisfied for all $r, s, a \in A^{rs}, b \in A^{rs}(h_a), k$.

We assume that $W_{bc}^{rs}(k) = W_c^{rs}(k + \tau_b(k))$ is satisfied for all $\iota < \kappa, b \in B^{rs}(i_\iota), c \in A^{rs}(i_\iota)$. For a given $a \in B^{rs}(i_\kappa)$ and $\forall b \in A^{rs}(h_a)$, the order of node h_b is less than that of node $l_b = h_a = i_\kappa$, and hence we have

$$W_{ab}^{rs}(k) = L_{ab}^{rs}(k)\left\{\delta_{h_b}^s + \sum_{c \in A^{rs}(h_b)} W_{bc}^{rs}(k + \tau_a(k))\right\}$$

$$= L_b^{rs}(k + \tau_a(k))\left\{\delta_{h_b}^s + \sum_{c \in A^{rs}(h_b)} W_c^{rs}(k + \tau_a(k) + \tau_b(k + \tau_a(k)))\right\}$$

$$= W_b^{rs}(k + \tau_a(k)).$$

Therefore, $W_{bc}^{rs}(k) = W_c^{rs}(k + \tau_b(k))$ is satisfied for all $b \in B^{rs}(i_\kappa), c \in A^{rs}(i_\kappa)$. In addition, if $b \in A^{rs}(i_1)$, then we have $h_b = s$, since the destination is the only node with a smaller order than node i_1. Therefore, $W_{ab}^{rs}(k) = W_b^{rs}(k + \tau_a(k))$ is satisfied for all $a \in B^{rs}(i_1), b \in A^{rs}(i_1)$. Hence, we have: $W_{ab}^{rs}(k) = L_{ab}^{rs}(k) = L_b^{rs}(k + \tau_a(k)) = W_b^{rs}(k + \tau_a(k))$. This completes the proof. □

A.8. Proof of Proposition 8

Proof. According to Proposition 7, we have $\alpha_{ab}^{rs*}(k) = \alpha_b^{rs*}(k + \tau_a^*(k)), \forall r,s,a,b,k$. Under Assumption A1, $\tau_a(k) = \tilde{\tau}_a(k\delta)$ when the interval length approaches to zero . Since $\tilde{p}_a^*(k\delta) = \tilde{\tau}_a^*(k\delta - \tilde{p}_a^*(k\delta))$, according to the proof of Proposition 6, we have

$$\tilde{\alpha}_{ab}^{rs*}(k\delta - \tilde{p}_a^*(k\delta)) = \tilde{\alpha}_b^{rs*}(k\delta - \tilde{p}_a^*(k\delta) + \tilde{\tau}_a^*(k\delta - \tilde{p}_a^*(k\delta))) = \tilde{\alpha}_b^{rs*}(k\delta) = \alpha_b^{rs*}(k).$$

Substituting Eqs. (34) and (35) in Eq. (26), we have

$$u_a^{rs*}(k) = \sum_{b \in B^{rs}(l_a)} u_{ab}^{rs*}(k) = \alpha_a^{rs*}(k)q^{rs}(k)\delta_{l_a}^r + \sum_{b \in B^{rs}(l_a)} \tilde{\alpha}_{ba}^{rs*}(k\delta - \tilde{p}_a^*(k\delta)) \sum_{\ell \le k - \tau_b^0} \zeta_{b,k}^{\ell*} u_b^{rs*}(\ell)$$

$$= \alpha_a^{rs*}(k)q^{rs}(k)\delta_{l_a}^r + \sum_{b \in B^{rs}(l_a)} \alpha_a^{rs*}(k) \sum_{\ell \le k - \tau_b^0} \zeta_{b,k}^{\ell*} u_b^{rs*}(\ell)$$

$$= \alpha_a^{rs*}(k)q^{rs}(k)\delta_{l_a}^r + \alpha_a^{rs*}(k) \sum_{b \in B^{rs}(l_a)} \sum_{\ell \le k - \tau_b^0} \zeta_{b,k}^{\ell*} u_b^{rs*}(\ell),$$

where $\delta_{l_a}^r = 1$ if $l_a = r$ and $\delta_{l_a}^r = 0$ otherwise. This completes the proof. □

A.9. Proof of Proposition 9

Proof. According to Propositions 4 and 7, we have $\alpha_{ab}^{r_1s}(k) = \alpha_b^{r_1s}(k + \tau_a(k))$, $\alpha_{ab}^{r_2s}(k) = \alpha_b^{r_2s}(k + \tau_a(k))$, and $\alpha_b^{r_1s}(k + \tau_a(k)) = \alpha_b^{r_2s}(k + \tau_a(k))$. Therefore, we have $\alpha_{ab}^{r_1s}(k) = \alpha_{ab}^{r_2s}(k)$. This completes the proof. □

A.10. Proof of Proposition 11

Proof. Since $A^s = \bigcup_r A^{rs}$, we have $A^{rs}(i) \subset A^s(i)$. Because link $a \in A^s(i)$ takes travelers closer to destination s under Definition D2, link a also takes travelers from origin r closer to destination s. Hence, $A^{rs}(i) = A^s(i)$, $\forall a \in A^s(i)$. □

A.11. Proof of Proposition 13

Proof. Let $\boldsymbol{\alpha}' = \mathbf{h}(\mathbf{y}(\bar{\boldsymbol{\alpha}}))$ and $\bar{\boldsymbol{\alpha}}'' = \bar{\mathbf{h}}(\bar{\boldsymbol{\alpha}})$, where $\boldsymbol{\alpha}' = [\alpha_a^{rs\,\prime}(k), \alpha_{ab}^{rs\,\prime}(k)]$ and $\bar{\boldsymbol{\alpha}}'' = [\alpha_a^{rs\,\prime\prime}(k), \alpha_{ab}^{s\,\prime\prime}(k)]$. According to Definition D8, we only need to prove that $\alpha_{ab}^{rs\,\prime}(k) = \alpha_{ab}^{s\,\prime\prime}(k)$ is satisfied for all $r, s, a \in A^{rs}, b \in A^{rs}(h_a), k$ and $\alpha_a^{rs\,\prime}(k) = \alpha_a^{rs\,\prime\prime}(k)$ is satisfied for all $r, s, a \in A^{rs}(r), k$.

To simplify the notation, we add an apostrophe and a ditto to the variables corresponding to $\boldsymbol{\alpha}'$ and $\bar{\boldsymbol{\alpha}}''$, respectively. We consider a particular OD pair (r, s) and the corresponding destination s. According to Definition D1, we have $L_{ab}^{rs\,\prime}(k) = L_{ab}^{s\,\prime\prime}(k)$, $\forall i \in N^{rs}, a \in A^{rs}(i), b \in A^{rs}(h_a)$ and $L_a^{rs\,\prime}(k) = L_a^{s\,\prime\prime}(k)$, $\forall a \in A^{rs}(r)$, since the likelihood of each intersection movement is calculated under the same set of link travel times.

To prove that $\alpha_{ab}^{rs\,\prime}(k) = \alpha_{ab}^{s\,\prime\prime}(k)$ is satisfied for all $a \in A^{rs}, b \in A^{rs}(h_a), k$, we only need to prove that $W_{ab}^{rs\,\prime}(k) = W_{ab}^{s\,\prime\prime}(k)$ is satisfied for all $a \in A^{rs}, b \in A^{rs}(h_a), k$. Let $\{s, i_1, i_2, \cdots, i_m, r\}$ be the node sequence in the ascending pass of the sub-network G^{rs}. The method of mathematical induction can also be used to prove that $W_{ab}^{rs\,\prime\prime}(k) = W_{ab}^{s\,\prime\prime}(k)$ is satisfied for all $a \in A^{rs}, b \in A^{rs}(h_a), k$.

We assume that $W_{ab}^{rs\,\prime}(k) = W_{ab}^{s\,\prime\prime}(k)$ is satisfied for all $\iota < \kappa, b \in B^{rs}(i_\iota), c \in A^{rs}(i_\iota)$. For a given $a \in B^{rs}(i_\kappa)$, and $\forall b \in A^{rs}(h_a)$, the topological order of node h_b is less than that of node $l_b = h_a = i_\kappa$. According to Proposition 11, we have $A^{rs}(h_b) = A^s(h_b)$ and hence we have

$$W_{ab}^{rs\,\prime}(k) = L_{ab}^{rs\,\prime}(k) \left\{ \delta_{h_b}^s + \sum_{c \in A^{rs}(h_b)} W_{bc}^{rs\,\prime}(k + \tau_a(k)) \right\} = L_{ab}^{s\,\prime\prime}(k) \left\{ \delta_{h_b}^s + \sum_{c \in A^s(h_b)} W_{bc}^{s\,\prime\prime}(k + \tau_a(k)) \right\} = W_{ab}^{s\,\prime\prime}(k).$$

In addition, if $b \in A^{rs}(i_1)$, then we have $h_b = s$, since the destination is the only node with a smaller topological order than node i_1. Therefore, we have: $W_{ab}^{rs\,\prime}(k) = L_{ab}^{rs\,\prime}(k) = L_{ab}^{s\,\prime\prime}(k) = W_{ab}^{rs\,\prime\prime}(k)$. Therefore, $W_{ab}^{rs\,\prime}(k) = W_{ab}^{s\,\prime\prime}(k)$ is satisfied for all $a \in A^{rs}, b \in A^{rs}(h_a), k$, and hence $\alpha_{ab}^{rs\,\prime}(k) = \alpha_{ab}^{s\,\prime\prime}(k)$ is satisfied for all $a \in A^{rs}, b \in A^{rs}(h_a), k \in K$.

According to Eq. (31), we have

$$W_a^{rs\,\prime}(k) = L_a^{is\,\prime}(k) \left\{ \delta_{h_a}^s + \sum_{b \in A^{rs}(h_a)} W_{ab}^{is\,\prime}(k) \right\} = L_a^{s\,\prime\prime}(k) \left\{ \delta_{h_a}^s + \sum_{b \in A^s(h_a)} W_{ab}^{s\,\prime\prime}(k) \right\} = W_a^{s\,\prime\prime}(k).$$

Therefore, Eqs. (33) and (46) imply that $\alpha_a^{rs\,\prime}(k) = \alpha_a^{rs\,\prime\prime}(k)$ is satisfied for all $r, s, a \in A^{rs}(r), k$. This completes the proof. □

A.12. Proof of Proposition 14

Proof. Since $\bar{\boldsymbol{\alpha}}^*$ is a solution of the FP problem (49), we have $\bar{\boldsymbol{\alpha}}^* = \bar{\mathbf{h}}(\bar{\boldsymbol{\alpha}}^*)$. According to Proposition 13, we have $\mathbf{h}(\mathbf{y}(\bar{\boldsymbol{\alpha}}^*)) = \mathbf{y}(\bar{\mathbf{h}}(\bar{\boldsymbol{\alpha}}^*)) = \mathbf{y}(\bar{\boldsymbol{\alpha}}^*)$. $\mathbf{y}(\bar{\boldsymbol{\alpha}}^*)$ satisfies Eq. (36), and hence $\boldsymbol{\alpha}^* = \mathbf{y}(\bar{\boldsymbol{\alpha}}^*)$ is a solution of the FP problem (36). □

A.13. Proof of Proposition 16

Proof. Let $\overline{\boldsymbol{\alpha}} = [\overline{\alpha}_a^{rs}(k), \overline{\alpha}_{ab}^s(k)]$, $\overline{\boldsymbol{\alpha}}' = \overline{\mathbf{h}}(\overline{\boldsymbol{\alpha}}) = [\overline{\alpha}_a^{rs'}(k), \overline{\alpha}_{ab}^s{}'(k)]$, $\boldsymbol{\alpha} = \mathbf{y}(\overline{\boldsymbol{\alpha}}) = [\tilde{\alpha}_a^{rs}(k), \tilde{\alpha}_{ab}^{rs}(k)]$, and $\boldsymbol{\alpha}' = \mathbf{h}(\mathbf{y}(\overline{\boldsymbol{\alpha}})) = [\tilde{\alpha}_a^{rs'}(k), \tilde{\alpha}_{ab}^{rs'}(k)]$. According to the definition of transfer function, we have $\tilde{\alpha}_a^{rs}(k) = \overline{\alpha}_a^{rs}(k)$, $\forall r,s,a \in A^{rs}(r),t$ and $\tilde{\alpha}_{ab}^{rs}(k) = \overline{\alpha}_{ab}^s(k)$, $\forall r,s,a \in A^{rs}, b \in A^{rs}(h_a),k$. According to Proposition 13, we have $\mathbf{h}(\mathbf{y}(\overline{\boldsymbol{\alpha}})) = \mathbf{y}(\overline{\mathbf{h}}(\overline{\boldsymbol{\alpha}}))$. Therefore, $\tilde{\alpha}_a^{rs'}(k) = \overline{\alpha}_a^{rs'}(k)$, $\forall r,s,a \in A^{rs}(r),k$ and $\tilde{\alpha}_{ab}^{rs'}(k) = \overline{\alpha}_{ab}^s{}'(k)$, $\forall r,s,a \in A^{rs}, b \in A^{rs}(h_a),k$. Since $A^s = \bigcup_r A^{rs}$, we have $\sum_r \delta_{l_a}^r \geq 1$ for all $a \in A^s$. i.e., any link in A^s is used at least by one OD pair.

We have

$$\sum_r \sum_s \sum_{a \in A^{rs}} \sum_{b \in A^{rs}(h_a)} \left| \tilde{\alpha}_{ab}^{rs'}(k) - \tilde{\alpha}_{ab}^{rs}(k) \right|$$

$$= \sum_s \sum_r \sum_{a \in A^{rs}} \sum_{b \in A^{rs}(h_a)} \left| \overline{\alpha}_{ab}^s{}'(k) - \overline{\alpha}_{ab}^s(k) \right|$$

$$= \sum_s \sum_r \sum_{a \in A^s} \sum_{b \in A^s(h_a)} \left| \overline{\alpha}_{ab}^s{}'(k) - \overline{\alpha}_{ab}^s(k) \right| \delta_{l_a}^r$$

$$= \sum_s \sum_{a \in A^s} \sum_{b \in A^s(h_a)} \left| \overline{\alpha}_{ab}^s{}'(k) - \overline{\alpha}_{ab}^s(k) \right| \sum_r \delta_{l_a}^r$$

$$\geq \sum_s \sum_{a \in A^s} \sum_{b \in A^s(h_a)} \left| \overline{\alpha}_{ab}^s{}'(k) - \overline{\alpha}_{ab}^s(k) \right|,$$

and hence

$$\left\| \mathbf{h}(\mathbf{y}(\overline{\boldsymbol{\alpha}})) \text{-} \mathbf{y}(\overline{\boldsymbol{\alpha}}) \right\|_1 = \sum_k \sum_r \sum_s \sum_{a \in A^{rs}(r)} \left| \tilde{\alpha}_a^{rs'}(k) - \tilde{\alpha}_a^{rs}(k) \right| + \sum_k \sum_r \sum_s \sum_{a \in A^{rs}} \sum_{b \in A^{rs}(h_a)} \left| \tilde{\alpha}_{ab}^{rs'}(k) - \tilde{\alpha}_{ab}^{rs}(k) \right|$$

$$\geq \sum_k \sum_r \sum_s \sum_{a \in A^{rs}(r)} \left| \overline{\alpha}_a^{rs'}(k) - \overline{\alpha}_a^{rs}(k) \right| + \sum_k \sum_s \sum_{a \in A^s} \sum_{b \in A^s(h_a)} \left| \overline{\alpha}_{ab}^s{}'(k) - \overline{\alpha}_{ab}^s(k) \right|$$

$$= \left\| \overline{\mathbf{h}}(\overline{\boldsymbol{\alpha}}) \text{-} \overline{\boldsymbol{\alpha}} \right\|_1 .$$

We also have:

$$\left\| \mathbf{h}(\mathbf{y}(\overline{\boldsymbol{\alpha}})) \text{-} \mathbf{y}(\overline{\boldsymbol{\alpha}}) \right\|_\infty = \max \left\{ \left| \tilde{\alpha}_a^{rs'}(k) - \tilde{\alpha}_a^{rs}(k) \right|, \left| \tilde{\alpha}_{ab}^{rs'}(k) - \tilde{\alpha}_{ab}^{rs}(k) \right| \right\}$$

$$= \max \left\{ \left| \overline{\alpha}_a^{rs'}(k) - \overline{\alpha}_a^{rs}(k) \right|, \left| \overline{\alpha}_{ab}^s{}'(k) - \overline{\alpha}_{ab}^s(k) \right| \right\}$$

$$= \left\| \overline{\mathbf{h}}(\overline{\boldsymbol{\alpha}}) \text{-} \overline{\boldsymbol{\alpha}} \right\|_\infty .$$

This completes the proof. □

A.14. Proof of Proposition 18

Proof. Under Assumption A3, the minimum travel time for travelers departing from node i to destination s during interval k, i.e., $\pi^{is}(k)$, is continuous with respect to $\boldsymbol{\alpha}$, because $\pi^{is}(k)$ is the minimum travel time of all routes connecting i and s, and all route travel times are continuous with respect to $\boldsymbol{\alpha}$ (See Proposition 17). Moreover, the link travel time $\tau_a(k)$ is continuous with respect to $\boldsymbol{\alpha}$. Hence, $\boldsymbol{\pi}$, which is defined by nested functions as shown in Eqs. (1) and (2), is a continuous function of the intersection movement choice probability vector $\boldsymbol{\alpha}$. This further implies that $\mathbf{h}(\boldsymbol{\alpha})$ defined by Eqs. (28)-(33) is continuous with respect to $\boldsymbol{\pi}$. Therefore, $\mathbf{h}(\boldsymbol{\alpha})$ is a continuous function of $\boldsymbol{\alpha}$.□

A.15. Proof of Proposition 20

Proof. According to definition, $\boldsymbol{\alpha}$ is the vector of intersection movement choice probabilities, and hence we have $\boldsymbol{\alpha} \in \Phi$, where Φ is the domain of the function $\mathbf{h}(\boldsymbol{\alpha})$ described by

$$\Phi = \{\boldsymbol{\alpha} \geq \mathbf{0}: \sum_{a \in B^{rs}(r)} \alpha_a^{rs}(k) = 1, \forall r \in R, s \in S, k \in K_D, \sum_{b \in B^{rs}(h_a)} \alpha_{ab}^{rs}(k) = 1, \forall r \in R, s \in S, a \in A^{rs}, k \in K\}.$$

Obviously, Φ is a compact and convex nonempty set. $\mathbf{h}(\boldsymbol{\alpha})$ is defined by Eqs. (28)-(33), and Eqs. (32) and (33) imply that the codomain of $\mathbf{h}(\boldsymbol{\alpha})$ is a subset of Φ, i.e., $\mathbf{h}(\boldsymbol{\alpha}) \in \Phi$. According to Proposition 18, $\mathbf{h}(\boldsymbol{\alpha})$ is a continuous function of $\boldsymbol{\alpha}$. By Brouwer's FP theorem, the FP $\boldsymbol{\alpha} = \mathbf{h}(\boldsymbol{\alpha})$ has at least one solution. This completes the proof. □

Appendix B: The modified link transmission model

B.1. Notations for the modified link transmission model

The following notations are used throughout this subsection:

$N^{rs}(k)$ cumulative number of vehicles that depart from origin r to destination s by the end of interval k.

$N_a^r(k)$ cumulative number of vehicles that depart from origin r by the end of interval k and travel through link $a \in A(r)$.

$N_a^{rs}(k)$ cumulative number of vehicles that depart from origin r by the end of interval k and travel through link $a \in A(r)$ to destination s.

$M_a^r(k)$ cumulative number of vehicles that depart from origin r and enter link $a \in A(r)$ by the end of interval k.

$M_a^{rs}(k)$ cumulative number of vehicles that depart from origin r to destination s and enter link $a \in A(i)$ by the end of interval k.

$S_a(k)$ maximum possible number of vehicles leaving link a during interval k.

$R_a(k)$ maximum possible number of vehicles received by link a during interval k.

$G_{ab}(k)$ number of vehicles that travel from link a to link $b \in A(h_a)$ during interval k.

$G_a^r(k)$ number of vehicles that depart from origin r and enter link $a \in A(r)$ during interval k.

$R_{ab}(k)$ maximum possible number of vehicles from link a received by link $b \in A(h_a)$ during interval k.

$R_a^r(k)$ maximum possible number of vehicles from origin r received by link $a \in A(r)$ during interval k.

$S_{ab}(k)$ maximum possible number of vehicles sent from link a to link $b \in A(h_a)$ during interval k.

$C_a(k)$ outflow capacity of link a during interval k.

$Q_a(k)$ inflow capacity of link a during interval k.

$q^{rs}(k)$ demand between (r, s) generated during interval k.

B.2. Link model

We adopt the modified LTM as the traffic flow model in this paper. It is based on the LTM that assumes a triangular fundamental diagram (see Fig. 18). Five parameters are found in the diagram,

namely a free-flow speed (v), a backward shock-wave speed (w), a maximum flow or capacity (q_{max}), a critical density (ρ_{crit}), and a jam density (ρ_{jam}). Only v, q_{max} and ρ_{jam} are used to define the diagram and the other two parameters can be deduced from them.

The LTM consists of two types of submodels, namely link and node models. In the link model, the LTM uses Newell's (1993) cumulative flow theory to depict traffic propagation on a link and determine the sending and receiving flows of that link. The sending flow of a link is constrained by both the boundary condition at the upstream end of that link and the outflow capacity of that link; the receiving flow of a link is constrained by both the boundary condition at the downstream end of a link and the inflow capacity of that link. According to Newell's simplified theory, if a free-flow traffic state occurs at the downstream end of a link at the end of interval k, then this state must have been emitted from the upstream end d_a / v_a time units earlier (i.e., the free-flow travel time τ_a^0), where v_a is the free-flow speed on link a; if a congested traffic state occurs at the upstream end of a link at the end time of interval k, then this state must have been emitted from the downstream end of the link ($-d_a / w_a$) time units earlier, where w_a is the free-flow speed on link a. The sending flow and the receiving flow of link a can be mathematically expressed as follows:

$$S_a(k) = \min\{U_a(k+1-d_a/(v_a\delta)) - V_a(k), C_a(k)\} \text{, and} \tag{60}$$

$$R_a(k) = \min\{V_a(k+1+d_a/(w_a\delta)) + d_a\rho_{jam} - U_a(k), Q_a(k)\} . \tag{61}$$

B.3. Node model

The node model in the LTM is used to determine the transition flows from upstream links to downstream links at an intersection, which rely on the sending and receiving flows corresponding to the intersection movements. For a particular link a, its sending flow can be disaggregated into sending sub-flows by its receiving links, we have:

$$S_{ab}(k) = U_{ab}(U_a^{-1}(V_a(k) + S_a(k))/\delta) - V_{ab}(k), \tag{62}$$

where $U_a^{-1}(V_a(k) + S_a(k))$ is the time instant at which $V_a(k) + S_a(k)$ vehicles have cumulatively entered link a. Because this time instant divided by the interval length δ may not be an integer, linear interpolation is used to calculate $U_{ab}(.)$. The linear interpolation may also be used for other cumulative flows.

Since the inflow capacity of each receiving link is limited, a priority parameter $p_{ab}(k)$ is adopted to assign the receiving flow to link b from each upstream link a. The resulting receiving sub-flow $R_{ab}(k)$ is given as follows:

$$R_{ab}(k) = p_{ab}(k)R_b(k) . \tag{63}$$

In this paper, the priority parameters are assumed to be determined by the outflow capacity of each link, and hence the following is used:

$$p_{ab}(k) = \frac{C_a(k)}{\displaystyle\sum_{a' \in B(l_b)} C_{a'}(k)}, \forall b, a \in B(l_b) .$$

It is assumed that the road traffic flows satisfy link FIFO, i.e., vehicles that enter a link earlier leave it sooner. According to this assumption, we can deduce a time instant λ_a^k by which vehicles entering link a completely leave this link before the end of interval $k+1$. This time instant is

determined by the sending and receiving sub-flows associated with link a, and can be formulated as follows:

$$\lambda_a^k = \max\{\lambda : 0 \le \lambda \le k\delta, U_{ab}(\lambda/\delta) - V_{ab}(k) \le \min\{S_{ab}(k), R_{ab}(k)\}, \forall b \in B(h_a)\}. \tag{64}$$

Then, the transition flow $G_{ab}(k)$ can be formulated as follows:

$$G_{ab}(k) = U_{ab}(\lambda_a^k/\delta) - V_{ab}(k). \tag{65}$$

B.4. Loading of demand at nodes

We assume that vehicles from upstream links have a higher priority to enter downstream links than the vehicles generated at the intersection of both upstream and downstream links, and the vehicles generated at the intersections (i.e., origins) can enter downstream links only if the capacities of downstream links are not fully used. The receiving flow of the downstream link from an origin can be calculated by:

$$R_a^r(k) = R_a(k) - \sum_{b \in B(r)} G_{ba}(k). \tag{66}$$

By definition, we have:

$$N^{rs}(k+1) = N^{rs}(k) + q^{rs}(k). \tag{67}$$

The intersection movement choice probabilities can be used to calculate the cumulative generated flows with respect to each downstream link, given by:

$$N_a^{rs}(k+1) = N_a^{rs}(k) + \alpha_a^{rs}(k)q^{rs}(k), \tag{68}$$

where the second term on the right hand side of Eq. (68) is the flow from origin r passing through link a during interval k to destination s.

By definition, we also have:

$$N_a^r(k+1) = \sum_s N_a^{rs}(k+1). \tag{69}$$

The sending flow and the transition flow from origin i to link a can then be formulated as follows:

$$S_a^r(k) = N_a^r(k+1) - M_a^r(k), \text{ and} \tag{70}$$

$$G_a^r(k) = \min\{S_a^r(k), R_a^r(k)\}. \tag{71}$$

B.5. Update cumulative flows

If Model I is adopted, then the inputs to the LTM are OD-based intersection movement choice probabilities, which are used to update the cumulative flows. Using the link and node models, we can determine the inflow and outflow of each link, and consequently update the cumulative inflows, given by:

$$U_a(k+1) = U_a(k) + \sum_{b \in B(l_a)} G_{ba}(k) + G_a^{l_a}(k), \tag{72}$$

where the second and third terms in the right hand side of Eq. (72) are the inflows from upstream links and from the origin node, respectively.

The cumulative number of vehicles that enter the network can be updated as follows:

$$\begin{cases} M_a^r(k+1) = N_a^r(\lambda_a^{r,k}/\delta), \\ M_a^{rs}(k+1) = N_a^{rs}(\lambda_a^{r,k}/\delta), \end{cases} \tag{73}$$

where $\lambda_a^{r,k}$ is the time instant before which vehicles from origin r has completely entered link a by the end of interval $k + 1$, and $\lambda_a^{r,k}$ can be expressed as follows:

$$\lambda_a^{r,k} = \max\{\lambda : 0 \le \lambda \le (k+1)\delta, N_a^r(\lambda/\delta) - M_a^r(k) \le G_a^r(k)\}.$$

Similarly, the last vehicle that passes the downstream boundary of link a during interval $k + 1$ enters link a at time instant λ_a^k (see Eq. (64)). Hence, we have:

$$\begin{cases} V_a(k+1) = U_a(\lambda_a^k/\delta), \\ V_{ab}(k+1) = U_{ab}(\lambda_a^k/\delta), \end{cases} \text{ and} \tag{74}$$

$$\begin{cases} V_a^{rs}(k+1) = U_a^{rs}(\lambda_a^k/\delta), \\ V_{ab}^{rs}(k+1) = U_{ab}^{rs}(\lambda_a^k/\delta). \end{cases} \tag{75}$$

The conservation law of traffic flows at intersections can be used to determine the cumulative link inflows with respect to OD pair (r, s) into downstream link a:

$$U_a^{rs}(k+1) = \sum_{b \in B(l_a)} V_{ba}^{rs}(k+1) + \delta_a''^r M_a^{l_a s}(k+1), \tag{76}$$

where $\delta_a''^r$ equals 1 if $a \in A^{rs}(r)$ and 0 otherwise.

The intersection movement choice probabilities can be used to calculate the cumulative inflows on each approach with respect to each OD pair or each destination and downstream link as follows:

$$U_{ab}^{rs}(k+1) = U_{ab}^{rs}(k) + \alpha_{ab}^{rs}(k)[U_a^{rs}(k+1) - U_a^{rs}(k)]. \tag{77}$$

Then, the total cumulative flows on each approach to a downstream link can be aggregated by summing up the cumulative flows to all OD pairs through the downstream link, given by

$$U_{ab}(k+1) = \sum_r \sum_s U_{ab}^{rs}(k+1). \tag{78}$$

Using the link and node models, we can determine the inflow and outflow of each link, and consequently update the cumulative inflows.

If Model II is adopted, then the inputs of the LTM are destination-based intersection movement choice probabilities, and hence Eqs. (75)-(78) are, respectively, replaced by the following equations to update the cumulative flows:

$$\begin{cases} V_a^s(k+1) = U_a^s(\lambda_a^k/\delta), \\ V_{ab}^s(k+1) = U_{ab}^s(\lambda_a^k/\delta), \end{cases} \tag{79}$$

$$U_a^s(k+1) = \sum_{b \in B(l_a)} V_{ba}^s(k+1) + M_a^{l_a s}(k+1), \tag{80}$$

$$U_{ab}^s(k+1) = U_{ab}^s(k) + \alpha_{ab}^s(k)[U_a^s(k+1) - U_a^s(k)], \text{ and} \tag{81}$$

$$U_{ab}(k+1) = \sum_s U_{ab}^s(k+1). \tag{82}$$

B.6. MLTM solution algorithm

The MLTM solution algorithm can be summarized as follows:

Step 0: *Initialization.* Initialize all the cumulative inflows and outflows to be zero, and set $k = 0$.

Step 1: *Determining sending and receiving flows by the link model.* Use Eqs. (60) and (61) to determine the sending flow $S_a(k)$ and the receiving flow $R_a(k)$ for all $a \in A$.

Step 2: *Determining transition flows by the node model.* Compute the transition flow $G_{ab}(k)$ for all $a \in A, b \in A(h_a)$ by Eq. (65).

Step 3: *Loading origin flows into networks*. Update the cumulative departure flows by Eqs. (67)-(69) and compute the transition flow $G_a^r(k)$ for all $r \in R, a \in A(r)$ by Eq.(71).

Step 4: *Cumulative flow update*. Compute the cumulative inflows and outflows by Eqs. (72)-(78) for Model I, or by Eqs. (72)-(74) and Eqs. (79)-(82) for Model II.

Step 5: *Checking the termination criterion*. Terminate the algorithm if $k = \underline{K}$. Otherwise, set $k = k + 1$, and return to Step 2.

The MLTM is a physical queue model that considers the physical lengths of vehicles and the storage capacity of each link. Once a physical queue fills up all the vacant spaces of a link (i.e., the queue uses up the storage capacity of a link), the queue will pass over the upstream junction and spill backward to upstream links. In the physical-queue model, link inflow constraints (e.g., the receiving flow conditions in the MLTM) are considered and the link inflow drops to zero if a queue spills backward to the upstream boundary of a link.

References

Anand, A., Ramadurai, G., Vanajakshi, L., 2014. Data fusion-based traffic density estimation and prediction. Journal of Intelligent Transportation Systems 18(4), 367-378.

Balijepalli, N.C., Ngoduy, D., Watling, D.P., 2014. The two-regime transmission model for network loading in dynamic traffic assignment problems. Transportmetrica A 10 (7), 563-584.

Ban, X., Liu, H.X., Ferris, M.C., Ran, B., 2008. A link-node complementarity model and solution algorithm for dynamic user equilibria with exact flow propagations. Transportation Research Part B 42 (9), 823-842.

Ban, X., Pang, J.S., Liu, H.X., Ma, R., 2012. Modeling and solving continuous-time instantaneous dynamic user equilibria: A differential complementarity systems approach. Transportation Research Part B 46, 389-408.

Bliemer, M.C.J., Versteegt, H.H., Castenmiller, R.J., 2004. INDY: a new analytical multiclass dynamic traffic assignment model. In: TRISTAN V Conference Proceedings, Guadeloupe, France.

Blumberg-Nitzani, M., Bar-Gera, H., 2014. The effect of signalised intersections on dynamic traffic assignment solution stability. Transportmetrica A 10(7), 622-646.

Cantarella, G.E., 1997. A general fixed-point approach to multimode multi-user equilibrium assignment with elastic demand. Transportation Science 31 (2), 107-128.

Carey, M., 1987. Optimal time-varying flows on congested networks. Operations Research 35 (1), 58-69.

Carey, M., 1990. Extending and solving a multi-period congested network flow model. Computers and Operations Research 17 (5), 495-507.

Carey, M., Srinivasan, A., 1993. Externalities, average and marginal costs, and tolls on congested networks with time-varying flows. Operations Research 41 (1), 217-231.

Carey, M., Subrahmanian, E., 2000. An approach to modeling time-varying flows on congested networks. Transportation Research Part B 34 (3), 157-183.

Carey, M., Watling, D., 2012. Dynamic traffic assignment approximating the kinematic wave model: System optimum, marginal costs, externalities and tolls. Transportation Research Part B 46(5), 634-648.

Chabini, I., 2001. Analytical dynamic network loading problem: Formulation, solution algorithms, and computer implementations. Transportation Research Record 1771 191-200.

Chen, H.K., Feng, G., 2000. Heuristics for the stochastic/dynamic user-optimal route choice problem European Journal of Operational Research 126(1), 13-30.

Chen, X., Li, Z., Li, L., Shi, Q., 2014. Characterising scattering features in flow–density plots using a stochastic platoon model. Transportmetrica A 10(9), 820-848.

Chiou, Y.-C., Lan, L.W., Tseng, C.-M., 2014. A novel method to predict traffic features based on rolling self-structured traffic patterns. Journal of Intelligent Transportation Systems 18(4), 352-366.

Chiabaut, N., Xie, X., Leclercq, L., 2014. Performance analysis for different designs of a multimodal urban arterial. Transportmetrica B 2(3), 229-245.

Daganzo, C.F., 1994. The cell transmission model: A simple dynamic representation of highway traffic. Transportation Research Part B 28 (4), 269-287.

Daganzo, C.F., 1995. The cell transmission model, Part II: Network traffic. Transportation Research Part B 29 (2), 79-93.

Dial, R.B., 1971. A probabilistic multipath traffic assignment model which obviates path enumeration. Transportation Research 5 (2), 83-111.

Doan, K., Ukkusuri, S.V., 2012. On the holding-back problem in the cell transmission based dynamic traffic assignment models. Transportation Research Part B 46 (9), 1218-1238.

Florian, M., Mahut, M., Tremblay, N., 2008. Application of a simulation-based dynamic traffic assignment model. European Journal of Operational Research 189 (3), 1381-1392.

Friesz, T.L., Bernstein, D., Suo, Z., Tobin, R.L., 2001. Dynamic network user equilibrium with state-dependent time lags. Networks and Spatial Economics 1, 319–347.

Friesz, T.L., Bernstein, D., Smith, T.E., Tobin, R.L., Wie, B., 1993. A variational inequality formulation of the dynamic networks user equilibrium problem. Operations Research 41 (1), 179-191.

Friesz, T.L., Han, K., Neto, P.A., Meimand, A., Yao, T., 2013. Dynamic user equilibrium based on a hydrodynamic model. Transportation Research Part B 47, 102-126.

Friesz, T.L., Kim, T., Kwon, C., Rigdon, M., 2010. Approximate network loading and dual-time-scale dynamic user equilibrium. Transportation Research Part B 45 (1), 176–207.

Friesz, T.L., Luque, J., Tobin, R.L., Wie, B.W., 1989. Dynamic network traffic assignment considered as a continuous-time optimal-control problem. Operations Research 37 (6), 893-901.

Friesz, T.L., Meimand, A., 2014. A differential variational inequality formulation of dynamic network user equilibrium with elastic demand. Transportmetrica A 10(7), 661-668.

Ghosh, B., Smith, D.P., 2014. Customization of automatic incident detection algorithms for signalized urban arterials. Journal of Intelligent Transportation Systems 18(4), 426-441.

Golani, H., Waller, S.T., 2004. Combinatorial approach for multiple-destination user optimal dynamic traffic assignment. Transportation Research Record 1882, 70-78.

Han, K., Friesz, T.L., Yao, T., 2013a. A partial differential equation formulation of Vickrey's bottleneck model, part I: Methodology and theoretical analysis. Transportation Research Part B 49, 55-74.

Han, K., Friesz, T.L., Yao, T., 2013b. A partial differential equation formulation of Vickrey's bottleneck model, part II: Numerical analysis and computation. Transportation Research Part B 49, 75-93.

Han, K., Friesz, T.L., Yao, T., 2013c. Existence of simultaneous route and departure choice dynamic user equilibrium. Transportation Research Part B, 53, 17-30.

Han, S., 2003. Dynamic traffic modelling and dynamic stochastic user equilibrium assignment for general road networks. Transportation Research Part B 37(3), 225-249.

Huang, H.J., 1995. A combined algorithm for solving and calibrating the stochastic traffic assignment model. Journal of Operational Research Society 46 (8), 977-987.

Huang, H.J., Lam, W.H.K., 2002. Modeling and solving the dynamic user equilibrium route and departure time choice problem in network with queues. Transportation Research Part B 36 (3), 253-273.

Iryo, T., 2013. Properties of dynamic user equilibrium solution: existence, uniqueness, stability, and robust solution methodology. Transportmetrica B 1(1), 52-67.

Jiang, Y.Q., Wong, S.C., Ho, H.W., Zhang, P., Liu, R.X., Sumalee A., 2011. A dynamic traffic assignment model for a continuum transportation system. Transportation Research Part B 45 (2), 343-363.

Kuwahara, M., Akamatsu, T., 2001. Dynamic user optimal assignment with physical queues for a many-to-many OD pattern. Transportation Research Part B 35 (5), 461-479.

Leblanc, L.J., 1975. An algorithm for the discrete network design problem. Transportation Science 9 (3), 183-199.

Li, Y., Ziliaskopoulos, A.K., Waller, S.T., 1999. Linear programming formulations for system optimum dynamic traffic assignment with arrival time-based and departure time-based demands. Transportation Research Record 1667, 52-59.

Lighthill, M.H., Whitham, G.B., 1955. On kinematics wave II: A theory of traffic flow on long crowed roads. Proceedings of the Royal Society, London, Series A 229 (1178), 317-345.

Lim, Y., Heydecker, B., 2005. Dynamic departure time and stochastic user equilibrium assignment. Transportation Research Part B 39(2), 97-118.

Liu, H., He, X., He, B., 2009. Method of successive weighted averages (MSWA) and self-regulated averaging schemes for solving stochastic user equilibrium problem. Networks and Spatial Economics 9 (4), 485-503.

Lo, H.K., Szeto, W.Y., 2002a. A cell-based variational inequality formulation of the dynamic user optimal assignment problem. Transportation Research Part B 36 (5), 421-443.

Lo, H.K., Szeto, W.Y., 2002b. A cell-based dynamic traffic assignment model: formulation and properties. Mathematical and Computer Modelling 35(7-8), 849-865.

Lo, H.K., Szeto, W.Y., 2004. Modeling advanced traveler information services: Static versus dynamic paradigms. Transportation Research Part B 38(6), 495-515.

Long, J.C., Gao, Z.Y., Szeto, W.Y., 2011. Discretised link travel time models based on cumulative flows: Formulation and properties. Transportation Research Part B 45 (1), 232-254.

Long, J.C., Gao, Z.Y., Zhang, H.Z., Szeto, W.Y., 2010. A turning restriction design problem in urban road networks. European Journal of Operational Research 206(3), 569-578.

Long, J.C., Huang, H.J., Gao, Z.Y., 2013a. Discretised route travel time models based on cumulative flows. Journal of Advanced Transportation 47 (1), 105-125.

Long, J.C., Huang, H.J., Gao, Z.Y., Szeto, W.Y., 2013b. An intersection-movement-based dynamic user optimal route choice problem. Operations Research 61 (5), 1134-1147.

Long, J.C., Szeto, W.Y., Huang H.J., 2014. A bi-objective turning restriction design problem in urban road networks. European Journal of Operational Research 237 (1), 426-439.

Ma, R., Ban, X., Pang, J.-S., 2014. Continuous-time dynamic system optimum for single-destination traffic networks with queue spillbacks. Transportation Research Part B 68, 98-122.

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.

Mayeres, I., Ochelen, S., Proost, S., 1996. The marginal external costs of urban transport. Transportation Research Part D 1 (2), 111-130.

Meng, Q., Khoo, H.L. 2012. A computational model for the probit-based dynamic stochastic user optimal traffic assignment problem. Journal of Advanced Transportation 46 (1), 80-94.

Merchant, D.K, Nemhauser, G.L., 1978a. A model and an algorithm for the dynamic traffic assignment. Transportation Science 12 (3), 183-199.

Merchant, D.K, Nemhauser, G.L., 1978b. Optimality conditions for a dynamic traffic assignment model. Transportation Science 12 (3), 200-207.

Mesa-Arango, R., Ukkusuri, S.V., 2014. Modeling the car-truck interaction in a system-optimal dynamic traffic assignment model. Journal of Intelligent Transportation Systems 18(4), 327-338.

Miandoabchi, E., Farahani, R.Z., Szeto, W.Y., 2012a. Bi-objective bimodal urban road network design using hybrid metaheuristics. Central European Journal of Operations Research 20(4), 583-621.

Miandoabchi, E., Farahani, R.Z., Dullaert, W., Szeto, W.Y., 2012b. Hybrid evolutionary metaheuristics for concurrent multi-objective design of urban road and public transit networks. Networks and Spatial Economics 12(3), 441-480.

Nagurney, A., 2000. Congested urban transportation networks and emission paradoxes. Transportation Research Part D 5 (2), 145–151.

Newell, G.F., 1993. A simplified theory on kinematic wave in highway traffic, Part I: General theory; Part II: Queuing at freeway bottlenecks; Part III: Multi-destination flows. Transportation Research Part B 27 (4), 281-314.

Nie, X.J., Zhang, H.M., 2005. Delay-function-based link models: Their properties and computational issues. Transportation Research Part B 39 (8), 729-751.

Nie, Y., 2011. A cell-based Merchant-Nemhauser model for the system optimum dynamic traffic assignment problem. Transportation Research Part B 45 (2), 329-342.

Nielsen, O.A., Frederiksen, R.D., Simonsen, N., 1998. Stochastic user equilibrium traffic assignment with turn-delays in intersections. International Transactions in Operational Research 5 (6), 555-568.

Ng, M., Waller, S.T., 2012. A dynamic route choice model considering uncertain capacities. Computer-Aided Civil and Infrastructure Engineering 27 (4), 231-243.

Ngoduy, D., 2013. Platoon-based macroscopic model for intelligent traffic flow. Transportmetrica B 1(2), 153-169.

Pang, J., Stewart, D., 2008. Differential variational inequalities. Mathematical Programming 113 (2), 345-424.

Penic, M.A., Upchurch, J., 1992. TRANSYT-7F: Enhancement for fuel consumption, pollution emissions, and user costs. Transportation Research Record 1360, 104-111.

Qian, Z.S., Shen, W., Zhang, H.M., 2012. System-optimal dynamic traffic assignment with and without queue spillback: Its path-based formulation and solution via approximate path marginal cost. Transportation Research Part B 46 (7), 874-893.

Raadsen, M.P.H., Mein, H.E., Schilpzand, M.P., Brandt, F., 2010. Implementation of a single dynamic traffic assignment model on mixed urban and highway transport networks including junction modeling. DTA symposium, Takayama, Japan.

Ran, B., Boyce, D.E., 1996. Modeling Dynamic Transportation Network: An Intelligent Transportation System Oriented Approach. Springer, Heidelberg.

Ran, B., Boyce, D.E., Leblanc, L.J., 1993. A new class of instantaneous dynamic user-optimal traffic assignment models. Operations Research 41 (1) 192-202.

Richards, P.I., 1956. Shock waves on the highway. Operations Research 4 (1), 42-51.

Sheffi, Y., 1985. Urban Transportation Networks: Equilibrium Analysis with Mathematical Programming Methods. Prentice-Hall, Englewood Cliffs, New Jersey, USA.

Shen, W., Zhang, H.M., 2014. System optimal dynamic traffic assignment: Properties and solution procedures in the case of a many-to-one network. Transportation Research Part B 65, 1-17.

Smith, M.J., 1993. A new dynamic traffic and the existence and calculation of dynamic user equilibria on congestion capacity-constrained road networks. Transportation Research Part B 27 (1), 49-63.

Sumalee, A., Zhong, R.X., Pan, T.L., Szeto, W.Y., 2011. Stochastic cell transmission model (SCTM): a stochastic dynamic traffic model for traffic state surveillance and assignment. Transportation Research Part B, 45(3), 507-533.

Zhong, R.X., Sumalee, A., Pan, T.L., Lam, W.H.K., 2013. Stochastic cell transmission model for traffic network with demand and supply uncertainties. Transportmetrica A 9(7), 567-602.

Szeto, W.Y., 2008. The enhanced lagged cell transmission model for dynamic traffic assignment. Transportation Research Record 2085, 76-85.

Szeto, W.Y., Ghosh, B., Basu, B., O'Mahony, M., 2009. Cell-based short-term traffic flow forecasting using time series modelling. ASCE Journal of Transportation Engineering 135(9), 658-667.

Szeto, W.Y., Jiang, Y., Sumalee, A., 2011. A cell-based model for multi-class doubly stochastic dynamic traffic assignment. Computer-Aided Civil and Infrastructure Engineering 26 (8), 595-611.

Szeto, W.Y., Lo, H.K., 2004. A cell-based simultaneous route and departure time choice model with elastic demand. Transportation Research Part B 38 (7), 593-612.

Szeto, W.Y., Lo, H.K., 2006. Dynamic traffic assignment: Properties and extensions. Transportmetrica, 2 (1), 31-52.

Szeto, W.Y., Jaber, X.Q., O'Mahony, M., 2010. Time-dependent discrete network design frameworks considering land use. Computer-Aided Civil and Infrastructure Engineering 25(6), 411–426.

Szeto, W.Y., Jiang, Y., Wang, D.Z.W., Sumalee, A., 2014. A sustainable road network design problem with land use transportation interaction over time. Networks and Spatial Economics, DOI: 10.1007/s11067-013-9191-9.

Tian, Y., Chiu, Y.-C., 2014. A variable time-discretization strategies-based, time-dependent shortest path algorithm for dynamic traffic assignment. Journal of Intelligent Transportation Systems 18(4), 339-351.

Ukkusuri, S.V., Han, L., Doan, K., 2012. Dynamic user equilibrium with a path based cell transmission model for general traffic networks. Transportation Research Part B 46(10), 1657-1684.

Unnikrishnan, A., Waller, S.T., 2009. User equilibrium with recourse. Networks and Spatial Economics 9 (4), 575-593.

Waller, S.T., Ziliaskopoulos, A.K., 2006a. A combinatorial user optimal dynamic traffic assignment algorithm. Annals of Operations Research 144 (1), 249-261.

Waller, S.T, Ziliaskopoulos, A.K., 2006b. A chance-constrained based stochastic dynamic traffic assignment model: Analysis, formulation and solution algorithms. Transportation Research Part C 14 (6), 418-427.

Waller, S.T., Fajardo, D., Duell, M., Dixit, V., 2013. Linear programming formulation for strategic dynamic traffic assignment. Networks and Spatial Economics 13 (4), 427-443.

Waller, S.T., Ziliaskopoulos, A.K., 2001. Stochastic dynamic network design model. Transportation Research Record 1771, 106-113.

Wie, B.W., Tobin, R.L., Carey, M., 2002. The existence, uniqueness and computation of an arc-based dynamic network user equilibrium formulation. Transportation Research Part B 36 (10), 897-918.

Yagar, S., 1971. Dynamic traffic assignment by individual path minimisation and queuing. Transportation Research 5 (3), 179-196.

Yang, H., Meng, Q. 1998. Departure time, route choice and congestion toll in a queuing network with elastic demand. Transportation Research Part B 32, 247-260.

Ye, Q., Szeto, W.Y., Wong, S.C., 2012. Short-term traffic speed forecasting based on data recorded at irregular intervals. IEEE Transactions on Intelligent Transportation Systems 13(4), 1727-1737.

Yushimito, W.F., Ban, X., Holguín-Veras, J., 2014. A two-stage optimization model for staggered work hours. Journal of Intelligent Transportation Systems 18(4), 410-425.

Yperman, I., 2007. The link transmission model for dynamic network loading. Ph D. dissertation, Katholieke Universiteit Leuven, Leuven, Belgium.

Zhang, H.M., Nie, Y., Qian, Z., 2013. Modelling network flow with and without link interactions: the cases of point queue, spatial queue and cell transmission model. Transportmetrica B 1(1), 33-51.

Zhu, F., Lo, H.K., Lin, H.-Z., 2013. Delay and emissions modelling for signalised intersections. Transportmetrica B 1(2), 111-135.

Zhong, R.X., Sumalee, A., Pan, T.L., Lam, W.H.K., 2013. Stochastic cell transmission model for traffic network with demand and supply uncertainties. Transportmetrica A 9(7), 567-602.

Ziliaskopoulos, A.K., 2000. A linear programming model for the single destination system optimum dynamic traffic assignment problem. Transportation Science 34 (1), 37-44.

Table 1
The parameters of emission rate functions (Penic and Upchurch, 1992).

Grade of roadway		$g < -1$	$-1 \le g < 0$	$0 \le g < 1$	$1 \le g < 2$	$2 \le g$
NO$_x$	$A_m(s)$	$1.7325e^{0.011815s}$	$1.7325e^{0.011815s}$	$1.5718e^{0.040732s}$	$4.2279e^{0.050231s}$	$1.1096s^{1.2624}$
	$B_m(s)$	$1.7325e^{0.011815s}$	$1.5718e^{0.040732s}$	$4.2279e^{0.050231s}$	$1.1096s^{1.2624}$	$3.0515s^{1.1111}$
	C_m			1000		
VOC	$A_m(s)$	$2.9262e^{0.020118s}$	$2.9262e^{0.020118s}$	$2.7843e^{0.015062s}$	$307248e^{0.023644s}$	$4.2789e^{0.033437s}$
	$B_m(s)$	$2.9262e^{0.020118s}$	$2.7843e^{0.015062s}$	$307248e^{0.023644s}$	$4.2789e^{0.033437s}$	$5.2305e^{0.040708s}$
	C_m			10000		
CO	$A_m(s)$	$3.0741e^{0.0093192s}$	$3.0741e^{0.0093192s}$	$3.3963e^{0.014561s}$	$4.6927e^{0.031454s}$	$5.5812e^{0.047365s}$
	$B_m(s)$	$3.0741e^{0.0093192s}$	$3.3963e^{0.014561s}$	$4.6927e^{0.031454s}$	$5.5812e^{0.047365s}$	$6.5785e^{0.064392s}$
	C_m			10000		
	$h(g)$	$g+2$	$g+1$	g	$g-1$	$g-2$

Table 2
The monetary valuation of each type of pollutants (Mayeres et al., 1996).

Type of pollutant m	NO$_x$	VOC	CO
η_m (in Euros/kg)	13.80	2.95	0.01

Table 3
The path sets under Definitions D1 and D2 in Example 1.

	Under Definition D1		Under Definition D2	
	OD pair (1,3)	OD pair (2,3)	OD pair (1,3)	OD pair (2,3)
Path set	1-2-6-3 1-4-5-3	2-4-5-3 2-6-3 2-4-6-5-3	1-2-4-5-3 1-2-6-3 1-2-6-5-3 1-4-5-3	2-4-5-3 2-6-3 2-4-6-5-3

Table 4
Intersection movement choice probabilities under the free-flow condition in Example 1.

	Under Definition D1		Under Definition D2	
Intersection movement	OD pair (1,3)	OD pair (2,3)	OD pair (1,3)	OD pair (2,3)
2-6-3	1	0.731	0.731	0.731
4-6-5	0	0.269	0.269	0.269

Table 5
The free-flow travel time and the number of lanes of each link in the Sioux Falls network.

Link numbers	τ_a^0 (s)	Number of lanes	Link numbers	τ_a^0 (s)	Number of lanes	Link numbers	τ_a^0 (s)	Number of lanes
1 and 3	140	4	22 and 47	120	1	46 and 67	100	2
2 and 5	100	4	25 and 26	70	2	49 and 52	50	1
4 and 14	120	1	27 and 32	120	2	50 and 55	70	3
6 and 8	100	3	28 and 43	140	2	53 and 58	50	1
7 and 35	100	4	29 and 48	120	1	56 and 60	100	4
9 and 11	50	3	30 and 51	190	1	59 and 61	100	1
10 and 31	140	1	33 and 36	140	1	62 and 64	140	1
12 and 15	100	1	34 and 40	100	1	63 and 68	120	1
13 and 23	120	2	37 and 38	70	4	65 and 69	50	1
16 and 19	50	1	39 and 74	100	1	66 and 75	70	1
17 and 20	70	1	41 and 44	120	1	70 and 72	100	1
18 and 54	50	4	42 and 71	100	1	73 and 76	50	1
21 and 24	240	1	45 and 57	100	3			

Table 6
The average CPU time (in s) for each component of the algorithm

Definition	Model	Initialization	DNL	Link travel time determination	SNL	Gap determination	One iteration
D1	I	0.3819	0.4058	0.0048	0.2811	0.0641	0.7558
D2	I	0.7606	0.5269	0.0043	0.3466	0.0816	0.9595
D2	II	0.1594	0.1134	0.0039	0.0922	0.0491	0.2586
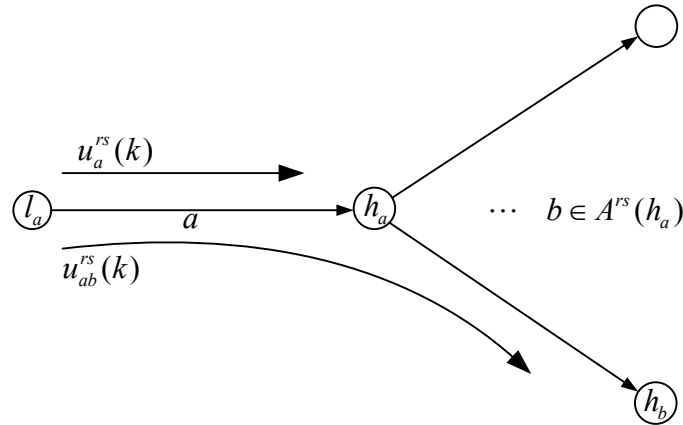
Fig. 1. A sketch of intersection movements.
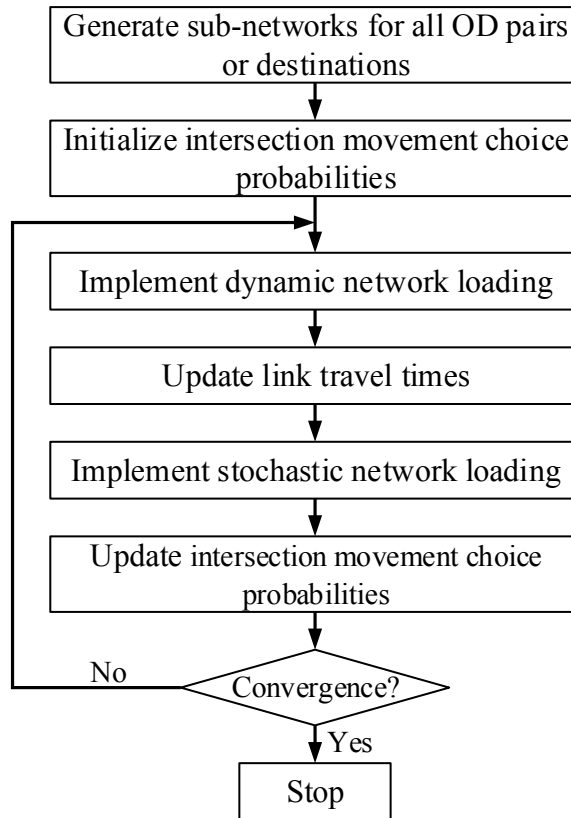
Fig. 2. The general scheme of the whole procedure for solving the intersection-movement-based SDUO problem.
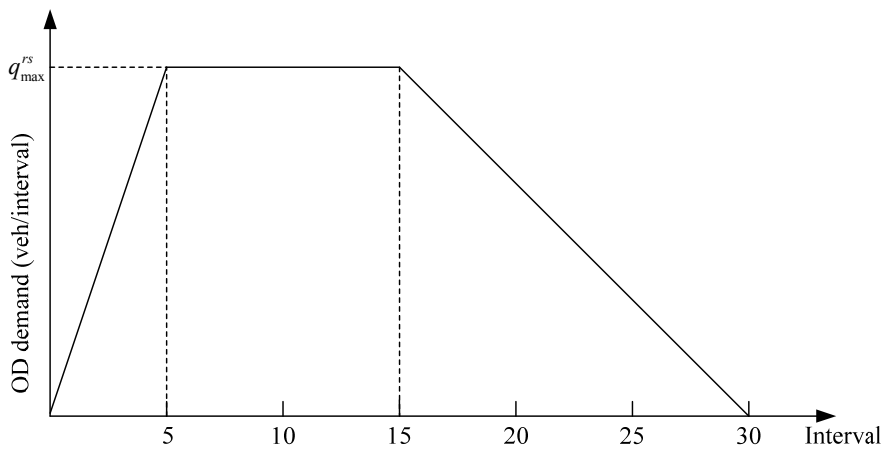
Fig. 3. The profile of OD demand.
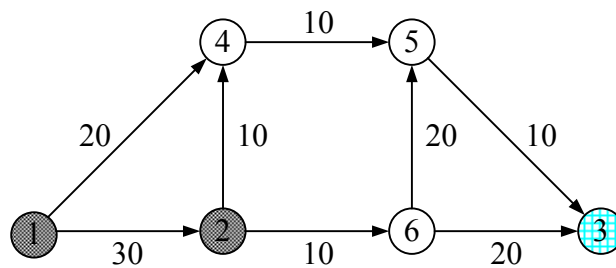
Fig. 4. An illustration network for Example 1.
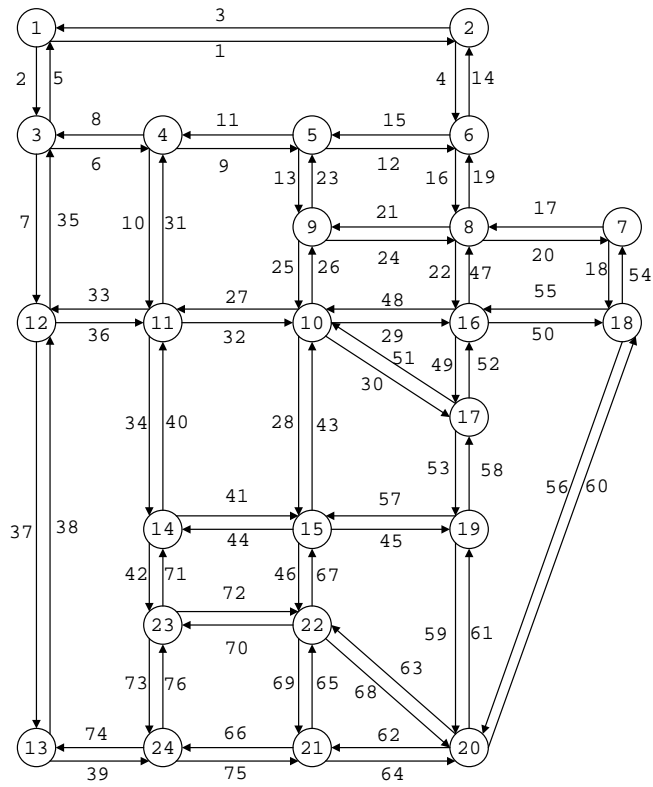
Fig. 5. The Sioux Falls network.
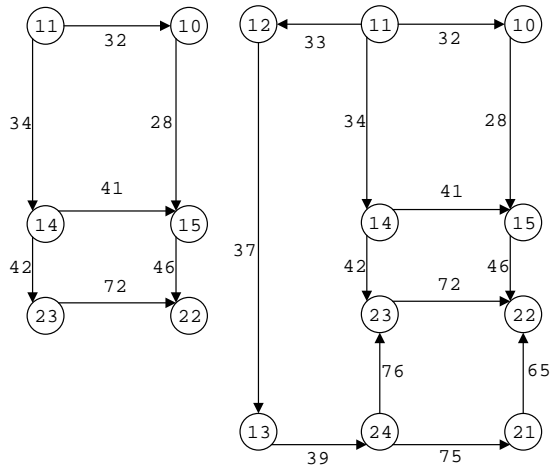
(a) Definition D1   (b) Definition D2

Fig. 6. The sub-network of OD pair (11, 22) under the two definitions.
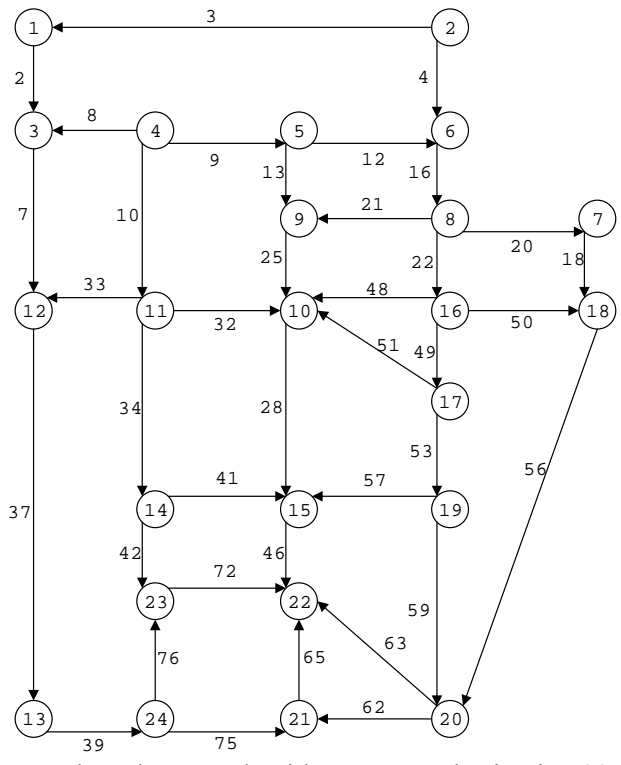
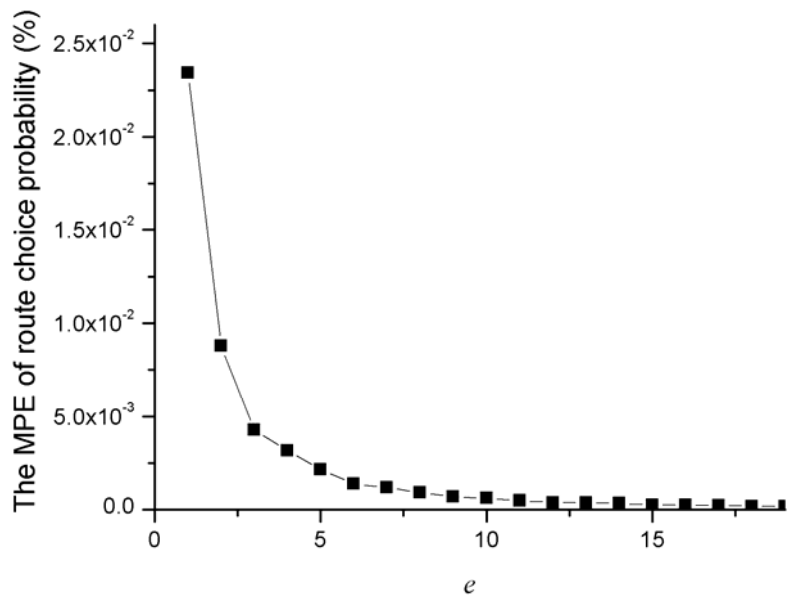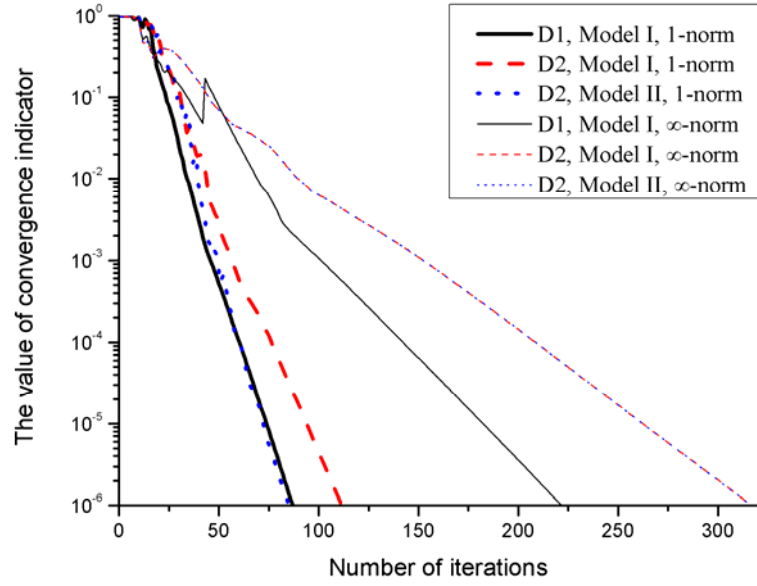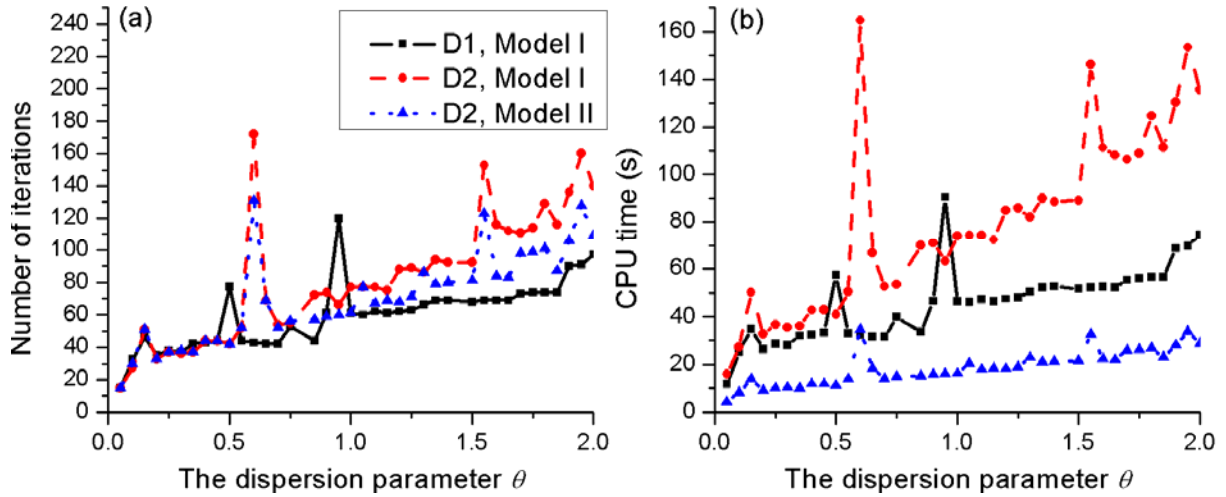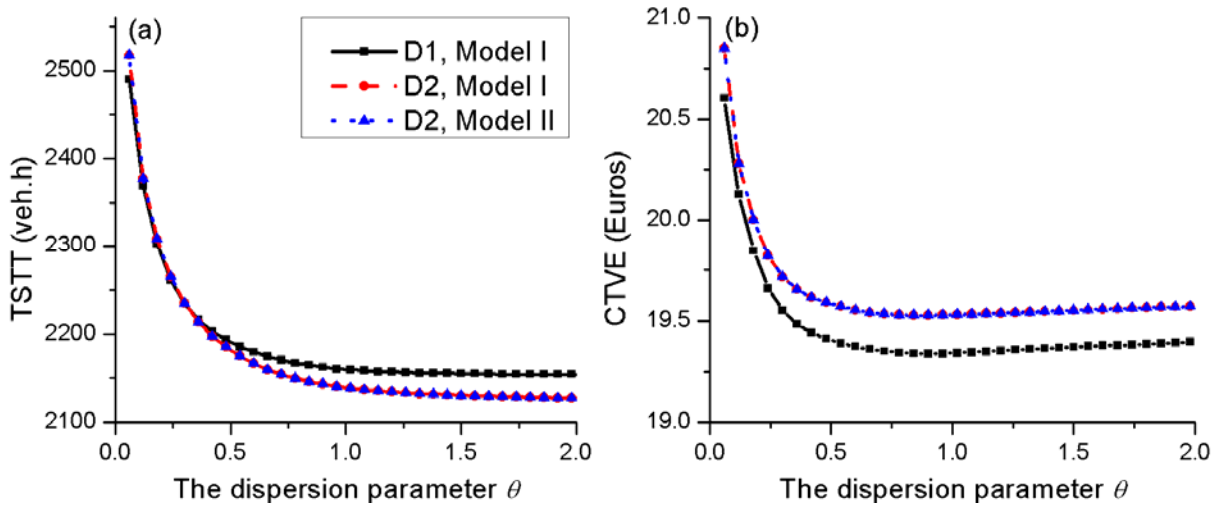**Fig. 7.** The sub-network with respect to destination 22.

Fig. 8. The MPE of route choice probability.

Fig. 9. The convergence of the solution algorithm for the SDUO problem ($\theta = 1.0$).

Fig. 10. The influence of the value of the dispersion parameter θ on algorithmic convergence.

Fig. 11. The influence of the value of the dispersion parameter θ on network performance for various models in Example 2.

(2, 0.5, 4, 20)

(4, 1.0, 2, 40)
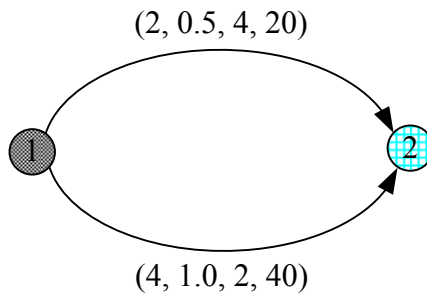
Fig. 12. The test network for Examples 3 and 4.
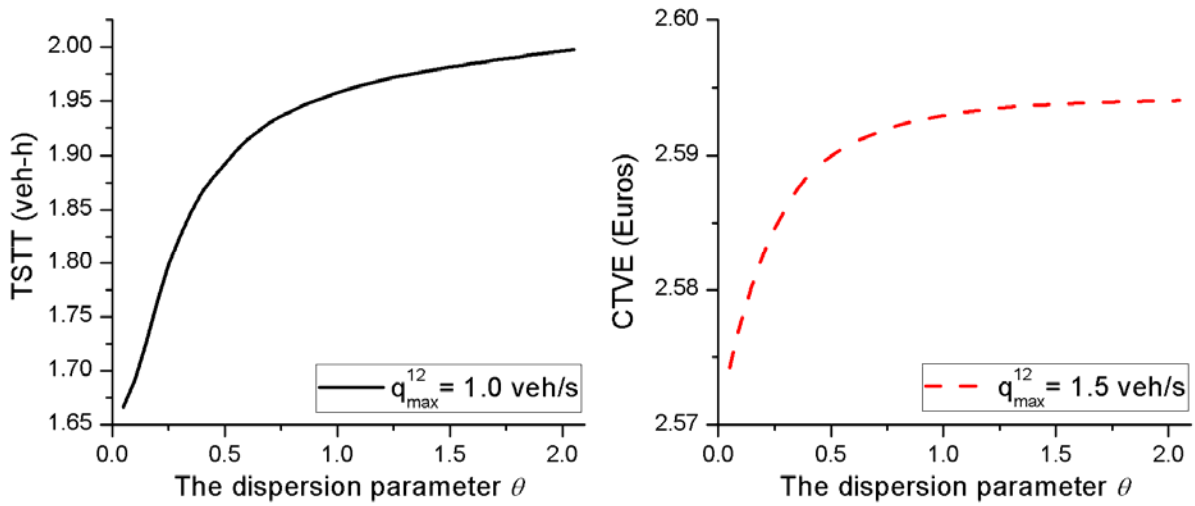
Fig. 13. The influence of the value of the dispersion parameter θ on network performance in Example 3.
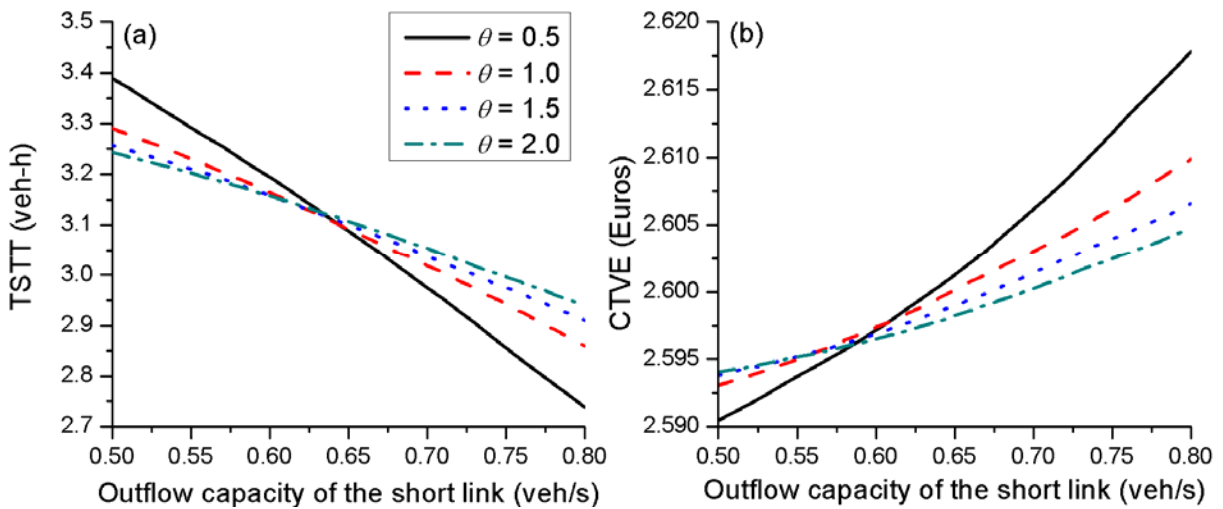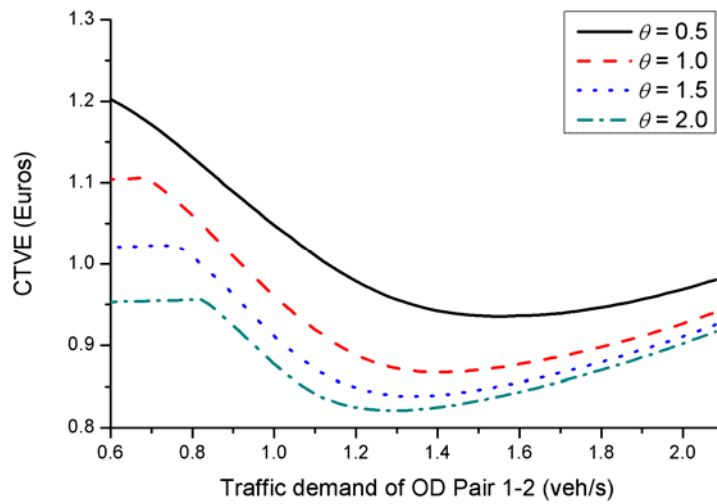

Fig. 14. The influence of outflow capacity improvement of a link on network performance in Example 4.
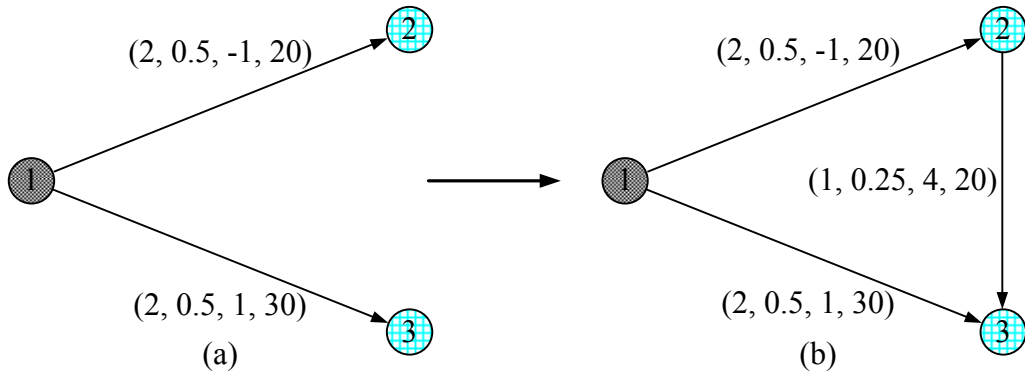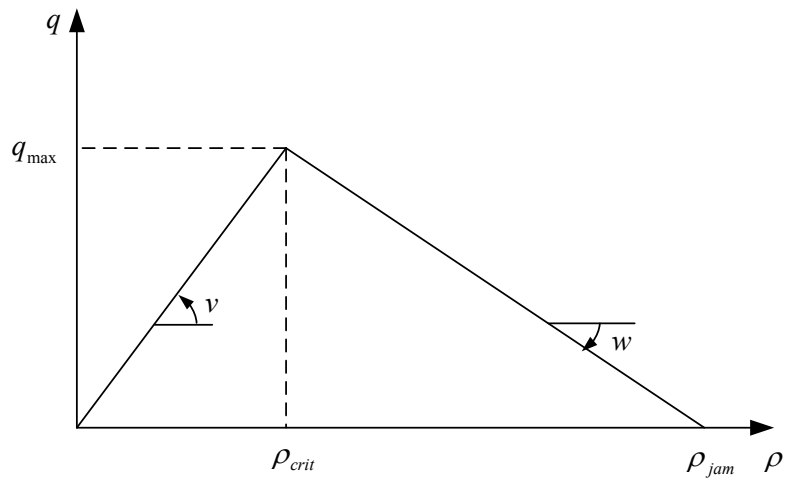
Fig. 15. Test networks for Examples 5 and 6.
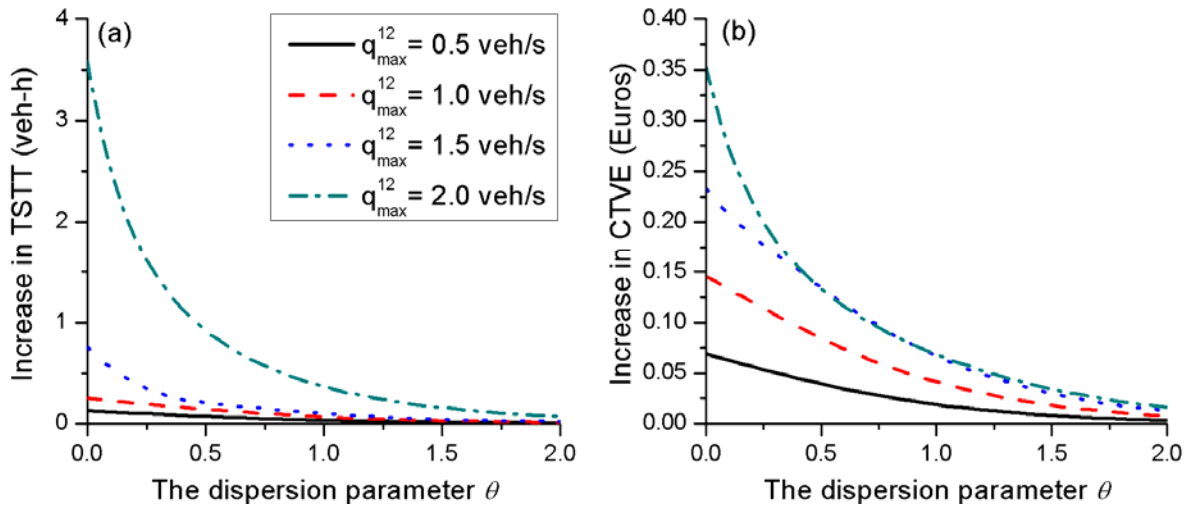
Fig. 16. The increase in (a) TSTT and (b) CTVE after the addition of link 2-3.

Fig. 17. CTVE against the traffic demand of OD pair (1,2).

Fig. 18. A triangular fundamental diagram (Yperman, 2007).