Submitted to *Operations Research*
manuscript OPRE-2015-11-655.R3

Link-based System Optimum Dynamic Traffic Assignment Problems in General Networks

Jiancheng Long
School of Automotive and Transportation Engineering, Hefei University of Technology, Hefei 230009, China, jianchenglong@hfut.edu.cn

W.Y. Szeto
Department of Civil Engineering, The University of Hong Kong, Pokfulam Road, Hong Kong, China, ceszeto@hku.hk

Most current system optimum dynamic traffic assignment (SO-DTA) models do not contain first-in-first-out (FIFO) constraints and are limited to single-destination network applications. In this study, we introduce the link transmission model (LTM) for the development of SO-DTA models either with or without FIFO constraints for general network applications. The proposed SO-DTA models include the LTM and can lead to a linear programming (LP) formulation if the FIFO constraints are not explicitly captured. The vehicle holding problem can be addressed by adding a penalty term to the objective function. We also formulate FIFO constraints in terms of the relationship between link cumulative inflows and outflows and the link entry time. Optimization models that integrate the proposed FIFO constraints into the proposed LP formulations for SO-DTA problems without FIFO constraints are also developed to formulate SO-DTA problems with FIFO constraints. Based on the properties of the proposed optimization problems, branch-and-bound algorithms are developed to solve SO-DTA problems with FIFO constraints. Two methods are developed to identify FIFO violations in feasible flow patterns and to design a branching scheme for the proposed branch-and-bound algorithms. Finally, numerical examples are set up to demonstrate the properties of the proposed models and the performance of the algorithms.

Key words: Dynamic traffic assignment; system optimum; link transmission model; branch-and-bound algorithm; vehicle holding problem; first-in-first-out

1. Introduction

A system optimal (SO) static traffic assignment problem is aimed at predicting an optimal traffic pattern that minimizes the total system travel time (TSTT) spent by all travelers in the network. This problem can serve as a benchmark for evaluating the performance of a transportation system (Ma et al. 2014) and can be formulated as a mathematical program with a convex solution set that can be easily solved. The resultant TSTT can be compared with that obtained by solving the

corresponding user equilibrium (UE) static traffic assignment problem, which can be formulated as a mathematical program or a variational inequality (VI) problem with a convex solution set. However, for the dynamic extensions of SO and UE static traffic assignment problems (i.e., dynamic traffic assignment (DTA) problems), their solution sets are always non-convex, mainly because they often capture desirable properties to make the solutions consistent with actual traffic behavior. These properties include queue spillback (e.g., Daganzo 1995; Lo and Szeto 2002; Ramadurai and Ukkusuri 2010; Ukkusuri et al. 2012; Ma et al. 2014; Long et al. 2015a, 2016; Jiang et al. 2016), first-in-first-out (FIFO) (e.g., Carey 1992; Long et al. 2011; Carey et al. 2014), and non-vehicle holding (NVH) (e.g., Shen et al. 2007; Zheng and Chiu 2011; Zhu and Ukkusuri 2013). Queue spillback refers to the queue spilling over to its upstream links. FIFO states that vehicles exit from the link in the same order they entered it. Vehicle holding implies that traffic flows are reluctant to move forward from upstream links to their downstream links even if vacant spaces are available on the downstream links. In DTA models, capturing queue spillback can lead to the non-existence of UE solutions (Szeto and Lo 2006); the FIFO requirement can yield a nonconvex constraint set (Carey 1992), and the NVH requirement can introduce discrete decision variables, making the resultant problem more difficult to solve efficiently. Developing SO-DTA models for the benchmarking that can both capture actual traffic behavior and be solved efficiently is thus challenging.

According to the available route and departure time choices, SO-DTA problems can be classified into three categories: (1) pure departure time choice problems (e.g., Vickrey 1969; Liu et al. 2015), (2) pure route choice problems (e.g., Merchant and Nemhauser 1978a,b; Ho 1980; Ziliaskopoulos 2000; Ukkusuri and Waller 2008; Nie 2011; Zheng and Chiu 2011; Zhu and Ukkusuri 2013; Zheng et al. 2015), and (3) simultaneous route and departure time choice (SRDTC) problems (e.g., Chow 2009; Doan and Ukkusuri 2012; Qian et al. 2012; Ma et al. 2014, 2017). The first two categories are special cases of the last. In pure departure time choice problems, travelers' route choices are not explicitly considered and only one route is modeled for each origin-destination (OD) pair (e.g.,

Vickrey 1969; Liu et al. 2015). The pure route choice problems assume that travelers' departure times are fixed. These problems have received more attention than the other two.

SO-DTA problems can be modeled based on two approaches: formulating them as (1) standard UE-DTA problems and (2) mathematical programming problems. The first approach commonly uses marginal path travel times to formulate SO-DTA problems as standard UE-DTA problems, and any solution algorithms developed for UE-DTA problems (e.g., Friesz et al. 1993; Lo and Szeto 2002; Perakis and Roels 2006; Long et al. 2013b; Guo et al. 2018) can thus be used to solve the SO-DTA problems. The key step in this approach is to evaluate marginal path travel times. However, when realistic traffic flow models are used, marginal path travel times are obtained through a procedure called dynamic network loading (DNL), and they have a complex dependence on path flows. The evaluation of marginal path travel times is thus notoriously difficult in general networks (Qian et al. 2012).

The second approach typically formulates SO-DTA problems as mathematical programming problems. The ease with which the resultant models can be solved is highly dependent on the underlying DNL models, such as point queue models (e.g., Han 2003; Ban et al. 2012; Han et al. 2013a,b; Bliemer et al., 2014; Jin 2015; Long et al. 2015b), exit flow models (e.g., Merchant and Nemhauser 1978a,b; Carey 1987; Carey and Srinivasan 1993), and kinematic wave models (e.g., Daganzo 1995; Yperman 2007; Ziliaskopoulos 2000; Ukkusuri and Waller 2008; Han et al. 2015a,b, 2016b). The exit flow models treat the outflow of a link or a segment of a link as a non-decreasing function of the number of vehicles on the whole link or link segment, respectively. SO-DTA problems that incorporate exit flow functions are usually formulated as nonlinear and nonconvex mathematical programming problems (e.g., Merchant and Nemhauser 1978a,b) that are difficult to solve. The kinematic wave models are based on either the solution scheme of Daganzo (1995) (i.e., the cell transmission model [CTM]) or that of Newell (1993) for the Lighthill and Whitham (1955) and Richards (1956) hydrodynamic model of traffic flow. The main difference between these models and exit flow models is that kinematic wave models consider storage capacity to capture the effects of physical queues such as queue spillback. In addition, SO-DTA models

that incorporate kinematic wave models can lead to a linear programming (LP) formulation, which makes the formulation computationally efficient and solvable for a reasonably sized network (Zhu and Ukkusuri 2013).

In existing SO-DTA models, FIFO is seldom considered in multi-destination networks. FIFO indicates that vehicles that enter a link earlier will leave it sooner (Lo and Szeto 2002; Long et al. 2011; Carey et al. 2014). The FIFO condition can be used to rule out traffic overtaking in traffic congestion or traffic jams. For a congested road, FIFO violations imply that faster vehicles are allowed to "jump over" the preceding slower vehicles. This is unrealistic, as in reality, the faster vehicles are slowed down by the slower vehicles ahead (Carey 2004). FIFO assumes that traffic of different types that enters the same link at approximately the same time usually travels at the same speed, and FIFO means traffic that enters an arc first will *on average* exit first (Carey 1992, 2004). Note that it is easily misunderstood that FIFO means vehicles should not overtake and pass each other on a link, as they do in practice. FIFO considers the average behavior of aggregate traffic flow; the FIFO solution can be interpreted as including localized overtaking (Carey 2004). If we wish to capture (non-localized) traffic overtaking in a DTA or traffic flow model, different types of drivers with different characteristics, such as speed, together with the FIFO condition for each type of driver can be introduced into these models.

The FIFO requirement can yield a nonconvex constraint set in DTA models, especially if they include multiple destinations or commodities (Carey 1992). Explicitly imposing this category of constraints may increase the complexity of the DTA models; hence, the applicability of most SO-DTA models is restricted in theory to the single-destination, single-commodity problem. In the literature, UE-DTA models tend to implicitly guarantee FIFO by using a proper travel time model that satisfies this property (e.g., Friesz et al. 1993; Lo and Szeto 2002; Long et al. 2013b). For example, Friesz et al. (1993) proposed an affine link travel time model with respect to link volume that satisfies strong FIFO, which was further used in their UE-DTA model. Unlike their model, the UE-DTA models proposed by Lo and Szeto (2002) and Long et al. (2013b) used discretised travel time models based on cumulative flows, in which link/route travel times were defined as the average

travel times for vehicles entering the link/route during the concerned interval and had been proved to satisfy weak FIFO (Long et al. 2011, 2013a). To the best of our knowledge, nonconvex FIFO constraints are rarely incorporated explicitly into SO-DTA models; most existing SO-DTA models are only developed for single-destination, single-commodity traffic networks (e.g., Merchant and Nemhauser 1978a,b; Ziliaskopoulos 2000; Zheng and Chiu 2011) or multi-destination networks without an explicit consideration of the FIFO requirement (e.g., Zhu and Ukkusuri 2013; Doan and Ukkusuri 2015).

The vehicle hold problem has received more attention than FIFO. This problem often occurs due to relaxation (i.e., replacing nonlinear equality constraints with inequality constraints) or linearization, and is a common problem for SO-DTA models (e.g., Merchant and Nemhauser 1978a; Ziliaskopoulos 2000). This cannot be found in a static SO assignment. Unless all vehicles are controlled by a central computer or smart traffic signal systems can give red signals to all vehicles that require holding, vehicle holding represents an unrealistic traffic flow phenomenon and should be completely eliminated. The methods used to address the vehicle holding problem can be classified into three categories: (a) the addition of a penalty term into the objective function (e.g., Lin and Wang 2004; Zhu and Ukkusuri 2013), (b) the introduction of a set of mixed-integer linear inequality constraints (e.g., Lo 1999; Pavlis and Recker 2009; Han et al. 2014, 2016a), and (c) a successive linear optimization approach (e.g., Ho 1980; Nie 2011). The first category of methods only solves the mathematical program once and is very efficient; however, the coefficient of the additional term must be carefully set. The second category formulates the NVH constraints as linear mixed-integer constraints and hence is not very efficient because it requires the solution of mixed-integer problems. The third category eliminates vehicle holding via a successive readjustment scheme that is realized by solving a series of LP problems and hence is less efficient than the first category. These methods do not consider the FIFO requirement, and no adequate method for addressing the vehicle holding problem under the FIFO requirement has been found.

In this paper, we introduce the link transmission model (LTM) of Yperman (2007) to formulate various SO-DTA problems with and without consideration of the FIFO and NVH requirements.

Unlike the LTM-based SO-DTA models developed by Long et al. (2018) and Ngoduy et al. (2016) that only considered networks with a single destination and an OD pair, respectively, in our proposed models we address the network with multiple origins and destinations. Our study also has a different focus from those of Long et al. (2018) and Ngoduy et al. (2016). Long et al. (2018) used the LTM to develop SO-DTA models with environmental objectives. Ngoduy et al. (2016) adopted an extension of the LTM to develop SO-DTA models to find DSO solutions that optimally distributed the congestion over links inside the network, which essentially eliminated avoidable queue spillbacks. Our paper focuses on modeling FIFO in a general network with the consideration of minimization of TSTT, unlike the single-destination SO-DTA models (e.g., Ziliaskopoulos 2000; Zheng and Chiu 2011), in which traffic flow satisfies FIFO by nature, and the LTM-based multiple-OD SO-DTA model proposed by Levin (2017), in which no FIFO is considered. We also propose branch-and-bound algorithms to solve SO-DTA problems with the consideration of FIFO.

The SO-DTA models with destination-based decision variables have fewer decision variables than those with OD-based decision variables and hence can be solved more efficiently. As with UE-DTA problems, SO-DTA problems can be formulated as either link- or path-based models. Although the outcomes of link-based DTA models are time-dependent link flows rather than route flows, these models do consider route choice behavior (i.e., the models are developed based on the assumption that travelers select their routes to minimize the total system travel cost). Path-based models can obtain path-related information, such as path flows and the path set, to model traffic at diverges and merges. Path-based models can, therefore, track spillback queues more easily when a realistic traffic flow model is used. However, a significant disadvantage of path-based models is that they require either a time-consuming path enumeration procedure or a path set generation heuristic in the solution procedure. The link-based models proposed in this paper can avoid the path enumeration or path set generation step in the solution procedure, but still model queue spillback.

This study makes the following contributions.

First, we introduce the LTM to formulate SO-DTA problems over a general network in terms of cumulative flows for general network applications (with multiple origins and destinations). To the best of our knowledge, our study is the first to do this using the concepts of the LTM and *FIFO*. Unlike single-destination SO-DTA models, explicitly capturing FIFO in the resultant formulations for general networks is not straightforward. Our results verify that our proposed LTM-based SO-DTA models outperform CTM-based models in terms of computational efficiency, and maintain the same level of accuracy in terms of guaranteeing the same total system travel time in general networks.

Second, we develop a novel method to address the vehicle holding problem in SO-DTA problems without FIFO constraints. This method requires only the introduction of an additional product of a sufficiently small coefficient and the sum of cumulative link outflows into the objective function of the link-based formulation. The proposed method is only required to solve one LP problem. Compared with other methods (e.g., Lin and Wang 2004; Zhu and Ukkusuri 2013), the parameter of the penalty term is easier to calibrate. We prove that the proposed model can obtain an NVH-SO flow pattern when the coefficient of the penalty term is positive and sufficiently small.

Third, we propose a definition of the FIFO condition for multi-destination traffic networks, and provide two nonconvex mixed-integer formulations for SO-DTA problems with FIFO constraints: one with NVH consideration and the other without. Two methods are developed for the identification of FIFO violations in any flow patterns.

Fourth, two branch-and-bound algorithms are developed to solve both SO-DTA problems with FIFO constraints, which can achieve global optimal solutions. The proposed algorithms are based on the properties of the proposed optimization problems. These algorithms only need to solve LP or MILP sub-problems to evaluate the branches, generate new branches based on the two critical link entry times obtained from the FIFO violation identification procedure, update both the lower and upper bounds of the TSTT, and fathom the unnecessary branches.

The remainder of this paper is organized as follows. In the next section, three link-based SO-DTA

problems without FIFO constraints are formulated as LP problems based on three different objectives. The two link-based SO-DTA problems with FIFO constraints are formulated as nonconvex optimization problems in Section 3. In Section 4, branch-and-bound algorithms are developed to solve SO-DTA problems with FIFO constraints. Numerical examples are given in Section 5, and conclusions are provided in Section 6.

2. SO-DTA problems without FIFO constraints

2.1. Notations

We consider a network $G(N, A)$ with multiple origins and destinations, where N and A denote the sets of nodes and links, respectively. $A(i)$ is the set of links whose tail node is i, and $B(i)$ is the set of links whose head node is i. R and S denote the sets of origin and destination nodes, respectively. The network has three types of links: source, destination, and general links. Each source (destination) link connects to only one origin (destination) within the network, and each source (destination) connects to only one origin (destination) link. Let A_R and A_S denote the sets of source and destination links, respectively. Both source and destination links are dummy links. All source and destination links have infinite inflow and storage capacities. Similar to the cell-based SO-DTA problems (e.g., Ziliaskopoulos 2000; Ukkusuri and Waller 2008; Zhu and Ukkusuri 2013), we assume that the outflow capacity of each destination link is zero, and vehicles finally arrive at destination links and remain there. We discretize the period T of interest into a finite set of time intervals $K = \{k = 1, 2, \cdots, \underline{K}\}$. Let δ be the interval length such that $\delta\underline{K} = T$. Without loss of generality, we let $\delta = 1$. Let Φ be the set of index pairs $\{(a, k) : a \in A, k \in K\}$. The following notations are adopted throughout this paper.

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

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

$D_a^s(k)$ cumulative demand between the entry of origin link a and destination s at the end of interval k

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

$U_a^s(k)$ cumulative number of vehicles that enter link a to destination s by the end of interval k

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

$V_a^s(k)$ cumulative number of vehicles that leave link a to destination s by the end of interval k

$\mathbf{U}$ the vector of cumulative inflows $\mathbf{U} = [U_a^s(k), a \in A, s \in S, k \in K]$

$\mathbf{V}$ the vector of cumulative outflows $\mathbf{V} = [V_a^s(k), a \in A, s \in S, k \in K]$

$\mathbf{x}$ the vector of a flow pattern $\mathbf{x} = [\mathbf{U}, \mathbf{V}]$

In this study, cumulative flows are defined at integer time instants. Cumulative flows at non-integer time instants are also required to formulate the FIFO constraints, as link entry links are used to define FIFO and can be non-integer time instants. Therefore, following Yperman (2007) and Long et al. (2011), a linear interpolation procedure is applied to express the cumulative flows at non-integer time instants in terms of the cumulative flows at integer time instants. For example, the cumulative flows of link a at time instant $k+\mu$ can be formulated as follows:

$$\begin{cases} U_a^s(k+\mu) = (1-\mu)U_a^s(k) + \mu U_a^s(k+1), \\ V_a^s(k+\mu) = (1-\mu)V_a^s(k) + \mu V_a^s(k+1). \end{cases} \tag{1}$$

Using the linear interpolation procedure (1), cumulative flows at non-integer time instants can be expressed as the cumulative flows at integer time instants. To simplify the proposed model formulations, the cumulative flows at non-integer time instants are used to describe the FIFO constraints.

2.2. An overview of the link transmission model

The LTM uses a triangular fundamental diagram (Yperman 2007) and is defined by three parameters: a fixed free-flow speed (v), a maximum flow or capacity (q_{max}), and a jam density (ρ_{jam}). Both the critical density ρ_{crit} and the backward shock-wave speed w can be derived by the three parameters, that is, $\rho_{crit} = q_{max}/v$ and $w = q_{max}v/(q_{max} - \rho_{jam}v)$. The LTM uses Newell's simplified method to determine sending and receiving flows, given as follows (Yperman 2007):

$$S_a(k) = \min\{U_a(k - \overline{\tau}_a) - V_a(k-1), C_a(k)\} \text{ and} \tag{2}$$

$$R_a(k) = \min\{V_a(k-\overline{\iota}_a) + L_a\rho_{jam} - U_a(k-1), Q_a(k)\}, \tag{3}$$

where L_a is the length of link a, and $\bar{\tau}_a$ and $\bar{\iota}_a$ are the free-flow travel time of vehicles on link a and the travel time required by the backward shock-wave from the exit to the entry of link a, respectively.

For each link, its inflow and outflow during interval k should be restricted by its sending and receiving flows, respectively, during this interval. Hence, we have

$$U_a(k) - U_a(k-1) \le R_a(k) \text{ and } V_a(k) - V_a(k-1) \le S_a(k), \forall a \in A, k \in K. \tag{4}$$

Substituting Eqs. (2) and (3) into the system of inequality (4), we obtain the following system of linear LTM-based flow constraints:

$$\begin{cases} V_a(k) \le U_a(k-\overline{\tau}_a), & \forall a \in A, k \in K, \\ V_a(k) - V_a(k-1) \le C_a(k), & \forall a \in A, k \in K, \\ U_a(k) \le V_a(k-\overline{\iota}_a) + L_a\rho_{jam}, & \forall a \in A, k \in K, \\ U_a(k) - U_a(k-1) \le Q_a(k), & \forall a \in A, , k \in K. \end{cases} \tag{5}$$

Based on the definition of cumulative flows, we have

$$U_a(k) = \sum_{s \in S} U_a^s(k) \text{ and } V_a(k) = \sum_{s \in S} V_a^s(k). \tag{6}$$

Substituting Eq. (6) into the system of inequality (5), we have

$$\begin{cases} \sum\limits_{s \in S} V_a^s(k) \le \sum\limits_{s \in S} U_a^s(k-\overline{\tau}_a), & \forall a \in A, k \in K, \\ \sum\limits_{s \in S} [V_a^s(k) - V_a^s(k-1)] \le C_a(k), & \forall a \in A, k \in K, \\ \sum\limits_{s \in S} U_a^s(k) \le \sum\limits_{s \in S} V_a^s(k-\overline{\iota}_a) + L_a\rho_{jam}, & \forall a \in A, k \in K, \\ \sum\limits_{s \in S} [U_a^s(k) - U_a^s(k-1)] \le Q_a(k), & \forall a \in A, k \in K. \end{cases} \tag{7}$$

Any cumulative outflow disaggregated by destination should also be constrained by the boundary condition at the upstream end of the link. Hence, we have

$$V_a^s(k) \le U_a^s(k-\overline{\tau}_a), \forall a \in A, s \in S, k \in K. \tag{8}$$

Note that the third inequality of (7) implies that $U_a^s(k) \le V_a^s(k-\overline{\iota}_a) + L_a\rho_{jam}$ is satisfied for all $a \in A$, $s \in S$, and $k \in K$ due to the non-decreasing property of cumulative flows and condition (8). This statement can be proved by contradiction. The proof is straightforward and hence is omitted.

Traffic flows in the LTM should also satisfy the FIFO, flow conservation, and definitional constraints. In this section, we do not consider the FIFO constraints, but they are formulated in the next section. The flow conservation constraints require that the flow that enters any node (except the destination node), together with the demand generated at that node, must exit from that node.

In our study, we assume that source links are dummy links, and all source links have infinite inflow and storage capacities. Hence, the entrance of a source link will never become congested and can always accommodate the demand. Therefore, the cumulative inflows of source links equal the cumulative demands. Equivalently, we have

$$U_a^s(k) = D_a^s(k),\ \forall a \in A_R, s \in S, k \in K. \tag{9}$$

Unlike the CTM that accounts for only simple merge and diverge intersections, we use a general node model to describe traffic transmission from link to link. General nodes do not generate traffic demand, so we have the following flow conservation constraints:

$$\sum_{a \in B(i)} V_a^s(k) = \sum_{b \in A(i)} U_b^s(k),\ \forall i \in N \backslash \{R, S\}, s \in S, k \in K. \tag{10}$$

Definitional constraints are used to describe the initial conditions and nonnegative and non-decreasing properties of cumulative flows. In our model, we use the following constraints:

$$U_a^s(k) - U_a^s(k-1) \geq 0,\ \forall a \in A, s \in S, k \in K, \tag{11}$$

$$V_a^s(k) - V_a^s(k-1) \geq 0,\ \forall a \in A, s \in S, k \in K, \text{ and} \tag{12}$$

$$U_a^s(0) = V_a^s(0) = 0,\ \forall a \in A, s \in S. \tag{13}$$

Constraints (11) and (12) imply that the cumulative flows are nonnegative, and constraint (13) implies that the cumulative flows initially equal zero.

Definition 1 (Basic feasible flow set): Constraints (7)-(13) form a basic feasible flow set for the LTM-based SO-DTA problem. The set is formulated as follows:

$$\Omega = \{\mathbf{x} | \text{Subject to constraints } \ (7) - (13)\}. \tag{14}$$

In addition to the basic constraints (7)-(13), NVH constraints should be developed to address the vehicle holding problem in SO-DTA problems. In Appendix EC.1, we provide a definition of vehicle holding and NVH flow patterns, and equivalently formulate the NVH conditions as mixed-integer constraints (Pavlis and Recker 2009).

2.3. The SO-DTA models without FIFO constraints

2.3.1. LP formulation for the relaxed SO-DTA problem

Following Ziliaskopoulos (2000), the objective of the LTM-based SO-DTA problem with multiple destinations is to minimize the TSTT in the network, which equals the number of vehicles on all links during the modeling horizon. Therefore, we can formulate the relaxed SO-DTA (R-SO-DTA) problem, which considers neither FIFO constraints nor NVH constraints, as the following LP problem:

$$\min_{\mathbf{x}\in\Omega} \eta = \sum_{a\in A\setminus A_S} \sum_{s\in S} \sum_{k\in K} \Big[U_a^s(k) - V_a^s(k) \Big]. \tag{15}$$

Lemma 1. LP problem (15) can be equivalently formulated as the following LP problem:

$$\max_{\mathbf{x}\in\Omega} \varpi = \sum_{a\in A_S} \sum_{s\in S} \sum_{k\in K} U_a^s(k). \tag{16}$$

The proof is given in Appendix EC.2.1.

2.3.2. LP formulation for the NVH-SO-DTA problem

Lin and Wang (2004) addressed the vehicle holding problem by adding a penalty term in the objective function of the LP formulation of Ziliaskopoulos (2000). However, they did not prove that such a term always removed vehicle holding flows, nor did they provide details about setting the coefficient associated with this term. Similar to the method proposed by Lin and Wang (2004), Zhu and Ukkusuri (2013) introduced a penalty term, the sum of weighted cell flows, to the objective function of the LP formulation of Ziliaskopoulos (2000), but they did not provide details about setting the weight associated with the penalty term for *cyclic* networks. In this study, we propose the following LP problem to address the vehicle holding problem:

Proposition 1: For a given $\kappa > 0$, let $\mathbf{x}^*$ be an optimal solution to the following LP problem:

$$\max_{\mathbf{x}\in\Omega} \tilde{\varpi} = \sum_{a\in A_S}\sum_{s\in S}\sum_{k\in K} U_a^s(k) + \kappa \sum_{a\in A}\sum_{s\in S}\sum_{k\in K} V_a^s(k). \tag{17}$$

If $\mathbf{x}^*$ is an optimal solution to LP problem (15), then $\mathbf{x}^*$ is also an NVH system optimal (NVH-SO) flow pattern, and all optimal solutions to the following LP problem are also NVH-SO flow patterns:

$$\max_{\mathbf{x}\in\Omega} \tilde{\varpi} = \sum_{a\in A_S}\sum_{s\in S}\sum_{k\in K} U_a^s(k) + \tilde{\kappa} \sum_{a\in A}\sum_{s\in S}\sum_{k\in K} V_a^s(k), \tag{18}$$

where $0 < \tilde{\kappa} < \kappa$.

The proof is given in Appendix EC.2.2. The results presented in Proposition 1 indicate that we can select a sufficiently small positive parameter κ and solve only one LP problem to obtain an NVH-SO flow pattern.

Maximizing the sum of cumulative link outflows can encourage the development of cyclic flows in cyclic networks, and hence an optimal solution to LP problem (17) may contain many cyclic flows for cyclic networks, as shown later in our numerical example. We develop the LP problem in Appendix EC.3 to eliminate unnecessary cyclic flows in cyclic networks and to obtain system optimal flows with minimum total system travel distance (MTSTD).

3. SO-DTA problems with FIFO constraints

3.1. Existing FIFO conditions and FIFO flow patterns

Kinematic wave models, including the LTM, are known to not satisfy strong FIFO (Lo and Szeto 2002; Szeto and Lo 2006). Hence, we only focus on weak FIFO. In the literature, weak link FIFO is defined based on link travel times as follows (Carey 2004):

Definition 2 (Existing weak link FIFO condition): Traffic flow on link a satisfies the FIFO condition if and only if

$$\ell' > \ell'' \Rightarrow \ell' + \tau_a(\ell') \geq \ell'' + \tau_a(\ell''), \forall \ell' \in [0,T], \ell'' \in [0,T], \tag{19}$$

where $\tau_a(\ell')$ ($\tau_a(\ell'')$) is the travel time of the vehicles entering link a at time instant ℓ' (ℓ'').

Using the link travel times of vehicles that travel to each destination, we can obviously reformulate the weak link FIFO condition and define a FIFO flow pattern as follows:

Definition 3 (Reformulated weak link FIFO condition): Traffic flow on link a satisfies the FIFO condition if and only if

$$\ell' > \ell'' \Rightarrow \ell' + \tau_a^{s_1}(\ell') \geq \ell'' + \tau_a^{s_2}(\ell''), \forall s_1 \in S, s_2 \in S, \ell' \in [0,T], \ell'' \in [0,T], \tag{20}$$

where $\tau_a^{s_1}(\ell')$ ($\tau_a^{s_2}(\ell'')$) is the travel time of the vehicles entering link a to destination s_1 (s_2) at time instant ℓ' (ℓ'').

Our proposed weak link FIFO condition is defined based on cumulative link inflows and outflows. These can be related by link travel times as follows (Long et al. 2011; Carey et al. 2014):

$$U_a(\ell) = V_a(\ell + \tau_a(\ell)), \forall a \in A, \ell \in [0,T]. \tag{21}$$

$\tau_a(\ell)$ is unique when the cumulative link outflows strictly increase. However, if this is not the case, $\tau_a(\ell)$ may not be unique, and multiple values satisfying Eq. (21) may exist. For any given time instant, if the cumulative outflows of a link that belongs to a destination do not strictly increase at this time instant, the corresponding outflows should be zero and the traffic flows to that destination at this time instant satisfy weak FIFO by nature. Therefore, we can ignore the situation in which the cumulative link outflows do not strictly increase when the weak FIFO requirement is considered in the SO-DTA problems.

Similar to the relationship (21), the cumulative link inflows and outflows disaggregated by destinations can also be related by the link travel times of vehicles that travel to each destination as follows:

$$U_a^s(\ell) = V_a^s(\ell + \tau_a^s(\ell)), \forall a \in A, \ell \in [0,T], s \in S. \tag{22}$$

Definition 4 (FIFO flow pattern): A flow pattern $\mathbf{x} \in \Omega$ is a FIFO flow pattern if a vector of link travel times $\tau = [\tau_a^s(\ell)]$ exists such that condition (22) is satisfied, and τ satisfies link FIFO condition (20) for all $a \in A$.

3.2. Identifying FIFO violations

The following two propositions can be used to identify FIFO violations; each allows us to develop a method of FIFO violation identification.

Proposition 2: For a given flow pattern $\mathbf{x} \in \Omega$ and an index pair $(a,k) \in \Phi$, if at least one destination s exists such that $U_a^s(\underline{\ell}_{a,k}^*) < V_a^s(k)$, where $\underline{\ell}_{a,k}^*$ is a critical time instant (i.e., the earliest entry time of vehicles that leave link a at the end of interval k, which may not be an integer) and

$$\underline{\ell}_{a,k}^* = \arg\max_{\tilde{\ell} \le k - \overline{\tau}_a} \{U_a^s(\tilde{\ell}) \le V_a^s(k),\ \forall s \in S\}, \tag{23}$$

then $\mathbf{x}$ involves FIFO violations.

The proof is given in Appendix EC.2.3.

Proposition 3: For a given flow pattern $\mathbf{x} \in \Omega$ and an index pair $(a,k) \in \Phi$, if at least one destination s exists such that $U_a^s(\overline{\ell}_{a,k}^*) > V_a^s(k)$, where $\overline{\ell}_{a,k}^*$ is a critical time instant (i.e., the latest entry time of vehicles that leave link a at the end of interval k, which may not be an integer) and

$$\overline{\ell}_{a,k}^* = \arg\min_{\tilde{\ell} \ge 0} \{U_a^s(\tilde{\ell}) \ge V_a^s(k),\ \forall s \in S\}, \tag{24}$$

then $\mathbf{x}$ involves FIFO violations.

The proof is similar to that of Proposition 2 and is thus omitted here. An example presented in Appendix EC.4 is developed to illustrate FIFO violations, and the procedure given in Appendix EC.5.1 is developed to identify these violations.

3.3. The proposed FIFO condition and an alternative FIFO flow pattern

Definition 5 (Link entry time): $p_a^s(k)$ is defined as the time instant at which vehicles enter link a going to destination s and leave the link at the end of interval k.

According to Definition 5, we have $k = p_a^s(k) + \tau_a^s(p_a^s(k))$, and can equivalently reformulate Eq. (22) as follows:

$$U_a^s(p_a^s(k)) = V_a^s(k), \forall a \in A, k \in K, s \in S. \tag{25}$$

When the cumulative link inflow curve is not strictly increasing, $p_a^s(k)$ may not be unique and can be multiple values. If multiple entry times for a given index pair (a, s, k) exist, then we have $U_a^s(p'^s_a(k)) = V_a^s(k) = U_a^s(p''^s_a(k))$, where $p'^s_a(k)$ and $p''^s_a(k)$ are, respectively, the minimum and maximum values of the entry times. The monotone increasing property of cumulative flows implies that the inflow of link a to destination s during period $(p'^s_a(k), p''^s_a(k))$ is zero. The traffic flows entering link a to destination s during period $(p'^s_a(k), p''^s_a(k))$ thus satisfy weak FIFO by nature, and the vehicles that enter the link at time instants $p'^s_a(k)$ and $p''^s_a(k)$ leave the link at the same time. Otherwise, the link entry time is unique.

Proposition 4: For any given $\mathbf{x} \in \Omega$, if $\mathbf{x}$ is a FIFO flow pattern, then there must exist a vector of entry times $\mathbf{p} = [p_a(k)]$ such that

$$V_a^s(k) = U_a^s(p_a(k)), \forall a \in A, k \in K, s \in S. \tag{26}$$

Proposition 2 can be used to prove this proposition, and the detailed proof is presented in Appendix EC.2.4. It can also be proved by using Proposition 3, which is omitted here.

According to Proposition 4, a FIFO flow pattern can also be defined as follows:

Definition 6 (Alternative FIFO flow pattern): A flow pattern $\mathbf{x} \in \Omega$ is a FIFO flow pattern if a vector of entry times $\mathbf{p} = [p_a(k)]$ exists such that condition (26) is satisfied.

Before solving SO-DTA problems with FIFO constraints, we do not know the exact value of the entry time $p_a(k)$ in condition (26). Hence, we cannot directly use the linear interpolation procedure (Eq. (1)) to reformulate condition (26), but instead use a linear combination of cumulative flows at the end of two adjacent time intervals. Therefore, according to Definition 6, condition (26) can be reformulated as follows:

$$V_a^s(k) = \sum_{l \le k - \overline{\tau}_a} \lambda_{a,k,l} U_a^s(l), \forall a \in A, k \in K, s \in S, \tag{27}$$

where $\lambda_{a,k,l}$ is the linear combination/interpolation coefficient and satisfies

$$\sum_{l \le k - \overline{\tau}_a} \lambda_{a,k,l} = 1, \ \forall a \in A, k \in K, \tag{28a}$$

$$\lambda_{a,k,l}\lambda_{a,k,l'} = 0,\ \forall a \in A, k \in K, l \le k - \overline{\tau}_a, l' \le l - 2, \text{and} \tag{28b}$$

$$\lambda_{a,k,l} \ge 0,\ \forall a \in A, k \in K, l \le k - \overline{\tau}_a, \tag{28c}$$

where (28a) is a definitional constraint that requires the sum of linear combination coefficients equal to one, (28b) ensures that at most two adjacent intervals are required to represent the linear combination due to linear interpolation, and (28c) ensures that the linear combination coefficients are nonnegative. We note that FIFO constraints (27) and (28) are developed for general links, and hence FIFO also holds for diverging and merging links if FIFO constraints are satisfied.

Proposition 5: If a vector $\lambda = [\lambda_{a,k,l}]$ exists such that a feasible flow pattern $\mathbf{x} \in \Omega$ satisfies constraints (27) and (28), then the vector $\mathbf{p} = [p_a(k)] = [\sum_{l \le k - \overline{\tau}_a} l\lambda_{a,k,l}]$ satisfies FIFO condition (26) and $\mathbf{x}$ is a FIFO flow pattern.

The proof is given in Appendix EC.2.5.

3.4. The FIFO-SO-DTA model

Integrating FIFO constraints (27) and (28) into LP problem (15), we can obtain the following FIFO-SO-DTA model:

$$\min_{\mathbf{x} \in \Omega, \lambda} \eta = \sum_{a \in A \setminus A_s} \sum_{s \in S} \sum_{k \in K} \Big[U_a^s(k) - V_a^s(k) \Big]. \tag{29}$$

Subject to constraints (27) and (28),

where the vector $\lambda = [\lambda_{a,k,l}]$.

In the preceding optimization problem, constraints (27) and (28b) contain bilinear terms, and the set of feasible solutions of this problem is nonconvex. The number of elements in the vector $\lambda = [\lambda_{a,k,l}]$ is $|A \setminus A_s||K|^2$ and can be large even when the number of time intervals $|K|$ is not very large, as the number depends on $|K|^2$. Directly solving the FIFO-SO-DTA model with traditional solution methods for nonconvex mathematical programming problems is therefore extremely challenging.

Based on the following theorem, we develop a branch-and-bound algorithm to solve the FIFO-SO-DTA model.

Theorem 1: Let $[\mathbf{x}^*, \lambda^*]$ be an optimal solution to FIFO-SO-DTA problem (29), and $\mathbf{p}^* = [p_a^*(k)] = [\sum_{l \le k-\overline{\tau}_a} l\lambda_{a,k,l}^*]$. Then, $\mathbf{x}^*$ is also an optimal solution to the following LP:

$$\min_{\mathbf{x}\in\Omega} \eta = \sum_{a\in A\setminus A_S} \sum_{s\in S} \sum_{k\in K} \Big[U_a^s(k) - V_a^s(k)\Big]. \tag{30a}$$

$$\text{Subject to } V_a^s(k) = U_a^s(p_a^*(k)), \forall a \in A, k \in K, s \in S. \tag{30b}$$

The proof is given in Appendix EC.2.6.

Remark 1: Using the linear interpolation in Eq. (1), constraint (30b) immediately becomes a linear constraint in terms of $\mathbf{x}$. $p_a^*(k)$ may not be an integer. Let $\ell^* = [p_a^*(k)]$ and $\lambda^* = p_a^*(k) - \ell^*$. If $p_a^*(k)$ is not an integer, according to Eq. (1), constraint (30b) can be equivalently formulated as $V_a^s(k) = U_a^s(p_a^*(k)) = U_a^s(\ell^* + \lambda^*) = (1-\lambda^*)U_a^s(\ell^*) + \lambda^* U_a^s(\ell^*+1)$. Otherwise, $p_a^*(k)$ is an integer and $\lambda^* = 0$, and then we have $V_a^s(k) = U_a^s(\ell^*)$. Again, constraint (30b) is also linear.

3.5. The NVH-FIFO-SO-DTA model

Integrating NVH constraint (EC.3) and FIFO constraints (27) and (28) into LP problem (15), we can obtain the following NVH-FIFO-SO-DTA model:

$$\min_{\mathbf{x}\in\Omega,\lambda,\theta} \eta = \sum_{a\in A\setminus A_s} \sum_{s\in S} \sum_{k\in K} \Big[U_a^s(k) - V_a^s(k)\Big]. \tag{31}$$

$$\text{Subject to constraints (27)-(28) and constraint (EC.3) for all } a \in A, k \in K,$$

where the vector $\theta = [\theta_a^i(k)]$ and $\theta_a^i(k)$ is used to define NVH condition (EC.3).

Theorem 2: Let $[\mathbf{x}^*, \lambda^*, \theta^*]$ be an optimal solution to NVH-FIFO-SO-DTA problem (31), and $\mathbf{p}^* = [p_a^*(k)] = [\sum_{l \le k-\overline{\tau}_a} l\lambda_{a,k,l}^*]$. Then, $[\mathbf{x}^*, \theta^*]$ is also an optimal solution to the following mixed-integer linear programming (MILP) problem:

$$\min_{\mathbf{x}\in\Omega,\theta} \eta = \sum_{a\in A\setminus A_S} \sum_{s\in S} \sum_{k\in K} \Big[U_a^s(k) - V_a^s(k)\Big].$$

$$\text{Subject to } V_a^s(k) = U_a^s(p_a^*(k)), \forall a \in A, k \in K, s \in S \text{ and}$$

$$\text{constraint (EC.3) for all } a \in A, k \in K.$$

The proof is similar to that of Theorem 1 and is omitted here.

Proposition 6: Let η_1^*, η_2^*, η_3^*, and η_4^* be the optimal TSTT of the R-SO-, the NVH-SO-, the FIFO-SO-, and the NVH-FIFO-SO-DTA problems, respectively. Then, $\eta_1^* = \eta_2^* \leq \eta_3^* \leq \eta_4^*$.

The proof is given in Appendix EC.2.7.

4. Branch-and-bound algorithms for SO-DTA models with FIFO constraints

4.1. Basic concept of branch-and-bound algorithms

Based on Theorem 1, an optimal solution to the FIFO-SO-DTA problem can be obtained by solving the R-SO-DTA problem with an additional linear constraint (30b). However, the vector of entry times $\mathbf{p}^*$ in Eq. (30b) is unknown beforehand, and hence the challenge is to develop methods to determine the active constraints to be incorporated into the R-SO-DTA problem, thus ensuring that the resultant solution satisfies FIFO. Our method is to develop a branch-and-bound algorithm to search for proper ranges of entry times, using the set of pairs of link and interval indices of the flow vector with FIFO violations identified by the conditions stated in Propositions 2 and 3. For each branch in the algorithm, we need only solve one LP sub-problem to evaluate the branch, generate new branches based on the two critical entry times obtained from the FIFO violation identification procedure, update both the lower and upper bounds of the TSTT, and fathom the unnecessary branches.

Compared with the FIFO-SO-DTA problem, the NVH-FIFO-SO-DTA problem contains extra constraints, that is, NVH constraints. Based on Theorem 2, only the active NVH constraints are required to solve the NVH-FIFO-SO-DTA problem. Using vehicle holding condition (44), a procedure in which a series of MILP sub-problems is solved is developed to retrieve the active NVH constraints. The proposed branch-and-bound algorithm for the FIFO-SO-DTA problem can then be directly extended to solve the NVH-FIFO-SO-DTA problem. For each branch in the branch-and-bound algorithm for the FIFO-SO-DTA problem, only one LP sub-problem is solved to obtain

a solution. However, a series of MILP sub-problems are solved in each branch in the algorithm for the NVH-FIFO-SO-DTA problem.

4.2. A branch-and-bound algorithm for FIFO-SO-DTA problems

Definition 7 (Branch node). Each node of the branch-and-bound tree is denoted by $n(a,k,\underline{\ell},\overline{\ell})$, where a and k are the link and time interval concerned, respectively; and $\underline{\ell}$ and $\overline{\ell}$ are the lower and upper bounds of the entry time of link a corresponding to outflow interval k, respectively. The branch node $n(a,k,\underline{\ell},\overline{\ell})$ considers the following linear constraints on cumulative inflows and outflows:

$$U_a^s(\underline{\ell}) \le V_a^s(k) \le U_a^s(\overline{\ell}),\ \forall s \in S. \tag{32}$$

The cumulative flows are monotonically non-decreasing and$V_a^s(k) = U_a^s(p_a(k))$ for all $s \in S$ if FIFO is satisfied, so constraint (32) can restrict that $p_a(k)$ falls into the range $[\underline{\ell},\overline{\ell}]$.

4.2.1. Bounding scheme

In the proposed algorithm, the input sequence is used to add constraints to LP problem (15). Let UT be the set of unfathomed nodes in the search tree. For a given search tree node $\alpha \in UT$, the additional constraints on cumulative flows to the R-SO-DTA problem can be obtained from the set of nodes that belong to the input sequence of search tree node α, denoted as IS_α. The set of the additional constraints with respect to search tree node α can be represented by

$$\Omega_\alpha = \{\mathbf{x} | U_{a_\beta}^s(\underline{\ell}_\beta) \le V_{a_\beta}^s(k_\beta) \le U_{a_\beta}^s(\overline{\ell}_\beta), \forall s \in S, \beta \in IS_\alpha\},$$

where $\underline{\ell}_\beta$ and $\overline{\ell}_\beta$ are the lower and upper bounds of the entry time for search tree node β, respectively. α_β and k_β are the link and interval indexes for search tree node β, respectively. Adding the additional constraints for search tree node α into LP problem (15), we have the following:

$$\eta_\alpha = \min_{x \in \Omega \cap \Omega_\alpha} \eta = \sum_{a \in A \setminus A_S} \sum_{s \in S} \sum_{k \in K} \Big[U_a^s(k) - V_a^s(k) \Big], \tag{33}$$

where η_α is the optimal TSTT for search tree node α.

Proposition 7: If an optimal solution to LP problem (33), $\mathbf{x}^*$, is a FIFO flow pattern, then the corresponding vector of entry times $\mathbf{p}^* = [p_a^*(k)]$ satisfies the following condition:

$$\underline{\ell}_\beta \le p_{a_\beta}^*(k_\beta) \le \overline{\ell}_\beta, \forall \beta \in IS_\alpha. \tag{34}$$

The proof is given in Appendix EC.2.8.

Proposition 7 confirms that for branch node $n(a,k,\underline{\ell},\overline{\ell})$, constraint (32) can restrict that the entry time $p_a(k)$ falls into the range $[\underline{\ell},\overline{\ell}]$. By solving LP problem (33), the lower bound of search tree node α can be updated by $LB_\alpha = \eta_\alpha$.

The upper and lower bounds of the TSTT for the FIFO-SO-DTA problem can be derived from the node in the tree with the current best solution (the corresponding solution gives a FIFO flow pattern but may not give the smallest TSTT) and the unbranched search tree nodes, respectively:

$$UB = \min_{\alpha \in DT}\{LB_\alpha\} \text{ and} \tag{35}$$

$$LB = \min_{\alpha \in UT}\{LB_\alpha\}, \tag{36}$$

where DT is the set of nodes in the search tree with solutions that give FIFO flow patterns.

4.2.2. Branching strategy and scheme

In each iteration of the algorithm, a branching scheme is implemented for a selected search tree node α to generate new branches. Based on Propositions 2 and 3, we can determine the set of index pairs $\widetilde{\Phi}_\alpha$ with FIFO violations. For all $(a,k) \in \widetilde{\Phi}_\alpha$, the corresponding critical entry times $\underline{\ell}_{a,k}^*$ and $\overline{\ell}_{a,k}^*$ can be used to divide the feasible region of the FIFO-SO-DTA problem into three subregions by the following additional linear constraints:

$$U_a^s(\underline{\ell}_{a,k}^*) \le V_a^s(k) \le U_a^s(\overline{\ell}_{a,k}^*), \forall s \in S, \tag{37}$$

$$V_a^s(k) \le U_a^s(\underline{\ell}_{a,k}^*), \forall s \in S, \text{and} \tag{38}$$

$$V_a^s(k) \ge U_a^s(\overline{\ell}_{a,k}^*), \forall s \in S. \tag{39}$$

Proposition 8: An optimal solution to LP problem (33), $\mathbf{x}^*$, is also optimal to the following LP:

$$\min_{x\in\Omega\cap\Omega_\alpha} \eta = \sum_{a\in A\setminus A_S}\sum_{s\in S}\sum_{k\in K}\Big[U_a^s(k) - V_a^s(k)\Big]. \tag{40}$$

$$\text{Subject to } U_a^s(\underline{\ell}_{a,k}^*) \le V_a^s(k) \le U_a^s(\overline{\ell}_{a,k}^*),\ \forall (a,k)\in\widetilde{\Phi}\subseteq\widetilde{\Phi}(\mathbf{x}^*), s\in S. \tag{41}$$

The proof is given in Appendix EC.2.9.

According to Proposition 8, integrating constraint (37) into LP problem (33) forms a new LP problem (40)-(41), which has the same optimal objective value as the previous LP problem (33), and therefore the solution of the resultant LP problem (40)-(41) is not necessary (because it has been solved before). This implies that adding constraint (37) to LP problem (33) does not change the optimal solution(s) of the problem (33), and hence constraint (37) is a non-active FIFO constraint in general. Based on the definition of $\widetilde{\Phi}_\alpha$ and Propositions 2 and 3, for a given index pair $(a,k)\in\widetilde{\Phi}_\alpha$, FIFO violations occur on link a during interval k, and we have $U_a^s(\underline{\ell}_{a,k}^*) < V_a^s(k)$ or $U_a^s(\overline{\ell}_{a,k}^*) > V_a^s(k)$ for some $s\in S$. This implies that $\mathbf{x}^*$ does not satisfy either constraint (38) or (39). Therefore, adding either constraint (38) or (39) to LP problem (33) forms a new LP problem that must not have the same optimal solution as the initial LP problem (33), and hence they must be active FIFO constraints.

Constraints (38) and (39) are active FIFO constraints, so the solutions to the search nodes with either one of these constraints are more likely to be FIFO flow patterns. To quickly determine a feasible FIFO flow pattern, which gives an upper bound of the FIFO-SO-DTA problem, the search nodes corresponding to constraints (38) and (39) should be preferentially selected to branch. In the proposed scheme, we assign a large value of the priority parameter $\overline{p}$ to the branches associated with constraint (37) and a small value of the priority parameter $\underline{p}$ to the branches associated with constraints (38) and (39).

In the branch-and-bound algorithm, the unbranched search tree node with the minimum sum of the lower bound and the cumulative priority score is preferentially chosen as a branch node. The index of the branch node can be obtained from

$$\alpha = \arg\min_{\beta\in UT}\{LB_\beta + P_\beta\}, \tag{42}$$

where $P_\alpha = \sum_{\beta \in IS_\alpha} p_\beta$ and p_β is the priority parameter for search tree node β. If two or more search tree nodes have the same minimum sum, any one of them can be chosen as a branch node. In this study, we randomly select one of them.

The procedure presented in Appendix EC.5.2 is developed to branch a selected search tree node.

4.2.3. Fathoming scheme

In the proposed branch-and-bound method, a search tree node will be fathomed if its TSTT is not less than the current upper bound of the TSTT. The procedure presented in Appendix EC.5.3 is developed to fathom search tree nodes.

4.2.4. The overall branch-and-bound method

The above branching and bounding strategies are incorporated into the proposed branch-and-bound algorithm for the FIFO-SO-DTA problem, which is presented in Algorithm 1.

Algorithm 1 (The overall branch-and-bound algorithm for solving the FIFO-SO-DTA problem)

Initialize $UB \leftarrow +\infty$, $LB \leftarrow 0$, and $DT = \varnothing$, generate an initial search tree node α_0, set $IS_{\alpha_0} \leftarrow \varnothing$ and $UT \leftarrow \{\alpha_0\}$, and select the precision parameter value $\varepsilon > 0$.
while $UB - LB > \varepsilon$ **do**
  Choose branch node α by Eq. (42);
  Solve LP problem (33) and obtain an optimal solution $\mathbf{x}^*$ and the optimal TSTT η^*;
  Set $LB_\alpha \leftarrow \eta^*$, $\mathbf{x}_\alpha \leftarrow \mathbf{x}^*$;
  FIFO_BRANCHING(α,$\mathbf{x}_\alpha$);// Implement Procedure EC.2
  $LB \leftarrow \min_{\alpha' \in UT} \{LB_{\alpha'}\}$;// Update the lower bound
  $UB \leftarrow \min_{\alpha' \in DT} \{LB_{\alpha'}\}$;// Update the upper bound
  FATHOMING ().// Implement Procedure EC.3
end while
Set $\alpha = \arg\min_{\alpha' \in DT} \{LB_{\alpha'}\}$, and output an optimal solution $\mathbf{x}_\alpha$.

4.3. A branch-and-bound algorithm for the NVH-FIFO-SO DTA problem

By adding the additional FIFO constraints and the active NVH constraints of search tree node α to LP problem (15), we have the following MILP:

$$\min_{\mathbf{x} \in \Omega \bigcap \Omega_\alpha, \theta} \eta = \sum_{a \in A \setminus A_S} \sum_{s \in S} \sum_{k \in K} \Big[U_a^s(k) - V_a^s(k) \Big]. \tag{43}$$

Subject to constraint (EC.3) for all $(a,k) \in \Phi_\alpha$,

where Φ_α is the set of pairs of link and time interval indices associated with NVH conditions for search tree node α and $\Phi_\alpha \in \Phi$.

If the optimal solution to MILP problem (43) is not an NVH flow pattern, procedure EC.4, presented in Appendix EC.5.4, is developed to retrieve the active NVH constraints for search tree node α. Many integer decision variables are present in NVH constraints (EC.3), so if all index pairs $(a,k) \in (A \backslash A_S) \times K$ are considered, MILP sub-problem (43) cannot be solved efficiently. Procedure EC.4 obtains an NVH flow pattern by solving a series of small MILP sub-problems.

The proposed branch-and-bound algorithm for the NVH-FIFO-SO DTA problem is presented in Algorithm 2.

Algorithm 2 (The overall branch-and-bound algorithm for solving the NVH-FIFO-SO-DTA problem)

Initialize $UB \leftarrow +\infty$ and $DT = \varnothing$, generate an initial search tree node α_0, set $IS_{\alpha_0} \leftarrow \varnothing$ and $UT \leftarrow \{\alpha_0\}$, and select the precision parameter value $\varepsilon > 0$.
while $UB - LB > \varepsilon$ **do**
  Choose branch node α by Eq. (42);
  NVH_INDEXSET_GENERATION(α);// Implement Procedure EC.4
  FIFO_BRANCHING(α,$\mathbf{x}_\alpha$);// Implement Procedure EC.2
  $LB \leftarrow \min_{\alpha' \in UT} \{LB_{\alpha'}\}$;// Update the lower bound
  $UB \leftarrow \min_{\alpha' \in DT} \{LB_{\alpha'}\}$;// Update the upper bound
  FATHOMING().// Implement Procedure EC.3
end while
Set $\alpha = \arg\min_{\alpha' \in DT} \{LB_{\alpha'}\}$, and output an optimal solution $\mathbf{x}_\alpha$.

Remark 2: The overall branch-and-bound algorithm for solving the NVH-FIFO-SO-DTA problem is extended from that used to solve the FIFO-SO-DTA problem. The main difference is that Algorithm 2 additionally implements Procedure EC.4 before implementing Procedure EC.2, which ensures that the solution $\mathbf{x}_\alpha$ is an NVH flow pattern.

5. Numerical examples

This section presents four numerical experiments to illustrate the properties of the proposed SO-DTA models and the performance of the solution algorithms. All of the experiments were run on a

computer with an Intel Core 2 Quad Q9550 2.83-GHz CPU with 3.5 GB RAM. All of the LP and MILP problems were solved by the commercial software IBM ILOG CPLEX (version 12.5).

Example 1. Comparing the optimal solutions of models with and without NVH consideration.

In this example, we use the modified network (see Figure EC.2) from Ziliaskopoulos (2000) to illustrate the optimal solutions of the proposed models with and without NVH consideration. The network contains nine nodes, eleven links, and one OD pair. The detailed scenario setting is presented in Appendix EC.6.1. Note that the original network of Ziliaskopoulos (2000) does not contain Link 11 and is acyclic. The modified network contains a cycle by adding Link 11. The parameters for LP problems in Section 2 and Appendix EC.7 are $\kappa = \kappa_1 = 0.0001$ and $\kappa_2 = 0.01$. Unless stated otherwise, the interval is used as the unit for the TSTT throughout the paper.

Table 1 shows the link occupancies corresponding to the optimal solutions to the R-SO-, NVH-SO-, and MTSTD-NVH-SO-DTA problems. The TSTTs of the three problems are identical and equal 215 intervals, which is the same as that in Ziliaskopoulos (2000). The results presented in Table 2 show that holding flows exist on Link 1 during intervals 2, 3, and 5 for the R-SO flow pattern, where $y_{ab}(k)$ is the number of vehicles that travel from link a to link b during interval k, and $\overline{y}_{ab}(k)$ is the maximum number of vehicles that can travel from link a to link b during interval k. For example, eight vehicles are on Link 1 at the beginning of interval 2, and the maximum number of vehicles that can travel into Link 2 during interval 2 is eight (see Table 2). However, only six vehicles travel into Link 2 during this interval, and two vehicles are held on Link 1. A similar phenomenon can be observed on Link 1 during intervals 3 and 5. In contrast to the R-SO flow pattern, we cannot see a vehicle holding phenomenon in both the NVH-SO and MTSTD-NVH-SO flow patterns. This is consistent with the results presented in Proposition 1. The TSTTs under different values of κ are shown in Figure 1. We find that the TSTTs are 215 intervals when $\kappa \leq 0.18$, which is also consistent with Proposition 1.

Table 1 also shows that the link occupancies of Link 11 at the beginning of intervals 5 and 6 are positive for the NVH-SO flow pattern. However, flows are not present on Link 11 at all times in the MTSTD-NVH-SO flow pattern because the NVH-SO-DTA model encourages cyclic traffic

flows, and the MTSTD-NVH-SO model eliminates unnecessary cyclic flows in cyclic networks. In Appendix EC.7, an example is developed to demonstrate that cyclic flows may not be completely eliminated in optimal solutions to NVH-SO-DTA and MTSTD-NVH-SO-DTA problems.

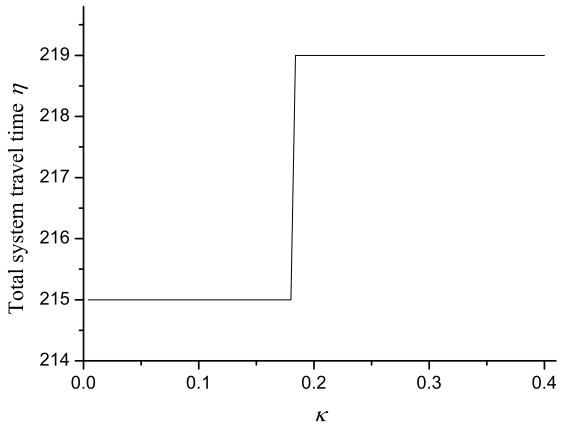

Figure 1 The TSTTs under different values of κ for Example 1.

Table 1. The optimal link occupancies for the three models in Example 1 ($\eta = 215$): the MTSTD-NVH-SO results are shown in normal face, the R-SO results are shown in italics, and the NVH-SO results are underlined.

Time Interval	Link 1	2	3	4	5	6	7	8	9	10	11
1	8 *8* 8	0	0	0	0	0	0	0	0	0	0
2	16 *18* 16	8 *6* 8	0	0	0	0	0	0	0	0	0
3	12 *22* 12	12 *4* 12	2 *0* 2	0	6 *6* 6	0	0	0	0	0	0
4	4 *10* 4	10 *12* 10	8 *0* 6	0	4 *4* 4	0 *0* 2	6 *6* 6	0	0	0	0
5	0 *6* 0	6 *4* 4	10 *6* 4	0	6 *6* 6	0 *0* 6	4 *4* 4	6 *6* 6	0	0	0 *0* **2**
6	0	2 *6* 0	7 *3* 3	3 *3* 3	4 *1* 3	0 *3* 1	6 *6* 6	4 *4* 4	6 *6* 6	0	0 *0* **6**
7	0	0	6 *6* 6	3 *3* 3	0	0	4 *4* 4	6 *6* 6	7 *7* 7	6 *6* 6	0
8	0	0	0	6 *6* 6	0	0	0	4 *4* 4	9 *9* 9	13 *13* 13	0
9	0	0	0	0	0	0	0	0	10 *10* 10	22 *22* 22	0
10	0	0	0	0	0	0	0	0	0	32 *32* 32	0

Example 2. Comparing the optimal solutions of models with and without FIFO or NVH consideration.

In this example, an X-shaped network (see Figure EC.3) is used to illustrate the proposed branch-and-bound algorithm for the FIFO-SO-DTA problem and to compare different optimal solutions. The X-shaped network has six nodes, five links, two origins, two destinations, and two OD pairs. The detailed scenario setting is presented in Appendix EC.6.2. The parameter values for LP problems in this example (i.e., κ, κ_1, and κ_2) are the same as those in Example 1. The values of the priority parameters are $\underline{p} = 0.01$ and $\overline{p} = 100$.

Table 2. The optimal flows of Links 1 and 2 of the R-SO-DTA model in Example 1.

Time interval	$U_1(k)$	$V_1(k)$	$y_{12}(k)$	$\overline{y}_{12}(k)$	$U_2(k)$	$V_2(k)$
0	0	0	0	0	0	0
1	8	0	0	0	0	0
2	24	***6***	***6***	***8***	6	0
3	32	***10***	***4***	***12***	10	6
4	32	22	12	12	22	10
5	32	***26***	***4***	***8***	26	22
6	32	32	6	6	32	26
7	32	32	0	0	32	32

The steps of the proposed branch-and-bound algorithm are illustrated in Appendix EC.8. The results of the optimal cumulative transfer flows for the R-SO-, NVH-SO-, FIFO-SO-, and NVH-FIFO-SO-DTA problems are shown in Table 3, where (a, b) indicates the flow from link a to link b. We can observe from the table that the optimal TSTTs for the first two problems are both 270 intervals. However, the optimal TSTTs increase to 290 and 322.5 intervals for the last two problems, respectively. This result is consistent with Proposition 6. When we introduce the penalty term into the FIFO-SO-DTA model, we obtain a FIFO-SO flow pattern if the coefficient approaches zero. To achieve minimum TSTT, the NVH property cannot be ensured. Therefore, the penalty approach cannot be applied for the NVH-FIFO formulation. Tables 3 and 4 also show that vehicle holding phenomena exist in the R-SO and FIFO-SO solutions. For example, in the R-SO solution, 20 vehicles are on Link 1 at the beginning of interval 3, and the maximum number of vehicles that can travel into Link 2 during interval 3 is 20 (see Table 4). However, only 15 vehicles travel into

Link 2 during this interval, and 5 are held on Link 1. In this example, if NVH constraints are not considered, the FIFO-SO flow pattern is an SO flow pattern and the NVH-FIFO-SO flow pattern is a UE flow pattern. Tables 3 and 4 show that the SO flow pattern holds some vehicles on Link 1 during interval 3 to reduce the overall travel time.

Table 3. The optimal cumulative transfer flows for various SO-DTA problems in the X-shaped network.

Time	R-SO(η=270)				NVH-SO(η=270)				FIFO-SO(η=290)				NVH-FIFO-SO(η=322.5)			
Interval	$(1,3)$	$(3,4)$	$(2,3)$	$(3,5)$	$(1,3)$	$(3,4)$	$(2,3)$	$(3,5)$	$(1,3)$	$(3,4)$	$(2,3)$	$(3,5)$	$(1,3)$	$(3,4)$	$(2,3)$	$(3,5)$
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	20	0	0	0	20	0	0	0	20	0	0	0	20	0	0	0
3	**35**	0	0	0	40	0	0	0	**25**	0	0	0	40	0	0	0
4	40	10	10	0	45	10	10	0	30	10	10	0	40	10	10	0
5	45	20	20	0	50	20	20	0	35	20	20	0	45	20	20	0
6	50	**30**	20	**10**	50	**30**	20	**10**	40	30	20	10	50	30	20	0
7	50	**40**	20	**20**	50	**40**	20	**20**	45	35	20	20	50	40	20	0
8	50	50	20	20	50	50	20	20	50	40	20	20	50	40	20	10
9	50	50	20	20	50	50	20	20	50	45	20	20	50	42.5	20	15
10	50	50	20	20	50	50	20	20	50	50	20	20	50	50	20	20

Table 4. The optimal cumulative flows of Link 1 of the R-SO and FIFO-SO solutions in Example 2.

Time		R-SO			FIFO-SO		
interval	$U_1(k)$	$V_1(k)$	$y_{13}(k)$	$\bar{y}_{13}(k)$	$V_1(k)$	$y_{13}(k)$	$\bar{y}_{13}(k)$
0	0	0	0	0	0	0	0
1	20	0	0	0	0	0	0
2	40	20	20	20	20	20	20
3	50	**35**	**15**	**20**	**25**	**5**	**20**
4	50	40	5	5	30	5	5
5	50	45	5	5	35	5	5
6	50	50	5	5	40	5	5
7	50	50	0	0	45	5	5
8	50	50	0	0	50	5	5

The optimal cumulative flows of Link 3 of the R-SO solution are provided in Table 5. We can observe from the table that $U_3^{s_2}(\underline{\ell}^*_{3,k}) < V_3^{s_2}(k)$ and $U_3^{s_1}(\overline{\ell}^*_{3,k}) > V_3^{s_1}(k)$ are satisfied for $k = 6$ and

$k = 7$. This implies that the vehicles moving to destination s_2 overpass those moving to destination s_1 during intervals 6 and 7. This result is consistent with Propositions 2 and 3. Similarly, FIFO violations can also be observed from the NVH-SO flow during intervals 6 and 7.

Table 5. The optimal cumulative flows of Link 3 for the R-SO-DTA model in Example 2.

Time interval	$U_3^{s_1}(k)$	$V_3^{s_1}(k)$	$U_3^{s_2}(k)$	$V_3^{s_2}(k)$	$\underline{\ell}^*_{3,k}$	$U_3^{s_1}(\underline{\ell}^*_{3,k})$	$U_3^{s_2}(\underline{\ell}^*_{3,k})$	$\overline{\ell}^*_{3,k}$	$U_3^{s_1}(\overline{\ell}^*_{3,k}))$	$U_3^{s_2}(\overline{\ell}^*_{3,k}))$
0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0
2	20	0	0	0	0	0	0	0	0	0
3	35	0	0	0	1	0	0	0	0	0
4	40	10	10	0	1.5	10	0	1.5	10	0
5	45	20	20	0	2	20	0	2	20	0
6	50	30	20	10	2.67	30	**0**	4	**40**	10
7	50	40	20	20	4	40	**10**	5	**45**	20
8	50	50	20	20	6	50	20	6	50	20

Example 3. Comparing the accuracy and computational efficiency of the LTM-based models with those of the CTM-based counterparts.

In this example, we adopt a modified Nguyen and Dupuis network (see Figure EC.4) to illustrate the accuracy and computational efficiency of the proposed LTM-based models. The network has 17 nodes, 23 links, and 4 OD pairs. The detailed specification of the network is presented in Appendix EC.6.3. To demonstrate the efficiency of the link-based SO-DTA models, we construct two scenarios with different link lengths. Except for the origin and destination links, the lengths of all of the normal links in Scenario 2 double those used in Scenario 1. The time interval is 10 s, and each of the OD demands last for the first 10 intervals. The modeling horizons are set at 35 and 70 intervals for Scenarios 1 and 2, respectively. The values of the priority parameters are the same as those given in Example 2.

In Table 6, we compare the sizes of the LTM-based R-SO-DTA problem without FIFO constraints with the corresponding CTM-based R-SO-DTA problem for Scenarios 1 and 2. We find that the LTM-based formulations have considerably fewer variables and constraints than the CTM-based counterparts. The LTM-based formulations can thus be solved more efficiently. In particular, as

shown in Table 7, the CPU times required to solve the CTM-based R-SO-DTA, NVH-SO-DTA, and FIFO-SO-DTA models are at least 2.28 times greater than those required to solve the LTM-based counterparts in Scenario 1. Compared with Scenario 1, the merit of the LTM-based models in terms of computational efficiency is more remarkable in Scenario 2. In this scenario, the network size for the LTM-based models is unchanged, whereas for the CTM-based counterparts the network size increases due to the increase in the link length, which leads to an increase in the number of cells for the CTM-based models, and the large increase in the model size and the CPU times required to solve the models (see Tables 6 and 7). In addition, we observe that the R-SO, NVH-SO, FIFO-SO, and NVH-FIFO-SO-DTA models, which are either formulated as LTM- or CTM-based models, give the same TSTTs of 5287.5 and 9635 intervals for Scenarios 1 and 2, respectively. This implies that the LTM-based SO-DTA models maintain the same level of accuracy as the corresponding CTM-based SO-DTA models.

Table 6. The sizes of the R-SO-DTA problem for Scenarios 1 and 2 in Example 3.

Problem size	Scenario 1		Scenario 2	
	LTM	CTM	LTM	CTM
Number of variables	1656	5400	3266	20732
Increase in the number of variables	0.00%	226.09%	0.00%	534.78%
Number of constraints	8407	29300	16734	115086
Increase in the number of constraints	0.00%	248.52%	0.00%	587.74%

Table 7. The CPU times (in seconds) required to solve the LTM- and CTM-based models in Example 3.

Scenario	R-SO		NVH-SO		FIFO-SO		NVH-FIFO-SO	
	CTM	LTM	CTM	LTM	CTM	LTM	CTM	LTM
1	1.14	0.50	1.25	0.39	11.65	4.49	36.32	5.48
2	13.47	1.08	14.58	1.20	625.23	19.32	72186.28	1431.33

Example 4. Comparing the CPU times required to solve various models in a modified Sioux Falls network.

In this example, we adopt a modified Sioux Falls network (see Figure EC.5) to illustrate the computational efficiency of the proposed models. The network has 43 nodes, 95 links, and 12 OD pairs. The detailed network setting is presented in Appendix EC.6.4. We have two scenarios (Scenarios 3 and 4) with different OD demands. The OD demands last for the first 30 min in Scenario 3 and the first 60 min in Scenario 4. The OD demand profiles of both scenarios are presented in Figure EC. 6. The time interval is 1 min, and the modeling horizon is set at 50 min and 80 min in Scenarios 3 and 4, respectively. Table 8 illustrates the CPU times required to solve different SO-DTA models. The results presented in Table 8 demonstrate that the proposed models and algorithms are computationally efficient and solvable for some networks of reasonable size, and the OD demands have a significant effect on the CPU time required to solve the proposed SO-DTA models.

Table 8. The CPU times (in seconds) required to solve the four LTM-based models in Example 4.

Scenario	R-SO	NVH-SO	FIFO-SO	NVH-FIFO-SO
3	80.74	77.51	226.25	292.36
4	329.35	285.10	919.45	4799.44

6. Conclusions

In this paper, the LTM is used to formulate link-based SO-DTA models in terms of cumulative flows. Similar to the well-known cell-based SO-DTA models, each of the link-based SO-DTA models without FIFO constraints can lead to an LP formulation. The LTM does not discretize links into small cells, so the link-based models have higher computational efficiency than the cell-based counterparts. We address the vehicle holding problem in SO-DTA problems without FIFO constraints by adding a penalty term, a sufficiently small coefficient multiplied by the sum of cumulative outflows, to the objective function of the R-SO-DTA problem. This method encourages cyclic SO flows to be produced in cyclic networks. To overcome this problem, we add another penalty term, a

sufficiently small coefficient multiplied by the total system travel distance, to the objective function of the R-SO-DTA problem to reduce the unnecessary cyclic flows.

We introduce an entry time to define the FIFO condition and then formulate the FIFO constraints. By integrating the FIFO constraints into the R-SO-DTA problem, we obtain the FIFO-SO-DTA problem. By integrating the NVH and FIFO constraints into the R-SO-DTA problem, we obtain the NVH-FIFO-SO-DTA problem. As the FIFO constraints contain bilinear terms, the feasible solution set of the SO-DTA problems with FIFO constraints is nonconvex, and it is difficult to guarantee that (local) optimal solutions of the corresponding optimization models will be obtained. According to the optimality conditions of the proposed optimization problems, branch-and-bound algorithms were developed to solve the two SO-DTA problems with FIFO constraints. Traditional branch-and-bound algorithms were developed for discrete programming problems, which include integer programming and combinatorial optimization problems (Mitten 1970), and for nonlinear programming problems, which minimize nonconvex objective functions (Lawler and Wood 1966). These algorithms directly branch decision variables. However, the proposed branch-and-bound algorithms are based on the properties of the proposed optimization problems. They do not directly branch decision variables, but aim to search for proper ranges of link entry times using the set of pairs of link and interval indices of the flow vector with FIFO violations, identified by the conditions stated in Propositions 2 and 3. The critical entry times obtained from the FIFO violation identification are used to develop the branching scheme.

Examples are given to show the properties of the proposed models and the performance of the algorithms. The results show that the proposed LP formulation for the NVH-SO-DTA problem can address the vehicle holding problem for general networks, and that the MTSTD-NVH-SO-DTA model can reduce the unnecessary cyclic flow. The proposed branch-and-bound algorithms can solve the two SO-DTA problems with FIFO constraints. When FIFO violations occur, the optimal TSTTs of the two SO-DTA problems with FIFO constraints can be greater than those without. The proposed link-based SO-DTA models are consistent with the cell-based SO-DTA counterparts

in terms of yielding an identical optimal TSTT. However, the link-based models outperform the cell-based counterparts in terms of computational efficiency.

The SO-DTA models considering both route and departure time choices are extensions of the models that only consider route choice. The proposed models can be directly extended to simultaneously model route and departure time choices under the DSO principle by modifying the objective functions to consider the penalty cost for early and late arrivals, and by modifying the flow conservation conditions at origins or destinations, to ensure that the total demand of each OD pair over the modeling period is fixed. In this paper, we propose link-based DSO models, in which we only consider link FIFO. As we do not distinguish the paths of the traffic that heads to the same destination, path FIFO cannot be considered. In the future, we will develop path-based DSO models, in which path FIFO will be considered.

Acknowledgments

This work is jointly supported by the National Natural Science Foundation of China (71431003, 71522001, 71621001), the Fundamental Research Funds for the Central Universities (JZ2016HGPB0736), and a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (HKU 17201915). The authors are grateful to the Area Editor, Associate Editor, and two reviewers for their constructive comments.

References

Ban X, Pang JS, Liu X, Ma R (2012) Continuous-time point-queue models in dynamic network loading. *Transportation Res. Part B* **46**(3):360-380.

Bliemer MCJ, Raadsen MPH, Smits ES, Zhou B, Bell MGH (2014) Quasi-dynamic traffic assignment with residual point queues incorporating a first order node model. *Transportation Res. Part B* **68**:363-384.

Carey M (1987) Optimal time-varying flows on congested networks. *Oper. Res.* **35**(1):58-69.

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

Carey M (1992) Nonconvexity of the dynamic traffic assignment problem. *Transportation Res. Part B* **26**(2):127-133.

Carey M (2004) Link travel times I: Desirable properties. *Networks and Spatial Econom.* **4**(3):257-268.

Carey M, Humphreys P, McHugh M, McIvor R (2014) Extending travel-time based models for dynamic network loading and assignment, to achieve adherence to first-in-first-out and link capacities. *Transportation Res. Part B* **65**:90-104.

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

Chow A (2009) Properties of system optimal traffic assignment with departure time choice and its solution method. *Transportation Res. Part B* **43**(3):325-344.

Daganzo CF (1995) The cell transmission model, Part II: Network traffic. *Transportation Res. Part B* **29**(2):79-93.

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

Doan K, Ukkusuri SV (2015) Dynamic system optimal model for multi-OD traffic networks with an advanced spatial queuing model. *Transportation Res. Part C* **51**:41-65.

Friesz TL, Bernstein D, Smith TE, Tobin RL, Wie B (1993) A variational inequality formulation of the dynamic networks user equilibrium problem. *Oper. Res.* **41**(1):179-191.

Guo RY, Yang H, Huang HJ (2018) Are we really solving the dynamic traffic equilibrium problem with a departure time choice? *Transportation Sci.* in press.

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

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

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

Han K, Gayah V, Piccoli B, Friesz TL, Yao T (2014) On the continuum approximation of the on-and-off signal control on dynamic traffic networks. *Transportation Res. Part B* **61**:73-97.

Han K, Liu H, Gayah V, Friesz TL, Yao T (2016a) A robust optimization approach for dynamic traffic signal control with emission considerations. *Transportation Res. Part C* **70**:3-26.

Han K, Friesz TL, Szeto WY, Liu H-C (2015a) Elastic demand dynamic network user equilibrium: Formulation, existence and computation. *Transportation Res. Part B* **81**:183-209.

Han K, Szeto WY, Friesz TL (2015b) Formulation, existence, and computation of boundedly rational dynamic user equilibrium with fixed or endogenous user tolerance. *Transportation Res. Part B* **79**:16-49.

Han K, Piccoli B, Szeto WY (2016b) Continuous-time link-based kinematic wave model: Formulation, solution existence, and well-posedness. *Transportmetrica B* **4**(3):187-222.

Ho JK (1980) A successive linear optimization approach to the dynamic traffic assignment problem. *Transportation Sci.* **14**(4):295-305.

Jiang Y, Szeto WY, Long JC, Han K. (2016) Multi-class dynamic traffic assignment with physical queues: Intersection-movement-based formulation and paradox. *Transportmetrica A* **12**(10):878-908.

Jin W (2015) Point queue model: A unified approach. *Transportation Res. Part B* **77**:1-16.

Lawler EL, Wood DE (1966) Branch-and-bound methods: A survey. *Oper. Res.* **14**(4):699-719.

Levin MW (2017) Congestion-aware system optimal route choice for shared autonomous vehicles. *Transportation Res. Part C* **82**:229-247.

Lighthill MH, Whitham GB (1955) On kinematics wave. II. A theory of traffic flow on long crowed roads. *Proc. Royal Society, London, Series A* **229** (1178):317-345.

Lin WH, Wang C (2004) An enhanced 0-1 mixed-integer LP formulation for traffic signal control. *IEEE Trans. Intell. Transp. Syst.* **5**(4):238-245.

Liu Y, Nie Y, Hall J (2015) A semi-analytical approach for solving the bottleneck model with general user heterogeneity. *Transportation Res. Part B* **71**:56-70.

Lo H (1999) A novel traffic signal control formulation. *Transportation Res. Part B* **33**(6):433-448.

Lo HK, Szeto WY (2002) A cell-based variational inequality formulation of the dynamic user optimal assignment problem. *Transportation Res. Part B* **36**(5):421-443.

Long JC, Gao ZY, Szeto WY (2011) Discretised link travel time models based on cumulative flows: Formulation and properties. *Transportation Res. Part B* **45**(1):232-254.

Long JC, Huang HJ, Gao ZY (2013a) Discretised route travel time models based on cumulative flows. *J. Adv. Transportation* **47**(1):105-125.

Long JC, Huang HJ, Gao ZY, Szeto WY (2013b) An intersection-movement-based dynamic user optimal route choice problem. *Oper. Res.* **61**(5):1134-1147.

Long JC, Szeto WY, Gao ZY, Huang HJ, Shi, Q (2016) The nonlinear equation system approach to solving dynamic user optimal simultaneous route and departure time choice problems. *Transportation Res. Part B* **83**:179-206.

Long JC, Chen JX, Szeto WY, Shi, Q (2018) Link-based system optimum dynamic traffic assignment problems with environmental objectives. *Transportation Res. Part D* **60**:56-75.

Long JC, Szeto WY, Huang HJ, Gao ZY (2015a) An intersection-movement-based stochastic dynamic user optimal route choice model for assessing network performance. *Transportation Res. Part B* **74**:182-217.

Long JC, Szeto WY, Shi Q, Gao ZY, Huang HJ (2015b) A nonlinear equation system approach to the dynamic stochastic user equilibrium simultaneous route and departure time choice problem. *Transportmetrica A* **11**(5):388-419.

Ma R, Ban XJ, Pang JS (2014) Continuous-time dynamic system optimum for single-destination traffic networks with queue spillbacks. *Transportation Res. Part B* **68**:98-122.

Ma R, Ban XJ, Szeto WY (2017). Emission modeling and pricing on single-destination dynamic traffic networks. *Transportation Res. Part B* **100**:255-283.

Merchant DK, Nemhauser GL (1978a) A model and an algorithm for the dynamic traffic assignment. *Transportation Sci.* **12**(3):183-199.

Merchant DK, Nemhauser GL (1978b) Optimality conditions for a dynamic traffic assignment model. *Transportation Sci.* **12**(3):200-207.

Mitten LG (1970) Branch-and-bound methods: General formulation and properties. *Oper. Res.* **18**(1):24-34.

Newell GF (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 Res. Part B* **27**(4):281-314.

Ngoduy D, Hoang NH, Vu HL, Watling D (2016) Optimal queue placement in dynamic system optimum solutions for single origin-destination traffic networks. *Transportation Res. Part B* **92**:148-169.

Nguyen S, Dupuis C (1984) An efficient method for computing traffic equilibria in networks with asymmetric transportation costs. *Transportation Sci.* **18**(2):185-202.

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

Pavlis Y, Recker W (2009) A mathematical logic approach for the transformation of the linear conditional piecewise functions of dispersion-and-store and cell transmission traffic flow models into linear mixed-integer form. *Transportation Sci.* **43**(1):98-116.

Perakis G, Roels G (2006) An analytical model for traffic delays and the dynamic user equilibrium problem. *Oper. Res.* **54**(6):1151-1171.

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

Ramadurai G, Ukkusuri S (2010) Dynamic user equilibrium model for combined activity-travel choices using activity-travel supernetwork representation. *Networks and Spatial Econom.* **10**(2):273-292.

Richards PI (1956) Shock waves on the highway. *Oper. Res.* **4**(1):42-51.

Shen W, Nie Y, Zhang H (2007). Dynamic network simplex method for designing emergency evacuation plans. *Transportation Res. Record* **2022**:83-93.

Szeto WY, Lo HK (2006) Dynamic traffic assignment: Properties and extensions. *Transportmetrica* **2**(1):31-52.

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

Ukkusuri SV, Waller ST (2008) Linear programming models for the user and system optimal dynamic network design problem: Formulations, implementations and comparisons. *Networks and Spatial Econom.* **8**(4):383-406.

Vickrey WS (1969) Congestion theory and transport investment. *Am. Econ. Rev.* **59**(2):251-260.

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

Zheng H, Chiu YC (2011) A network flow algorithm for the cell-based single-destination system optimal dynamic traffic assignment problem. *Transportation Sci.* **45**(1):121-137.

Zheng H, Chiu YC, Mirchandani PB (2015). On the system optimum dynamic traffic assignment and earliest arrival flow problems. *Transportation Sci.* **49**(1):13-27.

Zhu F, Ukkusuri SV (2013) A cell based dynamic system optimum model with non-holding back flows. *Transportation Res. Part C* **36**:367-380.

Zhu F, Ukkusuri SV (2015) A linear programming formulation for autonomous intersection control within a dynamic traffic assignment and connected vehicle environment. *Transportation Res. Part C* **55**:363-378.

Ziliaskopoulos AK (2000) A linear programming model for the single destination system optimum dynamic traffic assignment problem. *Transportation Sci.* **34**(1):37-49.

This page is intentionally blank. Proper e-companion title page, with INFORMS branding and exact metadata of the main paper, will be produced by the INFORMS office when the issue is being assembled.

EC.1. Non-vehicle holding constraints

Constraint (4) in the LTM is relaxed to a set of less-than-or-equal constraints (7) in SO-DTA models. For an optimal solution to a SO-DTA model, if a pair of a link and an interval exist such that all of the constraints in condition (7) fall into the inequality region, vehicles are likely to be held in the link without moving forward, even if its successor link has sufficient capacity. Such a solution property is referred to as "vehicle-holding"(Zheng and Chiu 2011).

Definition EC.1 (Vehicle holding flow and NVH flow pattern). Let $\mathbf{x} \in \Omega$ be a feasible flow pattern. $\mathbf{x}$ is defined as a vehicle holding flow pattern if a pair of a link $a \in A$ and an interval $k \in K$ exists such that the feasible flow pattern satisfies the following conditions:

$$\begin{cases} \sum_{s\in S} V_a^s(k) < \sum_{s\in S} U_a^s(k-\overline{\tau}_a), \\ \sum_{s\in S}\Big[V_a^s(k) - V_a^s(k-1)\Big] < C_a(k), \\ \sum_{s\in S} U_b^s(k) < \sum_{s\in S} V_b^s(k-\overline{t}_b) + L_b\rho_{jam},\ \forall b \in \Gamma(a),\ \text{and} \\ \sum_{s\in S}\Big[U_b^s(k) - U_b^s(k-1)\Big] < Q_b(k), \qquad \forall b \in \Gamma(a), \end{cases} \tag{EC.1}$$

where $\Gamma(a)$ denotes the set of successor links of link a (downstream links directly connected to link a). Otherwise, $\mathbf{x}$ is defined as an NVH flow pattern.

This definition of vehicle holding is consistent with that used by Shen et al. (2007), Zheng and Chiu (2011), and Zhu and Ukkusuri (2013). The definition of an NVH flow pattern implies that at least one of the following less-than-or-equal inequalities take equality for each $a \in A$ and $k \in K$:

$$\begin{cases} \sum_{s\in S} V_a^s(k) \le \sum_{s\in S} U_a^s(k-\overline{\tau}_a), \\ \sum_{s\in S}\Big[V_a^s(k) - V_a^s(k-1)\Big] \le C_a(k), \\ \sum_{s\in S} U_b^s(k) \le \sum_{s\in S} V_b^s(k-\overline{t}_b) + L_b\rho_{jam},\ \forall b \in \Gamma(a),\ \text{and} \\ \sum_{s\in S}\Big[U_b^s(k) - U_b^s(k-1)\Big] \le Q_b(k), \qquad \forall b \in \Gamma(a). \end{cases} \tag{EC.2}$$

For any link a (except destination links) with $|\Gamma(a)|$ successor links, the first two inequalities in system (EC.2) restrict the sending flow of link a, and the last two inequalities in system (EC.2) restrict the receiving flows of all successor links of link a. Hence, the number of constraints in system (EC.2) with respect to interval k is $2+2|\Gamma(a)|$. The NVH conditions require that an equality holds

in at least one of the $2+2|\Gamma(a)|$ constraints. We can equivalently formulate the NVH conditions as the following mixed integer constraints (Pavlis and Recker 2009):

$$-\Big[\theta_a^0(k)+\sum_{i=1}^{m_a}\theta_a^i(k)\Big]M \le \sum_{s\in S}[V_a^s(k)-U_a^s(k-\overline{\tau}_a)], \tag{EC.3a}$$

$$-\Big[1-\theta_a^0(k)+\sum_{i=1}^{m_a}\theta_a^i(k)\Big]M \le \sum_{s\in S}[V_a^s(k)-V_a^s(k-1)]-C_a(k), \tag{EC.3b}$$

$$-\Big[\sum_{i=1}^{m_a}\sigma_i^j+\theta_a^0(k)-\sum_{i=1}^{m_a}(2\sigma_i^j-1)\theta_a^i(k)\Big]M \le \sum_{s\in S}U_{b_j}^s(k)-\sum_{s\in S}V_{b_j}^s(k-\overline{t}_{b_j})-L_{b_j}\rho_{jam}, \forall j\in J_a, \tag{EC.3c}$$

$$-\Big[1+\sum_{i=1}^{m_a}\sigma_i^j-\theta_a^0(k)-\sum_{i=1}^{m_a}(2\sigma_i^j-1)\theta_a^i(k)\Big]M \le \sum_{s\in S}[U_{b_j}^s(k)-U_{b_j}^s(k-1)]-Q_{b_j}(k), \forall j\in J_a, \tag{EC.3d}$$

$$\sum_{i=0}^{m_a}2^i\theta_a^i(k)\le 1+2|\Gamma(a)|, \text{ and} \tag{EC.3e}$$

$$\theta_a^i(k)\in\{0,1\}, \forall i=0,1,\cdots,m_a, \tag{EC.3f}$$

where M is a very large positive value. $J_a=\{1,2,\cdots,|\Gamma(a)|\}$ is an index set for successor links of link a. b_j is the j-th link in $\Gamma(a)$. $m_a=\arg\min_m\{2^{m+1}\ge 2+2|\Gamma(a)|\}$, and σ_i^j is 0 or 1 such that $\sum_{i=1}^{m_a}2^{i-1}\sigma_i^j=j$. There are m_a+1 integer variables in system (EC.3), which can form 2^{m_a+1} combinations. There are $2+2|\Gamma(a)|$ LTM-based flow constraints in system (EC.2), and constraint (EC.3e) implies that we only use $2+2|\Gamma(a)|$ combinations of the integer variables.

EC.2. Proofs

EC.2.1. Proof of Lemma 1

Proof. Substituting Eqs. (9) and (10) into Eq. (15), we have

$$\begin{aligned}
\eta &= \sum_{a\in A\setminus A_S}\sum_{s\in S}\sum_{k\in K}U_a^s(k)-\sum_{a\in A\setminus A_S}\sum_{s\in S}\sum_{k\in K}V_a^s(k)\\
&= \sum_{i\in N\setminus S}\sum_{a\in A(i)}\sum_{s\in S}\sum_{k\in K}U_a^s(k)-\sum_{a\in A_S}\sum_{s\in S}\sum_{k\in K}U_a^s(k)-\sum_{i\in N\setminus\{R,S\}}\sum_{a\in B(i)}\sum_{s\in S}\sum_{k\in K}V_a^s(k)\\
&= \sum_{i\in R}\sum_{a\in A(i)}\sum_{s\in S}\sum_{k\in K}U_a^s(k)+\sum_{i\in N\setminus\{R,S\}}\sum_{a\in A(i)}\sum_{s\in S}\sum_{k\in K}[U_a^s(k)-V_a^s(k)]-\sum_{a\in A_S}\sum_{s\in S}\sum_{k\in K}U_a^s(k)\\
&= \sum_{i\in R}\sum_{a\in A(i)}\sum_{s\in S}\sum_{k\in K}U_a^s(k)-\sum_{a\in A_S}\sum_{s\in S}\sum_{k\in K}U_a^s(k)\\
&= \sum_{a\in A_R}\sum_{s\in S}\sum_{k\in K}D_a^s(k)-\sum_{a\in A_S}\sum_{s\in S}\sum_{k\in K}U_a^s(k).
\end{aligned}$$

Therefore, LP problem (15) is equivalent to the following LP:

$$\min_{\mathbf{x}\in\Omega}\eta=\sum_{a\in A_R}\sum_{s\in S}\sum_{k\in K}D_a^s(k)-\sum_{a\in A_S}\sum_{s\in S}\sum_{k\in K}U_a^s(k).$$

The cumulative demand $D_a^s(k)$ is a model input and is predetermined, so the optimal solution to LP problem (15) is also the optimal solution to LP problem (16).

EC.2.2. Proof of Proposition 1

Proof. We first prove that any optimal solution to the following LP problem is an NVH-SO flow pattern:

$$\max_{\mathbf{x}\in\Omega}\xi=\sum_{a\in A}\sum_{k\in K}\sum_{s\in S}V_a^s(k). \tag{EC.4}$$

$$\text{Subject to }\sum_{a\in A\backslash A_S}\sum_{s\in S}\sum_{k\in K}\Big[U_a^s(k)-V_a^s(k)\Big]=\eta^*, \tag{EC.5}$$

where η^* is the value of the objective function for an optimal solution to LP problem (15), that is, the minimum TSTT of the R-SO-DTA problem.

Let $Y_{ab}^s(k)$ be the cumulative number of vehicles that leave link a and enter link b to destination s by the end of interval k. By definition, we have

$$U_a^s(k)=\sum_{b\in\Gamma^{-1}(a)}Y_{ab}^s(k),\forall a\in A\backslash A_R,s\in S,k\in K \text{ and} \tag{EC.6}$$

$$V_a^s(k)=\sum_{b\in\Gamma(a)}Y_{ab}^s(k),\forall a\in A\backslash A_S,s\in S,k\in K, \tag{EC.7}$$

where $\Gamma^{-1}(a)$ denotes the set of predecessor links of link a (upstream links directly connected to link a).

Let $\mathbf{x}^*$ be an optimal solution to LP problem (EC.4)-(EC.5). Assume that $\mathbf{x}^*$ contains vehicle holding flows. Consider link a where a vehicle holding phenomenon occurs during interval k ($k>0$). Condition (EC.1) must be true. Let $\mathbf{Y}^*=[Y_{ab}^{s^*}(k)]$ such that $\mathbf{x}^*$ and $\mathbf{Y}^*$ satisfy conditions (EC.6) and (EC.7). Without loss of generality, we assume that a holding-back traffic flow exists on link a

during interval k. The following optimization problem is then used to track the NVH flow of link a during interval k:

$$\max \hbar$$
$$s.t. \begin{cases} \sum_{s\in S} V_a^{s^*}(\hbar) - \sum_{s\in S} V_a^{s^*}(k-1) \le C_a(k), \\ \sum_{s\in S} V_a^{s^*}(\hbar) \le \sum_{s\in S} U_a^{s^*}(k-\overline{\tau}_a), \\ \sum_{s\in S} Y_{ab}^{s^*}(\hbar) + \sum_{a'\in\Gamma^{-1}(b)/\{a\}} \sum_{s\in S} Y_{a'b}^{s^*}(k) - \sum_{a'\in\Gamma^{-1}(b)} \sum_{s\in S} Y_{a'b}^{s^*}(k-1) \le Q_b(k), \forall b\in\Gamma(a), \\ \sum_{s\in S} Y_{ab}^{s^*}(\hbar) + \sum_{a'\in\Gamma^{-1}(b)/\{a\}} \sum_{s\in S} Y_{a'b}^{s^*}(k) \le \sum_{s\in S} V_b^{s^*}(k-\overline{t}_b) + L_b\rho_{jam}, \forall b\in\Gamma(a). \end{cases} \tag{EC.8}$$

Let $\hbar^*$ be an optimal solution to optimization problem (EC.8). By definition, we have

$$\sum_{s\in S} V_a^{s^*}(\hbar^*) > \sum_{s\in S} V_a^{s^*}(k). \tag{EC.9}$$

Based on the optimal solution $\mathbf{x}^*$, we can define the following:

$$\widetilde{V}_a^{s^*}(k) = V_a^{s^*}(k) + \frac{U_a^{s^*}(k-\overline{\tau}_a) - V_a^{s^*}(k)}{\sum_{s\in S}\left[U_a^{s^*}(k-\overline{\tau}_a) - V_a^{s^*}(k)\right]} \Big[\sum_{s\in S} V_a^{s^*}(\hbar^*) - \sum_{s\in S} V_a^{s^*}(k)\Big]. \tag{EC.10}$$

Based on the cumulative flow defined by Eq. (EC.10), we can construct $\bar{\mathbf{V}}^* = [\bar{V}_a^{s^*}(k), a\in A, s\in S, k\in K]$ by changing the cumulative outflows of link a, such that

$$\bar{V}_a^{s^*}(l) = \begin{cases} \widetilde{V}_a^{s^*}(k), \text{ if } l\in\{l\ge k : V_a^{s^*}(l) \le \widetilde{V}_a^{s^*}(k)\}, \\ V_a^{s^*}(l), \text{ otherwise.} \end{cases} \tag{EC.11}$$

For all $s\in S$ and $k\in K$, a time instant $\hbar_k^{s^*}$ exists so that $\bar{V}_a^{s^*}(k) = V_a^{s^*}(\hbar_k^{s^*})$. The vector $\bar{\mathbf{Y}}^* = [Y_{ab}^{s^*}(k), a\in A\setminus A_S, b\in\Gamma(a), s\in S, k\in K]$ can be constructed by only changing the cumulative transfer flows of link a such that

$$\bar{Y}_{ab}^{s^*}(k) = Y_{ab}^{s^*}(\hbar_k^{s^*}), \forall b\in\Gamma(a), s\in S, k\in K. \tag{EC.12}$$

We construct $\bar{\mathbf{U}}^* = [\bar{U}_a^{s^*}(k), a\in A, s\in S, k\in K]$ through $\bar{\mathbf{Y}}^*$ by Eqs. (9) and (EC.6). We can then construct a new solution $\bar{\mathbf{x}}^* = [\bar{\mathbf{U}}^*, \bar{\mathbf{V}}^*]$ that satisfies constraint (9). By definition, we have $A(i) = \Gamma(a)$ and $B(i) = \Gamma^{-1}(b)$ for all $a\in B(i)$ and $b\in A(i)$. As $\bar{\mathbf{Y}}^*$ satisfies Eqs. (EC.6) and (EC.7), we have

$$\sum_{a\in B(i)} V_a^{s^*}(k) = \sum_{a\in B(i)} \sum_{b\in\Gamma(a)} Y_{ab}^{s^*}(k) = \sum_{b\in A(i)} \sum_{a\in\Gamma^{-1}(b)} Y_{ab}^{s^*}(k) = \sum_{b\in A(i)} U_b^{s^*}(k). \tag{EC.13}$$

This implies that $\bar{\mathbf{x}}^*$ must satisfy constraint (10).

The definition of $\hbar^*$ implies that $\bar{\mathbf{x}}^*$ satisfies constraint (7) and $\sum_{s\in S} V_a^{s^*}(\hbar) \le \sum_{s\in S} U_a^{s^*}(k-\overline{\tau}_a)$. According to Eq. (EC.10), we have

$$\begin{aligned}\widetilde{V}_a^{s^*}(k) =& V_a^{s^*}(k) + \frac{U_a^{s^*}(k-\overline{\tau}_a) - V_a^{s^*}(k)}{\sum_{s\in S}\left[U_a^{s^*}(k-\overline{\tau}_a) - V_a^{s^*}(k)\right]}\Big[\sum_{s\in S} V_a^{s^*}(\hbar^*) - \sum_{s\in S} V_a^{s^*}(k)\Big] \\ \le& V_a^{s^*}(k) + \frac{U_a^{s^*}(k-\overline{\tau}_a) - V_a^{s^*}(k)}{\sum_{s\in S}\left[U_a^{s^*}(k-\overline{\tau}_a) - V_a^{s^*}(k)\right]}\Big[\sum_{s\in S} U_a^{s}(k-\overline{\tau}_a) - \sum_{s\in S} V_a^{s^*}(k)\Big] \\ =& U_a^{s^*}(k-\overline{\tau}_a).\end{aligned} \tag{EC.14}$$

According to inequality (EC.14) and Eq. (EC.11), we have $\bar{V}_a^{s^*}(k) \le U_a^{s^*}(k-\overline{\tau}_a)$ for all $a \in A\setminus A_S$, $s \in S$ and $k \in K$, that is, $\bar{\mathbf{x}}^*$ satisfies constraint (8). In the new solution $\bar{\mathbf{x}}^*$, according to Eq. (EC.12), we have

$$\bar{V}_a^{s^*}(k) = V_a^{s^*}(\hbar_k^{s^*}) = \sum_{b\in\Gamma(a)} Y_{ab}^{s^*}(\hbar_k^{s^*}) = \sum_{b\in\Gamma(a)} \bar{Y}_{ab}^{s^*}(k), \forall s \in S, k \in K. \tag{EC.15}$$

Constraint (EC.5) implies that $\mathbf{x}^*$ is also an optimal solution to LP problem (15). Eq. (EC.12) implies that $\bar{\mathbf{Y}}^* \ge \mathbf{Y}$, and hence conditions (EC.6) and (EC.8) imply $\bar{\mathbf{U}}^* \ge \mathbf{U}^*$ and $\bar{\mathbf{V}}^* \ge \mathbf{V}^*$. As $\mathbf{x}^*$ satisfy non-decreasing and nonnegativity constraints (11)-(13), $\mathbf{x}^*$ also satisfies non-decreasing and nonnegativity constraints (11)-(13). Therefore, the value of objective function (16) with respect to $\bar{\mathbf{x}}^*$ is not less than that with respect to $\mathbf{x}^*$. According to Lemma 1, $\mathbf{x}^*$ is an optimal solution to LP problem (15). Therefore, $\bar{\mathbf{x}}^*$ satisfies constraint (EC.5). In summary, $\bar{\mathbf{x}}^*$ satisfies constraints (7)-(13) and (EC.5) and hence is a feasible solution to optimization problem (EC.4)-(EC.5).

As $\overline{V}^* \ge V^*$, according to conditions (EC.9) and (EC.15), we have

$$\sum_{s\in S} \overline{V}_a^{s^*}(k) = \sum_{s\in S} V_a^{s^*}(\hbar^*) > \sum_{s\in S} V_a^{s^*}(k).$$

Therefore, we have

$$\sum_{a\in A}\sum_{s\in S}\sum_{k\in K} \overline{V}_a^{s^*}(k) > \sum_{a\in A}\sum_{s\in S}\sum_{k\in K} V_a^{s^*}(k).$$

This contradicts that $\mathbf{x}^*$ is an optimal solution to LP problem (EC.4)-(EC.5), and hence $\mathbf{x}^*$ is an NVH flow pattern. Constraint (EC.5) implies that $\mathbf{x}^*$ is an optimal solution to LP problem (15), and hence $\mathbf{x}^*$ is an NVH-SO flow pattern.

We then prove that if $\mathbf{x}^*$ is also an optimal solution to LP problem (15), then $\mathbf{x}^*$ is also an optimal solution to LP problem (EC.4)-(EC.5), and all optimal solutions to LP problem (18) are also optimal to LP problem (EC.4)-(EC.5):

For any flow pattern $\mathbf{x} \in \Omega$, the following definitions will be used to prove this proposition:

$$\eta(\mathbf{x}) = \sum_{a \in A \setminus A_S} \sum_{s \in S} \sum_{k \in K} \Big[U_k^s(a) - V_a^s(k) \Big],$$
$$\varpi(\mathbf{x}) = \sum_{a \in A_s} \sum_{s \in S} \sum_{k \in K} U_k^s(a), \quad \text{and}$$
$$\xi(\mathbf{x}) = \sum_{a \in A} \sum_{s \in S} \sum_{k \in K} V_a^s(k).$$

Let $\overline{\mathbf{x}}^*$ be an optimal solution to LP problem (EC.4)-(EC.5). As $\mathbf{x}^*$ is an optimal solution to LP problem (15), we have $\eta(\mathbf{x}^*) = \eta(\overline{\mathbf{x}}^*) = \eta^*$. According to Lemma 1, both $\mathbf{x}^*$ and $\overline{\mathbf{x}}^*$ are optimal solutions to LP problem (16), and we have $\varpi(\mathbf{x}^*) = \varpi(\overline{\mathbf{x}}^*)$. By definition, $\mathbf{x}^*$ is an optimal solution to LP problem (17), so we have $\varpi(\mathbf{x}^*) + \kappa\xi(\mathbf{x}^*) \geq \varpi(\overline{\mathbf{x}}^*) + \kappa\xi(\overline{\mathbf{x}}^*)$.

Assume that $\mathbf{x}^*$ is not an optimal solution to LP problem (EC.4)-(EC.5). We then have $\xi(\mathbf{x}^*) < \xi(\overline{\mathbf{x}}^*)$. This implies $\varpi(\mathbf{x}^*) + \kappa\xi(\mathbf{x}^*) < \varpi(\overline{\mathbf{x}}^*) + \kappa\xi(\overline{\mathbf{x}}^*)$, which contradicts $\varpi(\mathbf{x}^*) + \kappa\xi(\mathbf{x}^*) \geq \varpi(\overline{\mathbf{x}}^*) + \kappa\xi(\overline{\mathbf{x}}^*)$. Therefore, $\mathbf{x}^*$ is also an optimal solution to LP problem (EC.4)-(EC.5), and $\mathbf{x}^*$ is an NVH-SO flow pattern.

Let $\tilde{\mathbf{x}}^*$ be an optimal solution to LP problem (18). Assume that $\tilde{\mathbf{x}}^*$ is not an optimal solution to LP problem (EC.4)-(EC.5). Equivalently, $\tilde{\mathbf{x}}^*$ is not an optimal solution to LP problem (15). According to Lemma 1, $\tilde{\mathbf{x}}^*$ is not an optimal solution to LP problem (16). As $\mathbf{x}^*$ is an optimal solution to LP problem (16), we have $\varpi(\mathbf{x}^*) > \varpi(\tilde{\mathbf{x}}^*)$. As $\mathbf{x}^*$ and $\tilde{\mathbf{x}}^*$ are optimal solutions to LP problems (17) and (18), respectively, we have

$$\varpi(\mathbf{x}^*) + \kappa\xi(\mathbf{x}^*) \geq \varpi(\tilde{\mathbf{x}}^*) + \kappa\xi(\tilde{\mathbf{x}}^*) \text{ and} \tag{EC.16}$$

$$\varpi(\mathbf{x}^*) + \tilde{\kappa}\xi(\mathbf{x}^*) \leq \varpi(\tilde{\mathbf{x}}^*) + \tilde{\kappa}\xi(\tilde{\mathbf{x}}^*). \tag{EC.17}$$

Because $\varpi(\mathbf{x}^*) > \varpi(\tilde{\mathbf{x}}^*)$, we can rearrange inequalities (EC.16) and (EC.17) to get

$$\frac{1}{\tilde{\kappa}} \leq \frac{\xi(\tilde{\mathbf{x}}^*) - \xi(\mathbf{x}^*)}{\varpi(\mathbf{x}^*) - \varpi(\tilde{\mathbf{x}}^*)} \leq \frac{1}{\kappa}. \tag{EC.18}$$

Inequality (EC.18) implies $\kappa \leq \tilde{\kappa}$. This contradicts $\tilde{\kappa} < \kappa$, and hence $\tilde{\mathbf{x}}^*$ is also an optimal solution to LP problem (EC.4)-(EC.5). Any optimal solutions to LP problem (EC.4)-(EC.5) are NVH-SO flow patterns, so $\tilde{\mathbf{x}}^*$ is an NVH-SO flow pattern.

Remark EC.1: Note that there are no weights in objective function (EC.4) and that the formulation can be used for either acyclic or cyclic networks. Before solving LP problem (EC.4)-(EC.5), we should solve LP problem (15) to obtain the minimum TSTT η^*. In other words, two LP problems should be solved sequentially to obtain an optimal solution to LP problem (EC.4)-(EC.5). However, Proposition 1 shows that we can obtain an optimal solution to LP problem (EC.4)-(EC.5) by solving a single LP problem, which can effectively enhance the computational efficiency for solving the NVH-SO-DTA problem.

Remark EC.2: LP problem (17) can be viewed as LP problem (15) with perturbations on the coefficients of the objective function. If the coefficient approaches zero, LP problem (17) immediately becomes LP problem (16), which is equivalent to LP problem (15). Optimal solutions to LP problem (17) are also optimal to LP problem (15) only if the coefficient is positive and sufficiently small. Otherwise, optimal solutions to LP problem (17) are not optimal to LP problem (15), because if the coefficient is large enough, LP problem (17) mainly attempts to maximize the second term of its objective function and cannot achieve the maximum value of the first term of its objective function (or equivalently the maximum value of the objective function of LP problem (16)). Shen et al. (2007) showed that an optimal solution to single-destination SO-DTA problems that maintain the non-vehicle-holding property always exists. In other words, SO-DTA problems without NVH constraints can have multiple optimal solutions, and at least one of them is an NVH-SO flow pattern. This result can be directly extended to multi-destination SO-DTA problems without FIFO constraints. The penalty term for LP problem (16) is introduced to determine an NVH-SO flow pattern from the set of SO flow patterns.

EC.2.3. Proof of Proposition 2

Proof. As $U_a^s(\underline{\ell}_{a,k}^*) < V_a^s(k)$ and the cumulative flows are non-decreasing, condition (20) implies that $U_a^s(\underline{\ell}_{a,k}^*) = V_a^s(\underline{\ell}_{a,k}^* + \tau_a^s(\underline{\ell}_{a,k}^*)) < V_a^s(k)$, and there exists a time instant $\overline{\ell}$ such that $U_a^s(\overline{\ell}) = V_a^s(k)$ and $U_a^s(\ell) < V_a^s(k)$ for all $\ell < \overline{\ell}$. Hence, we have $\ell + \tau_a^s(\ell) < k$ for all $\underline{\ell}_{a,k}^* < \ell < \overline{\ell}$. This implies that the vehicles that enter link a to destination s at any time instant $\ell \in (\underline{\ell}_{a,k}^*, \overline{\ell})$ leave link a before the end of time interval k.

According to the definition of $\underline{\ell}_{a,k}^*$ by Eq. (23), there exists at least one destination $\overline{s} \in S$ such that $U_a^{\overline{s}}(\underline{\ell}_{a,k}^*) = U_a^{\overline{s}}(k)$. According to condition (20), we have $\underline{\ell}_{a,k}^* + \tau_a^{\overline{s}}(\underline{\ell}_{a,k}^*) = k$. This implies that the vehicles that enter link a to destination $\overline{s}$ at time instant $\underline{\ell}_{a,k}^*$ leave link a at the end of time interval k. Therefore, the vehicles that enter link a to destination s at any time instant $\ell \in (\underline{\ell}_{a,k}^*, \overline{\ell})$ overpass the vehicles that enter link a to destination $\overline{s}$ at time instant $\underline{\ell}_{a,k}^*$. This implies that $\mathbf{x}$ involves FIFO violations.

EC.2.4. Proof of Proposition 4

Proof. As $\mathbf{x} \in \Omega$ is a FIFO flow pattern, according to Proposition 2, $U_a^s(\underline{\ell}_{a,k}^*) \geq V_a^s(k)$ must be satisfied for all $s \in S$. According to definition of $\underline{\ell}_{a,k}^*$ by Eq. (23), we have $U_a^s(\underline{\ell}_{a,k}^*) \leq V_a^s(k)$ for all $s \in S$. Therefore, we have $V_a^s(k) = U_a^s(\underline{\ell}_{a,k}^*)$. Let $p_a(k) = \underline{\ell}_{a,k}^*$. Then, Eq. (26) is satisfied. This completes the proof.

EC.2.5. Proof of Proposition 5

Proof. For an arbitrary index pair (a, k), constraint (28a) implies that only two cases should be considered: (i) $\lambda_{a,k,l} < 1$ for all $l \leq k - \overline{\tau}_a$, and (ii) one interval l' exists such that $\lambda_{a,k,l'} = 1$ and $\lambda_{a,k,l} = 0$ for all $l \neq l'$.

For case (i), constraints (28a)-(28c) imply that there must be two adjacent intervals l' and $l'+1$ such that $\lambda_{a,k,l'} + \lambda_{a,k,l'+1} = 1$ and $\lambda_{a,k,l} = 0$ for all $l \notin \{l', l'+1\}$. This implies $\lambda_{a,k,l'+1} = 1 - \lambda_{a,k,l'}$. According to Eq. (27), we have

$$V_a^s(k) = \lambda_{a,k,l'} U_a^s(l') + (1 - \lambda_{a,k,l'}) U_a^s(l'+1), \forall s \in S. \tag{EC.19}$$

Based on Eqs. (EC.19) and (1), we can derive the entry time $p_a^s(k) = l' + 1 - \lambda_{a,k,l'}$. As $l' + 1 - \lambda_{a,k,l'}$ is independent with destination s, we have $p_a(k) = l' + 1 - \lambda_{a,k,l'} = p_a^s(k), \forall s \in S$. Therefore, condition (26) is satisfied. This implies that $\mathbf{x}$ is a FIFO flow pattern.

Because $1 - \lambda_{a,k,l'} = \lambda_{a,k,l'+1}$ and $\lambda_{a,k,l'} + \lambda_{a,k,l'+1} = 1$, we have

$$p_a(k) = l'(\lambda_{a,k,l'} + \lambda_{a,k,l'+1}) + \lambda_{a,k,l'+1} = l'\lambda_{a,k,l'} + (l'+1)\lambda_{a,k,l'+1}. \tag{EC.20}$$

$\lambda_{a,k,l} = 0$ is satisfied for all $l \neq l'$ and $l \neq l' + 1$, according to Eq. (EC.20), so we have

$$p_a(k) = l'\lambda_{a,k,l'} + (l'+1)\lambda_{a,k,l'+1} + \sum_{l \neq l', l \neq l'+1, l \leq k - \overline{\tau}_a} l\lambda_{a,k,l} = \sum_{l \leq k - \overline{\tau}_a} l\lambda_{a,k,l}. \tag{EC.21}$$

For case (ii), we also have $\lambda_{a,k,l'} + \lambda_{a,k,l'+1} = 1$ and $\lambda_{a,k,l} = 0$ for all $l \notin \{l', l'+1\}$, and hence case (ii) is a special case of case (i). The proof for case (ii) is the same as for case (i). This completes the proof.

EC.2.6. Proof of Theorem 1

Proof. $[\mathbf{x}^*, \lambda^*]$ is an optimal solution to the FIFO-SO-DTA problem, so we have $\mathbf{x}^* \in \Omega$, and $[\mathbf{x}^*, \lambda^*]$ satisfies constraints (27) and (28). According to Proposition 5, $\mathbf{x}^*$ is a FIFO flow pattern and $\mathbf{x}^*$ satisfies constraint (30b). Hence, $\mathbf{x}^*$ is a feasible solution to LP problem (30). If $\mathbf{x}^*$ is not an optimal solution to LP problem (30), we can find an optimal solution $\overline{\mathbf{x}}^*$ to LP problem (30). Let η^* and $\overline{\eta}^*$ be the objective values with respect to $\mathbf{x}^*$ and $\overline{\mathbf{x}}^*$, respectively. We then have $\overline{\eta}^* < \eta^*$. By definition, $\overline{\mathbf{x}}^* \in \Omega$ satisfies constraint (30b), and hence $[\overline{\mathbf{x}}^*, \lambda^*]$ satisfies constraints (27) and (28). Therefore, $[\overline{\mathbf{x}}^*, \lambda^*]$ is a feasible solution to the FIFO-SO-DTA problem. $[\mathbf{x}^*, \lambda^*]$ is an optimal solution to the FIFO-SO-DTA problem, so we have $\eta^* \leq \overline{\eta}^*$. This contradicts $\overline{\eta}^* < \eta^*$. Therefore, $\mathbf{x}^*$ is an optimal solution to LP problem (30).

EC.2.7. Proof of Proposition 6

Proof. Constraint (EC.5) implies $\eta_1^* = \eta_2^*$. The feasible solution set of the FIFO-SO-DTA problem (with objective function (29) and constraints (27) and (28)) is a subset of that of LP problem (15),

so $\eta_1^* \leq \eta_3^*$. Similarly, the feasible solution set of the NVH-FIFO-SO-DTA problem (with objective function (31) and constraints (27), (28), and (EC.3)) is a subset of that of the FIFO-SO-DTA problem, so $\eta_3^* \leq \eta_4^*$. This completes the proof.

EC.2.8. Proof of Proposition 7

Proof. $\mathbf{x}^*$ is a FIFO flow pattern, so we have

$$V_{a_\beta}^{s^*}(k_\beta) = U_{a_\beta}^{s^*}(p_{a_\beta}^*(k_\beta)), \forall s \in S, \beta \in IS_\alpha. \tag{EC.22}$$

$\mathbf{x}^*$ is an optimal solution to LP problem (33), so we have

$$U_{a_\beta}^{s^*}(\underline{\ell}_\beta) \leq V_{a_\beta}^{s^*}(k_\beta) \leq U_{a_\beta}^{s^*}(\overline{\ell}_\beta), \forall s \in S, \beta \in IS_\alpha. \tag{EC.23}$$

Substituting Eq. (EC.22) into condition (EC.23), we have

$$U_{a_\beta}^{s^*}(\underline{\ell}_\beta) \leq U_{a_\beta}^{s^*}(p_{a_\beta}^*(k_\beta)) \leq U_{a_\beta}^{s^*}(\overline{\ell}_\beta), \forall s \in S, \beta \in IS_\alpha. \tag{EC.24}$$

The cumulative flows are monotone with respect to the time instant, so condition (EC.24) implies that condition (34) is satisfied. This completes the proof.

EC.2.9. Proof of Proposition 8

Proof. By definition, we have $\widetilde{\Phi}_\alpha = \widetilde{\Phi}(\mathbf{x}^*)$. The definitions of $\underline{\ell}_{a,k}^*$ and $\overline{\ell}_{a,k}^*$ (i.e., conditions (23) and (24), respectively) imply that an optimal solution to LP problem (33), $\mathbf{x}^*$, satisfies $U_a^s(\underline{\ell}_{a,k}^*) \leq V_a^s(k) \leq U_a^s(\overline{\ell}_{a,k}^*)$ for all $(a,k) \in \widetilde{\Phi}_\alpha$. This implies that $\mathbf{x}^*$ is a feasible solution to LP problem (40)-(41). The feasible solution set of LP problem (40)-(41) is also a subset of that of LP problem (33), and hence the optimal objective value of LP problem (40)-(41) is no more than that of LP problem (33). Therefore, $\mathbf{x}^*$ is also an optimal solution to LP problem (40)-(41).

EC.3. LP formulation for the MTSTD-NVH-SO-DTA problem

To reduce cyclic flows in cyclic networks, we develop the following LP problem to obtain SO flows with MTSTD:

$$\min_{\mathbf{x} \in \Omega} \zeta = \sum_{a \in A} \sum_{s \in S} L_a U_a^s(\underline{K}). \tag{EC.25}$$

Subject to constraint (EC.5).

Objective function (EC.25) minimizes the total system travel distance (TSTD) or the sum of the travel distance of all flows. To achieve MTSTD under the SO condition, the unnecessary cyclic flows, which can be avoided without increasing the TSTT, should be removed. Note that in this study we are not concerned with the circumstances in which cyclic flow is beneficial to the system. Similar to LP problem (EC.4)-(EC.5), we can formulate the MTSTD-NVH-SO-DTA problem as the following LP problem:

$$\max_{\mathbf{x}\in\Omega} \xi = \sum_{a\in A}\sum_{s\in S}\sum_{k\in K} V_a^s(k). \tag{EC.26}$$

Subject to constraint (EC.5) and

$$\sum_{a\in A}\sum_{s\in S} L_a U_a^s(\underline{K}) = \zeta^*, \tag{EC.27}$$

where ζ^* is the value of the objective function for an optimal solution to the LP problem with objective function (EC.25) and constraint (EC.5).

Similar to LP problem (17), we have the following LP problem to obtain an MTSTD-NVH-SO flow pattern:

Proposition EC.1: For given $\kappa_1 > 0$ and $\kappa_2 > 0$, let $\mathbf{x}^*$ be an optimal solution to the following LP problem:

$$\max_{\mathbf{x}\in\Omega} \hat{\varpi} = \sum_{a\in A_S}\sum_{s\in S}\sum_{k\in K} U_a^s(k) + \kappa_1 \sum_{a\in A}\sum_{s\in S}\sum_{k\in K} V_a^s(k) - \kappa_2 \sum_{a\in A}\sum_{s\in S} L_a U_a^s(\underline{K}). \tag{EC.28}$$

If $\mathbf{x}^*$ is also an optimal solution to LP problem (15), then $\mathbf{x}^*$ is also an optimal solution to the LP problem with objective function (EC.26) and constraints (EC.5) and (EC.27), and all optimal solutions to the following LP problem are also optimal to the LP problem with objective function (EC.26) and constraints (EC.5) and (EC.27):

$$\max_{\mathbf{x}\in\Omega} \hat{\varpi} = \sum_{a\in A_S}\sum_{s\in S}\sum_{k\in K} U_a^s(k) + \hat{\kappa}_1 \sum_{a\in A}\sum_{s\in S}\sum_{k\in K} V_a^s(k) - \hat{\kappa}_2 \sum_{a\in A}\sum_{s\in S} L_a U_a^s(\underline{K}), \tag{EC.29}$$

where $0 < \hat{\kappa}_1 < \kappa_1$ and $0 < \hat{\kappa}_2 < \kappa_2$.

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

The unnecessary cyclic flow elimination term has a higher priority to be optimized than the NVH elimination term. Hence, we have $0 < \kappa_1 \ll \kappa_2 \ll 1$. (Note that the weight of the first term is one). Proposition EC.1 indicates that we can select sufficiently small positive parameter values κ_1 and κ_2 that satisfy $0 < \kappa_1 \ll \kappa_2 \ll 1$ and can solve only one LP problem to obtain an MTSTD-NVH-SO flow pattern.

EC.4. An example illustrating FIFO violations

In this example, we consider a network with two destinations (i.e., s_1 and s_2). The cumulative inflows and outflows of link a are graphically shown in Figure EC.1. We observe that the entry times of vehicles leaving link a to destinations s_1 and s_2 at the end of time interval k are $p_a^{s_1}(k) = \underline{\ell}^*_{a,k}$ and $p_a^{s_2}(k) = \overline{\ell}^*_{a,k}$, respectively. The travel times of vehicles leaving link a to destinations s_1 and s_2 at the end of time interval k are $\tau_a^{s_1}(p_a^{s_1}(k)) = k - \underline{\ell}^*_{a,k}$ and $\tau_a^{s_2}(p_a^{s_2}(k)) = k - \overline{\ell}^*_{a,k}$, respectively. As shown in Figure EC.1, we have $\overline{\ell}^*_{a,k} > \underline{\ell}^*_{a,k}$, and hence $\tau_a^{s_1}(p_a^{s_1}(k)) = k - \underline{\ell}^*_{a,k} > k - \overline{\ell}^*_{a,k} = \tau_a^{s2}(p_a^{s2}(k))$, that is, the travel time of vehicles leaving link a to destination s_1 at the end of time interval k is greater than that of the vehicles leaving link a to destination s_2 at the end of time interval k. For all $\tilde{\ell} \in (\underline{\ell}^*_{a,k}, \overline{\ell}^*_{a,k})$, we have $\tilde{\ell} + \tau_a^{s_2}(\tilde{\ell}) < \underline{\ell}^*_{a,k} + \tau_a^{s_1}(\underline{\ell}^*_{a,k}) = k$. This implies that the vehicles that enter link a during time period $(\underline{\ell}^*_{a,k}, \overline{\ell}^*_{a,k})$ and travel to destination s_2 overpass the vehicles that enter link a at time instant $\underline{\ell}^*_{a,k}$ and travel to destination s_1, that is, FIFO violations occur. We can also observe from Figure EC.1 that $U_a^{s_1}(\overline{\ell}^*_{a,k}) > V_a^{s_1}(k)$ and $U_a^{s_2}(\underline{\ell}^*_{a,k}) < V_a^{s_2}(k)$ are satisfied. According to either Proposition 2 or 3, the flow pattern in Figure EC.1 involves FIFO violations. This is consistent with our previous analysis.

EC.5. Procedures

EC.5.1. Procedure of FIFO violations identification

The following procedure can be used to identify FIFO violations in any feasible flow pattern $\mathbf{x}$ and to provide the set of pairs of link and time interval indices associated with FIFO violations for this flow pattern, which is denoted as $\tilde{\Phi}(\mathbf{x})$:

Procedure EC.1: FIFO_IDENTIFICATION ($\mathbf{x}$)

Initialize $\tilde{\Phi}(\mathbf{x}) \leftarrow \varnothing$
for each link $a \in A \backslash A_S$ and interval $k \in K$ **do**
  Retrieve the entry times $\underline{\ell}^*_{a,k}$ and $\overline{\ell}^*_{a,k}$ from $\mathbf{x}$ by Eq. (23) and (24), respectively.
  for each destination $s \in S$ **do**
    if $U^s_a(\underline{\ell}^*_{a,k}) < V^s_a(k)$ (or $U^s_a(\overline{\ell}^*_{a,k}) > V^s_a(k)$) **then**
      $\tilde{\Phi}(\mathbf{x}) \leftarrow \tilde{\Phi}(\mathbf{x}) + \{(a,k)\}$, and break.
    end if
  end for
end for

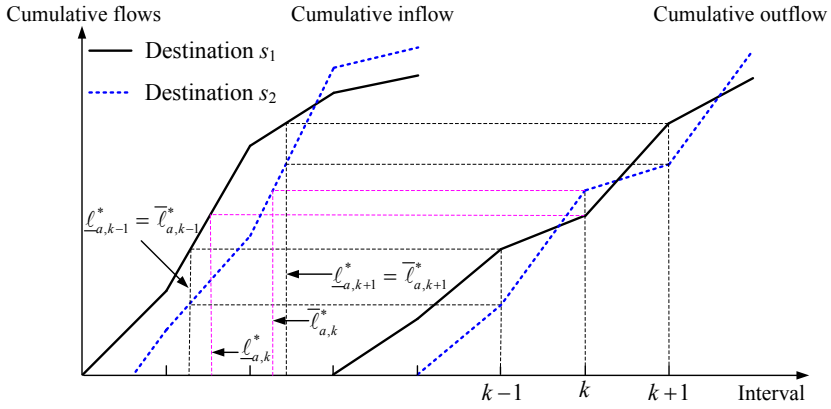

Figure EC.1 Illustration of FIFO violations.

EC.5.2. Procedure of branching

The following procedure is developed to branch a selected search tree node:

Remark EC.3: In the procedure FIFO_BRANCHING($\alpha, \mathbf{x}_\alpha$), we use the entry times $\underline{\ell}^*_{a,k}$ and $\overline{\ell}^*_{a,k}$ to check whether FIFO conditions are satisfied for all index pairs $(a,k) \in \Phi$. If a FIFO violation occurs for any index pair (a,k), new search tree nodes will be generated. One of the new generated search tree nodes (denoted as α_0 in the procedure) contains constraint (37). According to the description of the procedure, we have $\alpha_0 = n(a,k,\underline{\ell}^*_{a,k},\overline{\ell}^*_{a,k})$. The definition of the input sequence IS_{α_0} implies that the index pairs corresponding to all search tree nodes in $IS_{\alpha_0} - IS_\alpha$ are a subset of $\tilde{\Phi}_\alpha$. According to Proposition 8, $\mathbf{x}_\alpha$ is also an optimal solution to the LP problem corresponding to search tree node α_0, and hence we use it directly to generate more branches.

Remark EC.4: In this procedure, if $\tilde{\Phi}_\alpha = \varnothing$, $\mathbf{x}_\alpha$ is a FIFO flow pattern. However, if $\tilde{\Phi}_\alpha \neq \varnothing$, the

Procedure EC.2: FIFO_BRANCHING($\alpha, \mathbf{x}_\alpha$)

Initialize $\Upsilon \leftarrow \varnothing$ and $\overline{\alpha} \leftarrow \alpha$.
Obtain $\tilde{\Phi}(\mathbf{x}_\alpha)$ from FIFO_IDENTIFICATION($\mathbf{x}_\alpha$), and $\tilde{\Phi}_\alpha \leftarrow \tilde{\Phi}(\mathbf{x}_\alpha)$.
if $\tilde{\Phi}_\alpha = \varnothing$ **then**
  $DT \leftarrow DT + \{\alpha\}$.
else
  for each link $a \in A \setminus A_S$ and interval $k \in K$ **do**
    Retrieve the entry times $\underline{\ell}^*_{a,k}$ and $\overline{\ell}^*_{a,k}$ from $\mathbf{x}_\alpha$ by conditions (23) and (24), respectively.
    if $(a,k) \in \tilde{\Phi}_\alpha$ **then**
      Find the closest node $\beta \in IS_{\overline{\alpha}}$ to search tree node $\overline{\alpha}$ in terms of the distance between them in the sequence such that $a_\beta = a$ and $k_\beta = k$, where a_β and k_β are the link and interval indexes associated with β, respectively.
      if node β does not exist **then**
        Generate three nodes $\alpha_0 \leftarrow n(a,k,\underline{\ell}^*_{a,k},\overline{\ell}^*_{a,k})$, $\alpha_1 \leftarrow n(a,k,0,\underline{\ell}^*_{a,k})$, and $\alpha_2 \leftarrow n(a,k,\overline{\ell}^*_{a,k}, k-\vec{\iota}_a)$;
        Set $IS_{\alpha_i} \leftarrow IS_\alpha + \{\alpha_i\}$, $LB_{\alpha_i} \leftarrow LB_\alpha$, and $\tilde{\Phi}_{\alpha_i} \leftarrow \tilde{\Phi}_{\overline{\alpha}}$, $\forall i = 0,1,2$; $p_{\alpha_0} \leftarrow \overline{p}$, $p_{\alpha_1} \leftarrow \underline{p}$, and $p_{\alpha_2} \leftarrow \underline{p}$;
        Set $\Upsilon \leftarrow \Upsilon + \{\alpha_0\}$, $UT \leftarrow UT + \{\alpha_0,\alpha_1,\alpha_2\} - \{\overline{\alpha}\}$, and $\overline{\alpha} \leftarrow \alpha_0$.
      else if $\underline{\ell}^*_{a,k} > \underline{\ell}_\beta$ and $\overline{\ell}^*_{a,k} < \overline{\ell}_\beta$ **then**
        Generate three nodes $\alpha_0 \leftarrow n(a,k,\underline{\ell}^*_{a,k},\overline{\ell}^*_{a,k})$, $\alpha_1 \leftarrow n(a,k,\underline{\ell}_\beta,\underline{\ell}^*_{a,k})$, and $\alpha_2 \leftarrow n(a,k,\overline{\ell}^*_{a,k},\overline{\ell}_\beta)$;
        Set $IS_{\alpha_i} \leftarrow IS_\alpha + \{\alpha_i\}$, $LB_{\alpha_i} \leftarrow LB_\alpha$, and $\tilde{\Phi}_{\alpha_i} \leftarrow \tilde{\Phi}_{\overline{\alpha}}$, $\forall i = 0,1,2$; $p_{\alpha_0} \leftarrow \overline{p}$, $p_{\alpha_1} \leftarrow \underline{p}$, and $p_{\alpha_2} \leftarrow \underline{p}$;
        Set $\Upsilon \leftarrow \Upsilon + \{\alpha_0\}$, $UT \leftarrow UT + \{\alpha_0,\alpha_1,\alpha_2\} - \{\overline{\alpha}\}$, and $\overline{\alpha} \leftarrow \alpha_0$.
      else if $\underline{\ell}^*_{a,k} = \underline{\ell}_\beta$ and $\overline{\ell}^*_{a,k} < \overline{\ell}_\beta$ **then**
        Generate two nodes $\alpha_0 \leftarrow n(a,k,\underline{\ell}^*_{a,k},\overline{\ell}^*_{a,k})$, $\alpha_1 \leftarrow n(a,k,\overline{\ell}^*_{a,k},\overline{\ell}_\beta)$;
        Set $IS_{\alpha_i} \leftarrow IS_\alpha + \{\alpha_i\}$, $LB_{\alpha_i} \leftarrow LB_\alpha$, and $\tilde{\Phi}_{\alpha_i} \leftarrow \tilde{\Phi}_{\overline{\alpha}}$, $\forall i = 0,1$; $p_{\alpha_0} \leftarrow \overline{p}$ and $p_{\alpha_1} \leftarrow \underline{p}$;
        Set $\Upsilon \leftarrow \Upsilon + \{\alpha_0\}$, $UT \leftarrow UT + \{\alpha_0,\alpha_1\} - \{\overline{\alpha}\}$, and $\overline{\alpha} \leftarrow \alpha_0$.
      else if $\underline{\ell}^*_{a,k} > \underline{\ell}_\beta$ and $\overline{\ell}^*_{a,k} = \overline{\ell}_\beta$ **then**
        Generate two nodes $\alpha_0 \leftarrow n(a,k,\underline{\ell}^*_{a,k},\overline{\ell}^*_{a,k})$, and $\alpha_1 \leftarrow n(a,k,\underline{\ell}_\beta,\underline{\ell}^*_{a,k})$;
        Set $IS_{\alpha_i} \leftarrow IS_\alpha + \{\alpha_i\}$, $LB_{\alpha_i} \leftarrow LB_\alpha$, and $\tilde{\Phi}_{\alpha_i} \leftarrow \tilde{\Phi}_{\overline{\alpha}}$, $\forall i = 0,1$; $p_{\alpha_0} \leftarrow \overline{p}$ and $p_{\alpha_1} \leftarrow \underline{p}$;
        Set $\Upsilon \leftarrow \Upsilon + \{\alpha_0\}$, $UT \leftarrow UT + \{\alpha_0,\alpha_1\} - \{\overline{\alpha}\}$, and $\overline{\alpha} \leftarrow \alpha_0$.
      else
        $\Upsilon \leftarrow \Upsilon + \{\beta\}$.
      end if
    end if
  end for
  Set $\beta \leftarrow \arg\max_{\beta' \in \Upsilon}\{\overline{\ell}_{\beta'} - \underline{\ell}_{\beta'}\}$, and generate two nodes $\alpha_0 \leftarrow n(a,k,\underline{\ell}_\beta,(\underline{\ell}_\beta + \overline{\ell}_\beta)/2)$ and $\alpha_1 \leftarrow n(a,k,(\underline{\ell}_\beta + \overline{\ell}_\beta)/2,\overline{\ell}_\beta)$;
  Set $IS_{\alpha_i} \leftarrow IS_\alpha + \{\alpha_i\}$, $LB_{\alpha_i} \leftarrow LB_\alpha$, and $\tilde{\Phi}_{\alpha_i} \leftarrow \tilde{\Phi}_{\overline{\alpha}}$, $\forall i = 0,1$; $p_{\alpha_0} \leftarrow \overline{p}$, $p_{\alpha_1} \leftarrow \overline{p}$;
  Set $UT \leftarrow UT + \{\alpha_0,\alpha_1\} - \{\overline{\alpha}\}$.
end if

input $\mathbf{x}_\alpha$ is not a FIFO flow pattern, and at the end of the procedure a bisection method is used to generate two additional search tree nodes based on the last search tree node $\overline{\alpha}$.

Remark EC.5: Υ is used to keep nodes with non-active FIFO constraints and in the bisection method.

EC.5.3. Procedure of fathoming

The following proposed procedure is used to fathom search tree nodes:

Procedure EC.3: FATHOMING ()

for each $\alpha \in UT$ **do**
  if $LB_\alpha \geq UB$ **then**
    $UT \leftarrow UT - \{\alpha\}$.
  end if
end for

EC.5.4. Procedure for retrieving the active NVH constraints

The following proposed procedure is used to retrieve the active NVH constraints for search tree node α:

Procedure EC.4: NVH_INDEXSET_ENUMERATION(α)

Initialize $m \leftarrow 1$.
while $m > 0$ **do**
  Solve MILP problem (43) and obtain an optimal solution $\mathbf{x}^*$ and the TSTT η^*;
  Set $LB_\alpha \leftarrow \eta^*$, $\mathbf{x}_\alpha \leftarrow \mathbf{x}^*$;
  Reset $m = 0$.
  for each link $\alpha \in A \backslash A_S$ and interval $k \in K$ **do**
    if Eq. (EC.1) is satisfied **then**
      $\Phi_\alpha \leftarrow \Phi_\alpha + \{(a,k)\}$ and $m \leftarrow m + 1$.
    end if
  end for
end while

EC.6. Scenario settings of Examples 1-4

EC.6.1. Scenario setting for Example 1

The test network for Example 1 (see Figure EC.2) is modified from Ziliaskopoulos (2000) and contains nine nodes, eleven links, and one OD pair (from Node r to Node s). The original network of Ziliaskopoulos (2000) does not contain Link 11 and is acyclic. The modified network contains a cycle by adding Link 11. The time interval is 10 s. Both the free-flow and backward shock-wave travel times on all links are one interval. The initial state of the network is empty. Table EC.1 provides the maximum occupancy and flow capacity of each link. The flow capacity is 12 veh/interval for Links 2 and 9, and 6 veh/interval for other links except Link 4. Following Ziliaskopoulos (2000), we assume that an accident occurred on Link 4 and that the flow capacity of Link 4 is thus time-variant: 0 for the first four time intervals; 3 for time intervals 5 and 6; and 6 for the remaining time intervals. The traffic demands are eight vehicles for intervals 1 and 3, sixteen for interval 2, and zero for the rest of intervals.

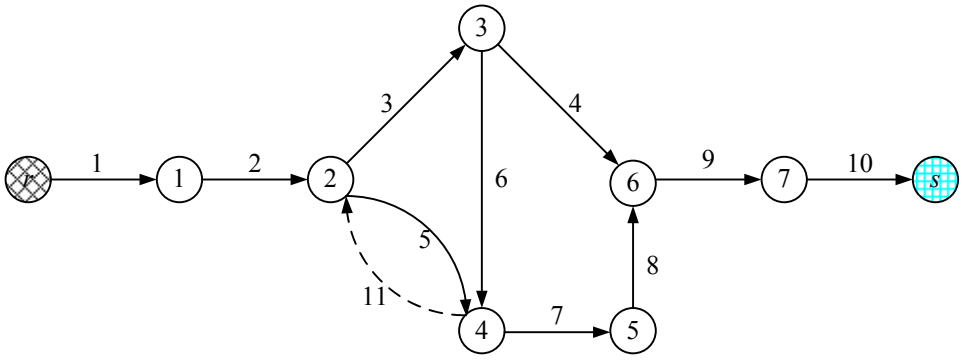

Figure EC.2 The modified Ziliaskopoulos (2000) network for Example 1.

Table EC.1 Setting for the modified Ziliaskopoulos (2000) network.

Link	1	2	3	4	5	6	7	8	9	10	11
N_a	100	20	10	10	10	10	10	10	20	100	10
C_a	12	12	6	-	6	6	6	6	12	12	9

EC.6.2. Scenario setting for Example 2

The X-shaped network for Example 2 (see Figure EC.3) has six nodes, five links, two origins (r_1 and r_2), two destinations (s_1 and s_2), and the two OD pairs (r_1, s_1) and (r_2, s_2). The input parameter values of the LTM for each link are the same: 133 veh/km jam density, 54 km/h free-flow speed, 18 km/h backward shock-wave speed, 1800 veh/h/lane flow capacity, and 10 s for each time interval δ.

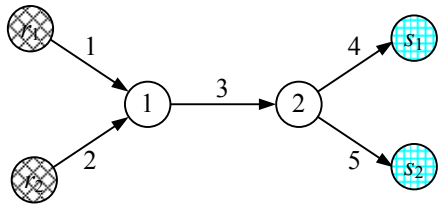

Figure EC.3 An X-shaped network for Example 2.

The link parameters for the X-shaped network are provided in Table EC.2. We assume that accidents occurred on Links 1 and 5. The outflow capacity of Link 1 is 20 veh/interval for the first three intervals and 5 veh/interval for the remainder of the time horizon. The inflow capacity of Link 5 is 10 veh/interval for the first eight intervals and 5 veh/interval for the rest of the time horizon. The traffic demands of OD pair (r_1, s_1) are 20 vehicles for intervals 1 and 2, 10 vehicles for interval 3, and 0 vehicles for other intervals. The traffic demands of OD pair (r_2, s_2) are 10 vehicles for intervals 3 and 4, and 0 vehicles for the other intervals.

Table EC.2 Setting of link parameter values for the X-shaped network in Example 2.

Link	1	2	3	4	5
Length (m)	150	150	300	150	150
Number of lanes	2	2	4	2	2
Maximum link occupancy (veh)	∞	∞	160	∞	∞
Flow capacity (veh interval)	-	10	20	10	-

EC.6.3. Network setting for Example 3

The modified Nguyen and Dupuis network for Example 3 (see Figure EC.4) has 17 nodes, 23 links, and 4 OD pairs. Following Long et al. (2013b), the OD demands are 5 veh/interval for OD pair (r_1, s_1), 10 veh/interval for OD pair (r_1, s_2), and 7.5 veh/interval for OD pairs (r_2, s_1) and (r_2, s_2). The input parameter values of the LTM are the same as those given in Example 2. Links 8-2, 12-8, and 13-3 have one lane, Links 7-8 and 9-13 have two lanes, and the other 14 links have three lanes. The outflow capacities of the links going into nodes 2, 3, 5, 6, 9, 10, and 11 are 3.0 veh/interval/lane. The maximum link occupancies and flow capacities of both origin and destination links are infinite. The lengths of the links in both scenarios are 150 m for both origin and destination links. The lengths of general links in Scenario 1 are provided in Table EC.3.

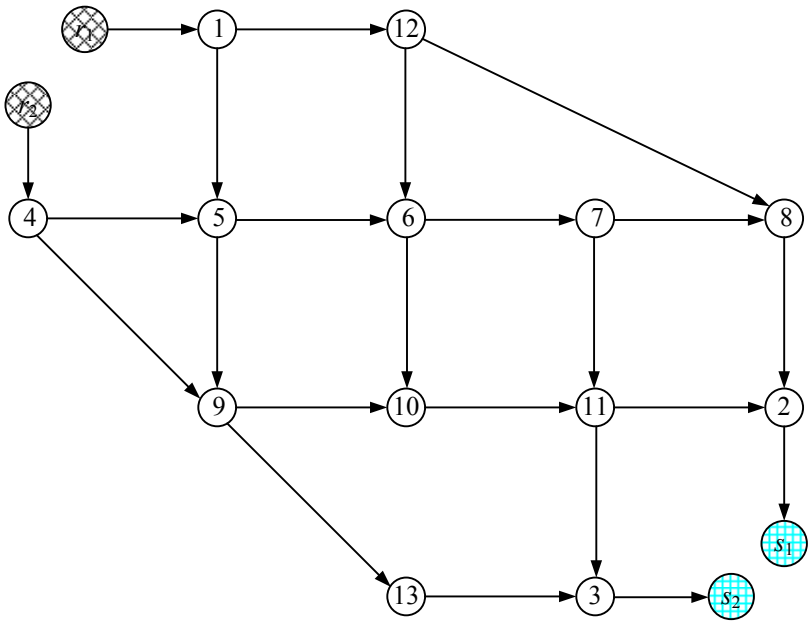

Figure EC.4 The modified Nguyen and Dupuis (1984) network for Example 3.

EC.6.4. Network setting for Example 4

The modified Sioux Falls network for Example 4 (see Figure EC.5) has 43 nodes and 95 links. Following Han (2003), we only consider 12 OD pairs specified in Table EC.4, and use the same

Table EC.3 The length of each link in the modified Nguyen and Dupius network for Scenario 1 in Example 3.

Link length (m)	150	450	600	600	750
Link	r_1-1	1-12	1-5	4-5	12-8
	r_2-4	7-8	4-9	5-6	
	2-s_1	7-11	5-9	6-7	
	3-s_2	9-13	6-10	9-10	
		8-2	11-2	11-3	
		10-11	12-6	13-3	

demand profile for all OD pairs in Figure EC.6. The input parameter values of the LTM for each link are the same: 133 veh/km jam density, 54 km/h free-flow speed, and 18 km/h backward shock-wave speed. The free-flow travel time and capacity of each link are provided in Table EC.5.

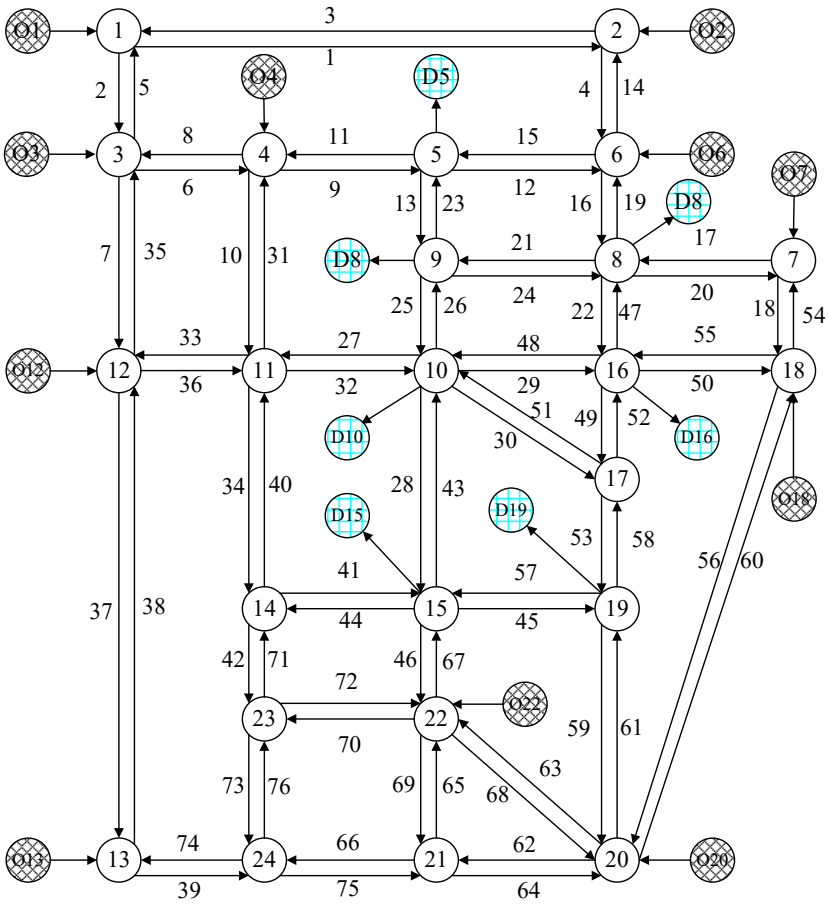

Figure EC.5 The modified Sioux Falls network.

Table EC.4 OD pairs in the modified Sioux Falls network.

Origin	Destination	Origin	Destination
O1	D10	O14	D8
O4	D19	O18	D5
O6	D15	O20	D9
O7	D15	O22	D8
O12	D19	O3	D15
O13	D10	O2	D16

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

Link numbers	$\overline{\tau}_a$ (min)	Capacity (veh min)	Link numbers	$\overline{\tau}_a$ (min)	Capacity (veh min)	Link numbers	$\overline{\tau}_a$ (min)	Capacity (veh min)
1 and 3	6	65	22 and 47	2	45	46 and 67	3	45
2 and 5	2	55	25 and 26	2	45	49 and 52	2	45
4 and 14	2	60	27 and 32	5	50	50 and 55	3	55
6 and 8	5	60	28 and 43	4	45	53 and 58	3	45
7 and 35	5	60	29 and 48	3	40	56 and 60	6	55
9 and 11	3	50	30 and 51	3	45	59 and 61	4	50
10 and 31	5	55	33 and 36	3	60	62 and 64	3	40
12 and 15	3	50	34 and 40	4	50	63 and 68	4	45
13 and 23	2	50	37 and 38	6	65	65 and 69	2	50
16 and 19	3	45	39 and 74	2	60	66 and 75	3	50
17 and 20	3	40	41 and 44	4	50	70 and 72	4	40
18 and 54	5	50	42 and 71	3	40	73 and 76	2	40
21 and 24	3	45	45 and 57	3	40			

EC.7. An example for SO-DTA problems in a cyclic network

In this example, the cyclic network shown in Figure EC.7 is used to illustrate that cyclic flows may not be completely eliminated in optimal solutions to NVH-SO- and MTSTD-NVH-SO-DTA problems. The network has four nodes, four links, and one OD pair (from Node r to Node s). The time interval is 10 s. Both the free-flow travel time and the backward shock-wave travel time on all links are one interval. The initial state for the network is empty. The traffic demands are 10 vehicles for the first 3 intervals and 0 vehicles for the others. The outflow capacities of Link 1 are 10 veh/interval for the first four intervals and 1 veh/interval for the others. The flow capacities for

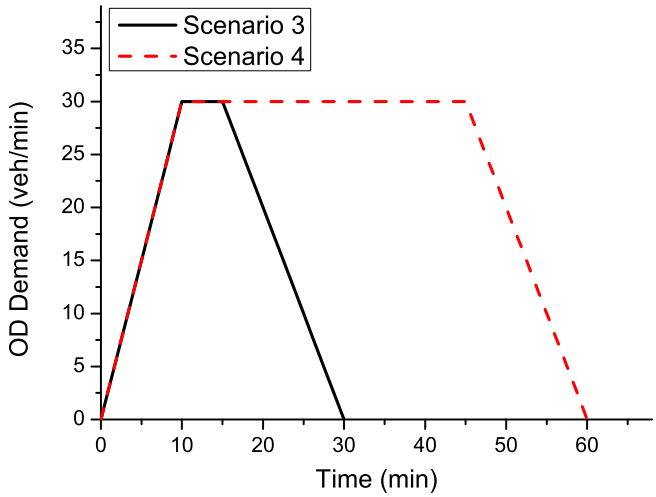

Figure EC.6 Demand profile in the modified Sioux Falls network.

Links 2 and 3 are 10 veh/interval. The inflow capacity of Link 4 is 5 veh/interval.

The optimal cumulative transfer flows between links for the R-SO-, NVH-SO-, and MTSTD-NVH-SO-DTA problems are shown in Table EC.6, where (i, j) indicates the flow from Link i to Link j and R-SO, NVH-SO, and MTSTD-NVH-SO correspond to the optimal solutions to the R-SO-, NVH-SO-, and MTSTD-NVH-SO-DTA models, respectively. We observe that the flows under the columns for (2,3) and (3,2) for the R-SO, NVH-SO, and MTSTD-HVH-SO solutions are positive over some intervals. This means that cyclic flows exist in these optimal solutions. This result further implies that optimal flow patterns to NVH-SO- and MTSTD-NVH-SO-DTA problems may not be acyclic and that other measures must be taken to ensure that the optimal solution is acyclic.

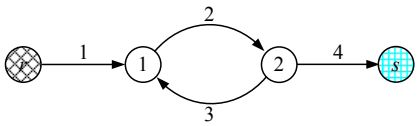

Figure EC.7 A cyclic network.

We also observe from the table that the TSTTs for the three flow patterns are 105 intervals.

Table EC.6 The optimal cumulative transfer flows of various SO-DTA problems in the cyclic network.

Time	R-SO (η =105)				NVH-SO (η =105)				MTSTD-NVH-SO (η =105)				A-NVH-SO (η =108)			
Interval	$(1,2)$	$(2,4)$	$(2,3)$	$(3,2)$	$(1,2)$	$(2,4)$	$(2,3)$	$(3,2)$	$(1,2)$	$(2,4)$	$(2,3)$	$(3,2)$	$(1,2)$	$(2,4)$	$(2,3)$	$(3,2)$
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	7	0	0	0	10	0	0	0	10	0	0	0	10	0	0	0
3	17	5	2	0	20	5	5	0	20	2	2	0	20	5	0	0
4	27	10	2	0	30	10	10	0	27	10	2	0	25	10	0	0
5	28	15	2	0	30	15	15	10	28	15	2	2	26	15	0	0
6	29	20	2	0	30	20	20	15	29	20	2	2	27	20	0	0
7	30	25	2	2	30	25	20	20	30	25	2	2	28	25	0	0
8	30	30	2	2	30	30	20	20	30	30	2	2	29	28	0	0
9	30	30	2	2	30	30	20	20	30	30	2	2	30	29	0	0
10	30	30	2	2	30	30	20	20	30	30	2	2	30	30	0	0

When we removed Link 3 and solved LP problem (17) to obtain an optimal solution (A-NVH-SO), we found that the TSTT increased to 108 intervals, indicating that cyclic flows do help to reduce the TSTT. In this example, the MTSTD-NVH-SO flow is an SO flow pattern and the A-NVH-SO flow is a UE flow pattern. We observe from Table EC.6 that the SO flow pattern assigns two vehicles to a cyclic path to reduce the overall travel time, whereas the UE flow pattern does not.

We also verify that the R-SO solution contains holding flow during interval 2. A maximum number of 10 vehicles can move from Link 1 to Link 2 during this interval, but only 7 vehicles are sent out while 3 vehicles are held on Link 1. Neither the NVH-SO nor the MTSTD-NVH-SO solution contains holding flows. However, the MTSTD-NVH-SO solution has fewer cyclic flows than the NVH-SO solution. This result confirms that 1) the NVH-SO-DTA model may give a solution that has unnecessary cyclic traffic flows and 2) the MTSTD-NVH-SO-DTA model gives an optimal solution that eliminates unnecessary cyclic flows in cyclic networks.

EC.8. Illustrating the steps of the proposed branch-and-bound algorithm

The process of the branch-and-bound algorithm for the FIFO-SO-DTA problem is illustrated in Figure EC.8. A search tree node is represented by $n(a, k, \underline{\ell}, \overline{\ell})$. The property of each search tree node is presented in Table EC.7. The detailed implementation for the first four iterations is given below, and the remaining iterations for the branch-and-bound algorithm for solving the FIFO-SO-DTA

problem are similar:

Iteration 0: Initially, we solved the NVH-SO-DTA problem and found FIFO violations during intervals 6 and 7 (see Table 5). According to the procedure FIFO_BRANCHING, we could generate Branches 1-8. See Remark EC.6 below.

Iteration 1: According to the proposed branching strategy, Branch 2 was selected to continue the branching scheme. We can find that Branch 2 generated a FIFO flow pattern with $\eta = 290$, and hence Branch 2 was fathomed, and updated the upper bound $UB = 290$.

Iteration 2: Branch 3 was selected to continue the branching scheme. Branch 3 generated a flow with $\eta = 290$, which was not less than the upper bound of the TSTT, and thus this branch was fathomed.

Iteration 3: Branch 5 was selected to continue the branching scheme. Branch 5 generated a flow pattern with $\eta = 275$. According to the procedure FIFO_BRANCHING, we could generate Branches 9 and 10 from this branch.

Iteration 4: Branch 6 was selected to continue the branching scheme. Branch 6 generated a flow with $\eta = 315$, and this branch was fathomed.

Table EC.7 The properties of each search tree node.

α	1	2	3	4	5	6	7	8	9	10
IS_α	{1}	{2}	{3}	{1,4}	{1,5}	{1,6}	{1,4,7}	{1,4,8}	{1,6,9}	{1,6,10}
LB_α	270	270	270	270	270	270	270	270	275	275
a_α	2	2	2	2	2	2	2	2	2	2
k_α	6	6	6	7	7	7	7	7	6	6
$\underline{\ell}_\alpha$	2.5	4	0	3	5	0	3	4	2.5	3.25
$\overline{\ell}_\alpha$	4	9	2.5	5	9	3	4	5	3.25	4
η_α	270	290	290	270	275	315	280	270	278.75	290
p_α	100	0.01	0.01	100	0.01	0.01	100	100	100	100
P_α	100	0.01	0.01	200	100.01	100.01	300	300	200.01	200.01

The branch-and-bound algorithm for the NVH-FIFO-SO problem has a similar process and hence is not presented.

Remark EC.6: In Iteration 0, according to the procedure FIFO_BRANCHING, for Link 2 and interval 6, we could directly generate Branches 1-3 by constraints (37)-(39), respectively. According

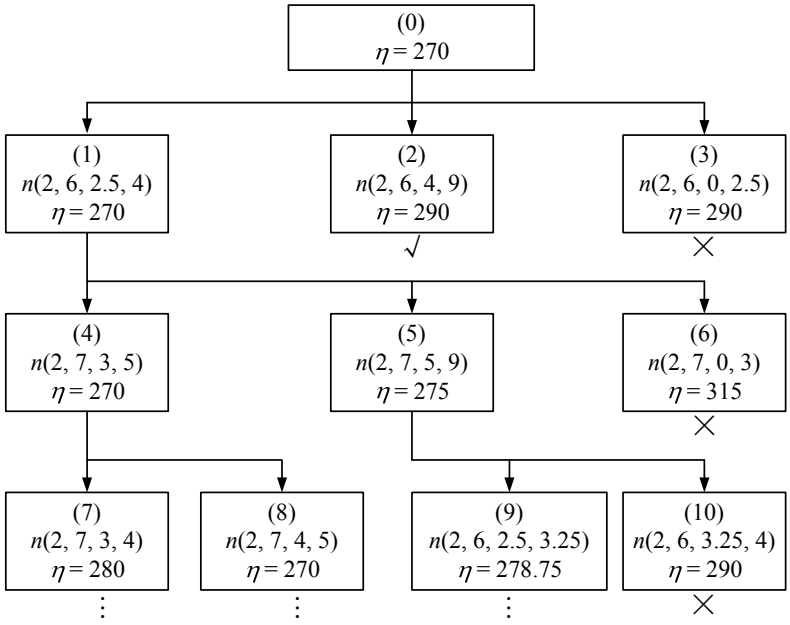

Figure EC.8 Illustration of the branch-and-bound algorithm for the FIFO-SO-DTA problem in Example 2.

to Proposition 8, Branch 1 has the same optimal solution as Branch 0, and we could generate Branches 4-6 for Link 2 and interval 7 based on Branch 1 by using constraints (37)-(39), respectively. Also, according to Proposition 8, Branch 1 has the same optimal solution as Branches 0 and 1, and a bi-section method was used to generate Branches 7 and 8.