A Link-Node Complementarity Model and Solution Algorithm for Dynamic User Equilibria with Exact Flow Propagations

Xuegang (Jeff) Ban*
California Center for Innovative Transportation (CCIT)
Institute of Transportation Studies (ITS)
University of California, Berkeley
2105 Bancroft Way, Suite 300
Berkeley, CA 94720
Tel: (510) 642-5112, Fax: (510) 642-0910
Email: xban@berkeley.edu
*** Corresponding Author**

Henry X. Liu
Department of Civil Engineering
University of Minnesota
122 Civil Engineering Building
500 Pillsbury Drive S.E.
Minneapolis, MN 55455
Tel: (612) 625-6347, Fax: (612) 626-7750
Email: henryliu@umn.edu

Michael C. Ferris
Computer Sciences Department
University of Wisconsin at Madison
1210 West Dayton Street
Madison, Wisconsin 53706
Tel: (608) 262-4281, Fax: (608) 262-9777
Email: ferris@cs.wisc.edu

Submittal to 86th Transportation Research Board Annual Meeting
August 1, 2006

Word Count: 5,703 + 1,500 (1 table + 5 figures) = 7,203

Abstract

We propose a link-node based complementarity formulation for the basic deterministic dynamic user equilibrium problem with single-user-class and fixed demands. In particular, the proposed model captures the exact flow propagation constraints that were usually approximated by previous studies. The solution existence and compactness condition for the proposed model is established under mild assumptions. The model is solved by an iterative algorithm with a relaxed NCP solved accurately and efficiently at each iteration. Therefore, the required number of relaxed NCP solves is fairly small. Numerical examples are also provided in this paper to illustrate the proposed model and solution approach.

1. Introduction

Dynamic traffic assignment (DTA) which can predict, in a short-term fashion, the future dynamic traffic states, has been extensively studied for decades, particularly accelerated in the last fifteen years since the emergence of the intelligent transportation systems (ITS). In this paper, we are interested in the so-called dynamic user equilibrium (DUE) which is the fundamental yet most challenging problem of DTA. Two distinct approaches have dominated the methodologies applied to the DTA research: the simulation-based (microscopic/mesoscopic) and the analytical (macroscopic) approach. In this paper, we focus on the analytical DTA models, especially the variational inequality (VI) method which is more capable of computing dynamic network equilibria than the constrained optimization approach (*1-3*).

VI has been applied for long for modeling various traffic interactions for static traffic assignment problems (*4*, *5*). It had not been used to model DTA until Friesz et al. (*6*). Later on, Ran and Boyce (*1*, *7*) extensively studied the issues of applying VI to formulate and solve DTA problems. The VI approach has also been used for DTA study by Lo et al. (*8*), Ran et al. (*9*), Heydecker and Verlander (*10*), to name just a few. In particular, Friesz et al. (*11*) and Friesz and Mookherjee (*12*) developed the differential variational inequality (DVI) technique to model and solve DUE in the continuous time domain. Although formulated continuously in the temporal domain, most DUE models, e.g. those by Friesz et al. (*6*) and Ran and Boyce (*1*), were solved by the time discretization since to date solving the continuous-time DUE model directly for practical transportation networks is still not feasible. However, the discretized models were only treated as part of the solution procedure without rigorous investigations on their mathematical properties such as the solution existence conditions. Chen and Heush (*13*) were among the first to investigate explicitly on the discrete-time VI model for DTA. Bliemer (*14*) and Bliemer and Bovy (*15*) further improved the model by Chen and Heush (*13*) and proposed the link-route based quasi-variational inequality (QVI) formulation. Lo and Szeto (*16*) integrated the cell transmission model (CTM) into DUE which was formulated as a route based VI problem. Due to the nature of CTM (*17*, *18*), the model by Lo and Szeto (*16*) is discretized in both temporal and spatial domains.

As one special case of VI, the nonlinear complementarity problem (NCP) has been fully studied in the mathematical programming community (*19* and references therein). Efficient solution approaches have been developed during the last decade for solving large scale NCPs (*20*-*22*). Generally, solving an NCP is much easier than a regular VI. However to date, NCP has not been widely applied in modeling and solving DUE. The only study, to the best of the author's knowledge, is Wie et al. (*23*) which formulated the discrete-time DUE with departure time choice as an NCP. It was also pointed out in Wie et al. (*23*) that continuous-time and discrete-time DUE models are significantly different. Generally, the former are infinite dimensional mathematical programming problems while the latter are finite dimensional mathematical programming problems. The NCP based DUE model by Wie et al. (*23*), nevertheless, projected the link exit flows to two neighboring time grids. In this sense, the exact flow propagation of DUE (*24*) was not fully respected. Further, the linear programming based solution approach in Wie et al. (*23*) is similar to the Frank-Wolfe (FW) algorithm. Due to the well-known convergence problem of FW, such a method may not be effective for solving DUE, especially for producing accurate solutions.

In this paper, we formulate the discrete-time DUE problem with exact flow propagations as a link-node based NCP. We consider the basic discrete-time DUE problem which is deterministic, single-user-class with fixed travel demands. We first start with the continuous-time DUE and formulate it as an infinite-dimensional mixed complementarity problem (MiCP) with side constraints. Based on this MiCP model, we adopt the discretization scheme by Astarita (*24*) and prove that an NCP formulation exists for discrete-time DUE with exact flow propagations. We further prove that the solution set of the proposed NCP is nonempty and compact, a sharper result compared with that in Wie et al. (*23*). To solve the NCP, we develop an iterative algorithm with a relaxed NCP solved at each iteration. Due to the exact solve of each relaxed NCP, the solution process requires much less iterations than previous methods, which is demonstrated by the case study conducted in this paper.

This paper is organized as follows. The continuous time DUE formulation is first discussed in Section 2, with a MiCP formulation provided. In Section 3, the method for deriving the discrete-time DUE model is presented with exact flow propagations. The derivation of the NCP model and its solution existence condition is established. Section 4 mainly presents the solution algorithm of the proposed model, including the network loading process and two gap functions to monitor the convergence of the algorithm. Numerical examples are provided in Section 5, followed by the concluding remarks and future research directions in Section 6.

2. Continuous time DUE model

In this section, we introduce the link-node based continuous-time formulation for DUE. Friesz et al. (*6*), Ran and Boyce (*1*), and Bliemer and Bovy (*15*) have extensively studied the path-based and link-based continuous-time DUE models.

Assume a given transportation network can be represented as a connected and directed graph, denoted as $G(N, A)$, where N is the set of nodes and A is the set of links (arcs). Since we are dealing with dynamic (or time-varying) traffic flows, we denote $t \in [0, T']$ as the continuous time and T' is the total study period. Also denote R and S as the origin and destination node set, respectively. Throughout this paper, we will use index $a \in A$ to denote a link, index $i \in N$ or $j \in N$ to denote a node, and index $s \in S$ to denote a destination. Moreover, we only consider the basic DUE problem, i.e., the deterministic and single-user-class DUE with fixed demands, for which $d_{is}(t)$ denotes the (fixed) travel demand rate from node i to destination s at time instant t. We conventionally set $d_{ss}(t) = 0, \forall s \in S, t \in [0, T']$.

2.1 DUE Condition

The link-node model is derived from the so-called (link-node based) DUE condition which describes the optimality condition of the DUE problem. The DUE condition is a dynamic extension to the Wardrop's first principle (*25*) for the static case and can be stated as follows:

If, from each decision node to every destination node at each instant of time, the actual travel times for all the routes that are being used are equal and minimal, then the dynamic traffic flow over the network is in a travel-time based dynamic user equilibrium (DUE) state.

In this condition, a "decision node" with respect to a given destination node means any node in the network which either generates OD trips (i.e., origin nodes) or is traversed by flows heading to the destination (i.e., the intermediate nodes). Hence, we require that a destination node must not be a decision node of itself. Mathematically, the DUE condition can be expressed as follows:

$$0 \le u_{as}(t) \perp \{\tau_a(t) + \pi_{h_a s}[t + \tau_a(t)] - \pi_{l_a s}(t)\} \ge 0, \forall a \in A, s \in S, t \in [0, T']. \quad (1)$$

Here $u_{as}(t)$ denotes the inflow rate to link *a* with respect to destination *s* at time instant *t*, $\pi_{l_a s}(t)$ and $\pi_{h_a s}(t)$ denote, respectively, the minimum travel time from the tail node (l_a) and head node (h_a) of link *a* to destination *s* at time *t* and $\tau_a(t)$ is the link travel time for link *a* at time instant *t*. We conventionally set $\pi_{ss}(t) = 0, \forall s \in S, t \in [0, T']$. In addition, "$\perp$" in (1) means "perpendicular" so that $x \perp y \Leftrightarrow x^T y = 0$. Note that the DUE condition can also be conveniently expressed in a Dynamic Programming (DP) manner. For details, one can refer to Han and Heydecker (*26*).

2.2 Dynamic Network Constraints

The dynamic network constraints describe the defining set of the DUE problem, which must be satisfied by any feasible solution. Five types of constraints have been identified in the literature (*1*, *15*), namely the mass balance constraints, the flow conservation constraints, the flow propagation constraints, First-in-first-out (FIFO) constraints, and other definitional constraints. In the following, we simply list these constraints without further discussions. For detailed descriptions, one can refer to Ran and Boyce (*1*) and Bliemer and Bovy (*15*).

2.2.1 Mass Balance Constraints

Mass balance constraints, as shown in equation (2), define the relationship between the link flow on a given link *a* with respect to destination *s* at time *t*, denoted as $x_{as}(t)$, and the inflow rate and exit flow rate for the same link to the same destination at the same time instant (denoted as $u_{as}(t)$ and $v_{as}(t)$, respectively).

$$\frac{dx_{as}(t)}{dt} = u_{as}(t) - v_{as}(t), \forall a \in A, s \in S, t \in [0, T'] \quad (2)$$

Equation (2) implies that $x_{as}(t) = \int_0^t [u_{as}(w) - v_{as}(w)]dw + x_{as}(0), \forall a \in A, s \in S, t \in [0, T']$. We further assume the initial condition as

$$x_{as}(0) = 0, \forall a \in A, s \in S. \quad (3)$$

Then, the mass balance constraints (2) can be rewritten as:

$$x_{as}(t) = \int_0^t [u_{as}(w) - v_{as}(w)]dw, \forall a \in A, s \in S, t \in [0, T']. \quad (4)$$

2.2.2 Flow Conservation Constraints

The flow conservation constraints require that for a given destination at any given time instant, the flows entering any node, together with the demands generated at this node, must all exit from this node unless this node is the destination itself. Mathematically, they can be expressed as:

$$\sum_{a\in A(i)} u_{as}(t) = d_{is}(t) + \sum_{a\in B(i)} v_{as}(t), \forall i \in N, s \in S, i \neq s, t \in [0,T'] , \quad (5)$$

where $A(i)$ is the set of links whose tail nodes are i and $B(i)$ is the set of links whose head nodes are i.

2.2.3 Flow Propagation Constraints

These constraints describe the consistent evolvement of traffic flows in both temporal and spatial domains. It has been proved that in a continuous time fashion, the flow propagation constraints can be represented as (*24*):

$$v_{as}[t'+\tau_a(t')] = \frac{u_{as}(t')}{1+d\tau_a(t')/dt'}, \forall a \in A, s \in S, t' \in [0,T'] . \quad (6)$$

Note that equation (6) can be easily obtained by differentiating both sides of the Type I flow propagation constraints in Ran and Boyce (*1*, page 74, equation (4.22)). Equation (6) can be rewritten as

$$v_{as}(t) = \int_{t'\in[0,T']} \lambda_a(t',t) u_{as}(t') d\mu(t'), \forall a \in A, s \in S, t \in [0,T'] . \quad (7)$$

Here $\lambda_a(t',t)$ is denoted as the indicator function defined as follows:

$$\lambda_a(t',t) = \begin{cases} \dfrac{1}{1+d\tau_a(t')/dt'}, & \text{if } t = t'+\tau_a(t') \\ 0, & \text{otherwise} \end{cases} . \quad (8)$$

From (8), it is obvious that $\lambda_a(t',t)$ is not continuous. Therefore, the integral in (7) is a Lebesgue integral and $\mu(t')$ is the corresponding Lebesgue measure. The indicator function defined in (8) is similar to the "dynamic effective flow rate factor" proposed by Bliemer (*14*). However, our definition here is link-based and thus much simpler compared with that in Bliemer (*14*). Also, since we assume vehicles to different destinations experience the same travel time as long as they enter the link at the same time, $\lambda_a(t',t)$ is independent of individual destinations.

2.2.4 FIFO Constraints

FIFO constraints were first introduced into the DTA study by Carey (*27*). Since then, it has been assumed to be a "discipline" respected by the dynamic traffic flows. FIFO requires that any vehicle entering into a link earlier must also exit from the link earlier. Ran and Boyce (*1*) have shown that in order for FIFO to hold, an extra restriction should be imposed on $\tau_a(t)$ as $d\tau_a(t)/dt > -1$. However, explicitly imposing such a constraint will dramatically increase the complexity of the resulting model. Therefore, most of the DUE models tend to implicitly guarantee FIFO by choosing a proper link performance function. How to design such a function, however, is beyond the scope of this paper. Here we conventionally assume $\tau_a(t)$ is a function of the link flow at time *t* on link *a*:

$$\tau_a(t) = g(x_a(t)). \tag{9}$$

Here $x_a(t)$ is the total link flow on link *a* at time *t* (see equation (10) below). The function in (9) satisfies FIFO when *g* is linear, or if the gradient of *g* with respect to *x* is bounded from above when *g* is nonlinear. For more discussions, one can refer to Nie and Zhang (*28*), Carey et al. (*29*), and Xu et al. (*30*). Note that such a function is only suitable for moderately congested networks (*31*). More sophisticated functional forms should be adopted for networks with heavy congestion.

2.2.5 Other Definitional Constraints:

Other definitional constraints are listed in equations (10) and (11) as follows.

$$\begin{cases} u_a(t) = \sum_{s\in S} u_{as}(t) \\ v_a(t) = \sum_{s\in S} v_{as}(t), \forall a \in A, t \in [0, T'] \\ x_a(t) = \sum_{s\in S} x_{as}(t) \end{cases}, \tag{10}$$

$$\begin{cases} u_{as}(t) \ge 0, v_{as}(t) \ge 0, x_{as}(t) \ge 0, \forall a \in A, s \in S, t \in [0, T'] \\ \pi_{is}(t) \ge 0, \forall i \in N, s \in S, i \ne s, t \in [0, T'] \end{cases}, \tag{11}$$

Here $u_a(t), v_a(t)$ denote, respectively, the total inflow rate and exit flow rate of link *a* at time *t*. Together with $x_a(t)$, they are referred as "aggregated" variables, while $u_{as}(t)$, $v_{as}(t)$ and $x_{as}(t)$ are called the "disaggregated" variables. For aggregated and disaggregated variables that satisfy (10), we call them "corresponding" to each other.

2.3 MiCP Formulation

Equations (1), (4), (5), (7) and (10) – (11) constitute the DUE model defined on disaggregated variables $x = (x_{as}(t))_{\forall a,s,t}, u = (u_{as}(t))_{\forall a,s,t}$, $v = (v_{as}(t))_{\forall a,s,t}$ and $\pi = (\pi_{is}(t))_{\forall i,s,t;i\ne s}$; whereas $\lambda = (\lambda_a(t',t))_{\forall a,t,t'}$ and $\tau = (\tau_a(t))_{\forall a,t}$ can be treated as functions of these defining variables. However, such a DUE model can be further simplified. Firstly, from equations (4) and (7), *x* and *v* can be readily represented by *u* and λ. Then we can model DUE using *u* and π only, while both λ and τ are functions of *u*. In this manner, the only significant equations are (1), (5), the nonnegativity constraints on *u* and π in (11), and the definitions of λ and τ in (8) and (9). As a result, we have the following model for DUE: trying to find (*u*, π) such that the following is satisfied

$$\begin{cases} 0 \le u_{as}(t) \perp \{\tau_a(t) + \pi_{h_a s}[t+\tau_a(t)] - \pi_{l_a s}(t)\} \ge 0, \forall a \in A, s \in S, t \in [0, T'], & \text{(12a)} \\ \sum_{a \in A(i)} u_{as}(t) = d_{is}(t) + \sum_{a \in B(i)} \int_0^t \lambda_a(t', t) u_{as}(t') d\mu(t'), \forall i \in N, s \in S, i \ne s, t \in [0, T'], & \text{(12b)} \\ \pi_{is}(t) \ge 0, \forall i \in N, s \in S, i \ne s, t \in [0, T'] & \text{(12c)} \end{cases}$$

, whereas λ and τ are defined in equations (8) and (9) respectively. Variables x and v are indeed intermediate and can be expressed by u through equations (4) and (7). Clearly, (12a) just repeats (1), and (12b) and (12c) correspond to (5) and the nonnegativity on π in (10). Note that (12a) and (12b) define an infinite dimensional MiCP (*32*), while (12c) imposes a side constraint which requires the minimum travel time from node i to destination s at time t must be nonnegative. Solving the infinite dimensional MiCP (12) with side constraints is generally difficult and we thus study the discretized problem starting from the next section.

3. Discrete time DUE model

3.1 Discrete Time DUE with Exact Flow Propagation

In order to obtain the discrete-time model, we can evenly divide the entire study period into K' time intervals by introducing the length of each time interval Δ such that $K'\Delta = T'$. The notation at discrete-time is first listed as follows:

$u_{as}^k = u_{as}(k\Delta)$: the inflow rate to link a towards destination s at time interval k, $u = (u_{as}^k)_{\forall a,s,k}$

$v_{as}^k = v_{as}(k\Delta)$: the exit flow rate from link a towards destination s at time interval k, $v = (v_{as}^k)_{\forall a,s,k}$

$u_{as}^k \Delta$: the inflows to link a towards destination s during time interval k

$v_{as}^k \Delta$: the exit flows from link a towards destination s during time interval k

$x_{as}^k = x_{as}(k\Delta)$: the flows of link a towards destination s at time interval k, $x = (x_{as}^k)_{\forall a,s,k}$

$u_s^k = \sum_{s \in S} u_{as}^k$: the aggregated inflow rate to link a at time interval k, $u^A = (u_a^k)_{\forall a,k}$

$v_a^k = \sum_{s \in S} v_{as}^k$: the aggregated exit flow rate from link a at time interval k, $v^A = (v_a^k)_{\forall a,k}$

$x_a^k = \sum_{s \in S} x_{as}^k$: the aggregated flows of link a towards destination s at time interval k, $x^A = (x_a^k)_{\forall a,k}$

$\tau_a^k(u) = \tau_a(k\Delta)$: the travel time of link a at time interval k, a function of u, $\tau = (\tau_a^k)_{\forall a,k}$

$\pi_{is}^k = \pi_{is}(k\Delta)$: the minimum travel time from node i to destination s at time interval k, $\pi = (\pi_{is}^k)_{\forall i,s,k,i \ne s}$

$d_{is}^k = d_{is}(k\Delta)$: the demand rate generated from node i to destination s at time interval k, $d = (d_{is}^k)_{\forall i,s,k,i \ne s}$

$d_{is}^k \Delta$: the demands generated from node i to destination s during time interval k

$e_a^k(u) \equiv (k-1)\Delta + \tau_a^k(u)$: the exit time for vehicles *e*ntering a at the start of time interval k, a function of u, $e = (e_a^k)_{\forall a,k}$

Using this simple discretization scheme, one should be cautious about how to discretize (12a) and 12(b). First of all, (12b) is related to the flow conservation (5) and flow propagation (6). Assuming $(k-1)\Delta = t$, (5) can be easily discretized as

$$\sum_{a\in A(i)} u_{as}^{k} = d_{is}^{k} + \sum_{a\in B(i)} v_{as}^{k}, \forall i \in N, s \in S, i \neq s, k = 1,2,\cdots,K' \cdot \tag{13}$$

According to (6), the exit flow rate v_{as}^{k} is related to inflow rate $u_{as}^{k'}$ if $e_a^{k'}(u) = (k'-1)\Delta + \tau_a^{k'} = k\Delta$. However, such an integer k' may not exist since travel time $\tau_a^{k'}$ is real valued. In this paper, we adopt the discretization method in Astarita (*24*) to address this problem. The method first constructs the pair of $(e_a^{k'}, v_{as}(e_a^{k'}))$ for any inflow at k' and we then have, by discretizing (6),

$$v_{as}(e_a^{k'}) = u_{as}^{k'} \cdot \lambda_a^{1,k'}(u), \forall a \in A, s \in S, k' = 1,2,\cdots,K', \tag{14}$$

and

$$\lambda_a^{1,k'}(u) = \frac{\Delta}{\tau_a^{k'+1}(u) - \tau_a^{k'}(u) + \Delta}. \tag{15}$$

Note that $\lambda_a^{1,k'}$ is a function of *u* since $\tau_a^{k'}$ is so. Then for any *k*, we try to find k' such that $e_a^{k'} \le k\Delta < e_a^{k'+1}$, as shown in Figure 1.

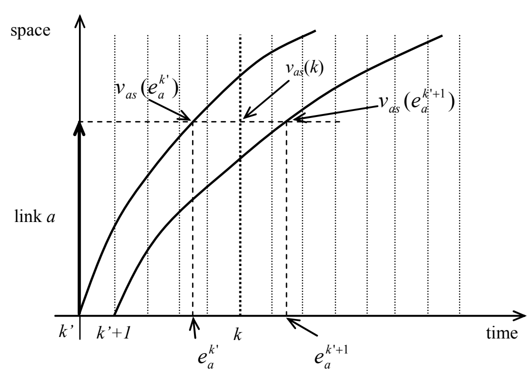

Figure 1 Discretization of flow propagation constraints

In Figure 1, the solid line represents the trajectory of traffic flows entering link *a* at time k' or $k'+1$. The exit flow rate at time interval *k* can be computed using linear interpolation of $v_{as}(e_a^{k'})$ and $v_{as}(e_a^{k'+1})$ as:

$$v_{as}^{k} = \sum_{k': e_a^{k'} \le k\Delta < e_a^{k'+1}} \lambda_a^{2,k',k}(u) \cdot v_{as}(e_a^{k'}) + (1 - \lambda_a^{2,k',k}(u)) \cdot v_{as}(e_a^{k'+1}), \tag{16}$$

where $\lambda_a^{2,k',k}(u)$ is defined as:

$$\lambda_a^{2,k',k}(u) = \frac{e_a^{k'+1} - k\Delta}{e_a^{k'+1} - e_a^{k'}} = \frac{\tau_a^{k'+1}(u) + (k'-k)\Delta}{\tau_a^{k'+1}(u) - \tau_a^{k'}(u) + \Delta}. \tag{17}$$

Substitute equations (14) – (17) to (13), we obtain the discretized version of (12b) as

$$\sum_{a\in A(i)} u_{as}^k = d_{is}^k + \sum_{a\in B(i)} \sum_{k': e_a^{k'} \le k\Delta < e_a^{k'+1}} [\lambda_a^{1,k'}(u)\cdot \lambda_a^{2,k',k}(u)\cdot u_{as}^{k'} + \lambda_a^{1,k'+1}(u)\cdot(1-\lambda_a^{2,k',k}(u))\cdot u_{as}^{k'+1}], \forall i\in N, s\in S, i\ne s, k=1,\cdots,K' . \quad (18)$$

Similarly, to discretized $\pi_{h_a s}[t+\tau_a(t)]$ in (12a), we project it to the two neighboring grids of $e_a^k = (k-1)\Delta + \tau_a^k$, denoted as l and l+1, such that $l\Delta \le e_a^k < (l+1)\Delta$. In other words,

$$\pi_{h_a s}[t+\tau_a(t)] = \pi_{h_a s}(e_a^k) = \lambda_a^{3,k,l}(u)\cdot \pi_{h_a s}^l + [1-\lambda_a^{3,k,l}(u)]\cdot \pi_{h_a s}^{l+1} . \quad (19)$$

Here we define

$$\lambda_a^{3,k,l}(u) = (l+2-k-\frac{\tau_a^k(u)}{\Delta}) . \quad (20)$$

Some observations thus follow. First, all $\lambda^1 = (\lambda_a^{1,k'})_{\forall a,k'}, \lambda^2 = (\lambda_a^{2,k',k})_{\forall a,k',k; e_a^{k'} \le k\Delta < e_a^{k'+1}}, \lambda^3 = (\lambda_a^{3,k,l})_{a,k,l; l \le e_a^k/\Delta < l+1}$ and e are functions of τ which is itself a function of the aggregated link inflow rate vector u^A. Due to the relation between aggregated and disaggregated variables in (10), the above four functions are also defined on the disaggregated link inflow vector u. Furthermore, since we assume the link travel time defined in (9) satisfies FIFO, $\lambda_a^{1,k'} > 0, \forall a,k'$. Meanwhile, from the definition of λ^2 and λ^3, we can easily observe $0 < \lambda_a^{2,k',k} \le 1$, $\forall a,k',k; e_a^{k'} \le k\Delta < e_a^{k'+1}$ and $0 < \lambda_a^{3,k,l} \le 1, \forall a,k,l; l \le e_a^k/\Delta < l+1$.

Finally substituting (19) to (12a), we have the following discretized DUE model with exact flow propagations:

$$\begin{cases} 0 \le u_{as}^k \perp \{\tau_a^k(u) + \sum_{l \le e_a^k(u)/\Delta < l+1} \lambda_a^{3,k,l}(u)\cdot \pi_{h_a s}^l + [1-\lambda_a^{3,k,l}(u)]\cdot \pi_{h_a s}^{l+1} - \pi_{l_a s}^k\} \ge 0, \forall a\in A, s\in S, k=1,\cdots K' & (21a) \\ \sum_{a\in A(i)} u_{as}^k = d_{is}^k + \sum_{a\in B(i)} \sum_{k': e_a^{k'}(u) \le k\Delta < e_a^{k'+1}(u)} [\lambda_a^{1,k'}(u)\cdot \lambda_a^{2,k',k}(u)\cdot u_{as}^{k'} + \lambda_a^{1,k'+1}(u)\cdot(1-\lambda_a^{2,k',k}(u))\cdot u_{as}^{k'+1}], \forall i\in N, s\in S, i\ne s, k=1,\cdots,K' & (21b) \\ \pi_{is}^k \ge 0, \forall i\in N, s\in S, i\ne s, k=1,\cdots,K' & (21c) \end{cases}$$

Note that (21) is a MiCP with side constraints (21c). Also since u is the defining variable, $e_a^{k'}(u)$ and $e_a^k(u)$ will both change as u does. This implies that the summation terms in the right hand sides of both (21a) and (21b) are not fixed; rather they change as u does. In this sense, model (21) is not close form.

We next show that (21) has an equivalent NCP formulation.

Theorem 1 If the link travel time function $\tau_a^k(u)$ is positive for any $a\in A, k=1,\cdots K'$ and $u \ge 0$, then model (21) is equivalent to the following NCP model: *find* (u,π) *such that*

$$\begin{cases} 0 \le u_{as}^k \perp \{\tau_a^k(u) + \sum_{l \le e_a^k(u)/\Delta < l+1} \lambda_a^{3,k,l}(u)\cdot \pi_{h_a s}^l + [1-\lambda_a^{3,k,l}(u)]\cdot \pi_{h_a s}^{l+1} - \pi_{l_a s}^k\} \ge 0, \forall a\in A, s\in S, k=1,\cdots K', & (22a) \\ 0 \le \pi_{is}^k \perp (\sum_{a\in A(i)} u_{as}^k - d_{is}^k - \sum_{a\in B(i)} \sum_{k': e_a^{k'}(u) \le k\Delta < e_a^{k'+1}(u)} [\lambda_a^{1,k'}(u)\cdot \lambda_a^{2,k',k}(u)\cdot u_{as}^{k'} + \lambda_a^{1,k'+1}(u)\cdot(1-\lambda_a^{2,k',k}(u))\cdot u_{as}^{k'+1}] \ge 0, \forall i\in N, s\in S, i\ne s, k=1,\cdots K'. & (22b) \end{cases}$$

Proof. We need to prove (***a***) if (u,π) solves (21), then it is also a solution to (22); and (***b***) if (u,π) solves (22), then it must also solves (21).

(***a***) is straightforward. If (u,π) solves (21), then it is obvious that (u,π) also solves (22).

In order to prove (***b***), suppose (u,π) solves (22). Since (u,π) is given here, $e_a^{k'}(u)$ and $e_a^{k}(u)$ are fixed for any $a \in A, k', k = 1, \cdots K'$. This implies that both (21) and (22) become close form. For the sake of contradiction, assume (u,π) is not a solution of (21). Then we must have $i \in N, s \in S, i \neq s$ at some time interval *k*, such that $\sum_{a\in A(i)} u_{as}^k - d_{is}^k - \sum_{a\in B(i)} \sum_{k': e_a^{k'}(u) \le k\Delta < e_a^{k'+1}(u)} [\lambda_a^{1,k'}(u)\cdot\lambda_a^{2,k',k}(u)\cdot u_{as}^{k'} + \lambda_a^{1,k'+1}(u)\cdot(1-\lambda_a^{2,k',k}(u))\cdot u_{as}^{k'+1}] > 0$. Hence we have $\pi_{is}^k = 0$ due to (22b). This implies $\pi_{l_a s}^k = \pi_{is}^k = 0$ for any link $a \in A(i)$. On the other hand, Since $\lambda^1 > 0, 0 < \lambda^2 \le 1$ and $u_{as}^{k'} \ge 0$, $u_{as}^{k'+1} \ge 0$, $d_{is}^k \ge 0$, we have $\sum_{a\in A(i)} u_{as}^k > d_{is}^k \ge 0$. Thus, we must have at least one link $a \in A(i)$ at time interval *k*, such that $u_{as}^k > 0$. Then, we have $\tau_a^k(u) + \sum_{l \le e_a^k(u)/\Delta < l+1} \lambda_a^{3,k,l}(u)\cdot\pi_{h_a s}^l + [1-\lambda_a^{3,k,l}(u)]\cdot\pi_{h_a s}^{l+1} - \pi_{l_a s}^k = 0$ due to (22a). This means $\tau_a^k(u) + \sum_{l \le e_a^k(u)/\Delta < l+1} \lambda_a^{3,k,l}(u)\cdot\pi_{h_a s}^l + [1-\lambda_a^{3,k,l}(u)]\cdot\pi_{h_a s}^{l+1} = 0$ since $\pi_{l_a s}^k = 0$. Nevertheless, $\tau_a^k(u) > 0$, $\pi_{h_a s}^l \ge 0$, $\pi_{h_a s}^{l+1} \ge 0$, and $0 < \lambda^3 \le 1$, this is a contradiction. □

Note that in NCP (22), *u* is the defining variable, and $\lambda^1, \lambda^2, \lambda^3$, *e* and τ are functions defined on *u*. However, (22) is not close form due to the same reason for model (21). We further define two vector functions

$$F_u(u,\pi) = \left(\tau_a^k(u) + \sum_{l \le e_a^k(u)/\Delta < l+1} \lambda_a^{3,k,l}(u)\cdot\pi_{h_a s}^l + [1-\lambda_a^{3,k,l}(u)]\cdot\pi_{h_a s}^{l+1} - \pi_{l_a s}^k \right)_{\forall a\in A, s\in S, k=1,\cdots K'}, \tag{23}$$

$$F_\pi(u,\pi) = \left(\sum_{a\in A(i)} u_{as}^k - d_{is}^k - \sum_{a\in B(i)} \sum_{k': e_a^{k'}(u) \le k\Delta < e_a^{k'+1}(u)} [\lambda_a^{1,k'}(u)\cdot\lambda_a^{2,k',k}(u)\cdot u_{as}^{k'} + \lambda_a^{1,k'+1}(u)\cdot(1-\lambda_a^{2,k',k}(u))\cdot u_{as}^{k'+1}] \right)_{\forall i\in N, s\in S, i\neq s, k=1,\cdots K'} \tag{24}$$

We can then compactly express (22) as follows.

$$\begin{cases} 0 \le u \perp F_u(u,\pi) \ge 0 \\ 0 \le \pi \perp F_\pi(u,\pi) \ge 0 \end{cases}. \tag{25}$$

We denote model (22) and (25) as *NCPDUE* thereafter in this paper.

3.2 Solution Existence and Compactness Condition

This section establishes the solution existence condition for *NCPDUE*. For this purpose, we need a solution existence condition for VIs, as stated in Lemma 1.

Lemma 1 Let $K \subseteq R^n$ be compact (closed and bounded) and convex and let $F: K \to R^n$ be continuous. Then the solution set of VI defined by *K* and *F* is nonempty and compact.

Proof. The proof can be found in Facchinei and Pang (*19*) in Corollary 2.2.5. □

The solution existence result discussed here can then be summarized as follows.

Theorem 2 Suppose *a)* the link travel time function $\tau_a^k(u)$ is positive for any $a \in A, k = 1, \cdots K'$ and bounded from above for any finite *u*, *b)* the given OD demand d_{is}^k is nonnegative and bounded from above for any $i \in N, s \in S, i \neq s, k = 1, \cdots K'$, *c)* λ^1 is bounded from above, and *d)* $F_u(u, \pi)$ and $F_\pi(u, \pi)$ are continuous with respect to (u, π). Then the solution set of *NCPDUE* is nonempty and compact.

Proof. First of all, choose three scalars as follows.

$$\alpha_d > \max_{s \in S} \max_{i \in N, i \neq s} \max_{k=1,\cdots,K'} d_{is}^k \tag{26a}$$

$$\alpha_u > \max_{s \in S} \max_{i \in N, i \neq s} \max_{k=1,\cdots,K'} \max_{u \geq 0} (d_{is}^k + \sum_{a \in B(i)} \sum_{k': e_a^{k'}(u) \leq k\Delta < e_a^{k'+1}(u)} [\lambda_a^{1,k'}(u) \cdot \lambda_a^{2,k',k}(u) \cdot u_{as}^{k'} + \lambda_a^{1,k'+1}(u) \cdot (1 - \lambda_a^{2,k',k}(u)) \cdot u_{as}^{k'+1}]) \tag{26b}$$

$$\alpha_\pi > \max_{s \in S} \max_{a \in A} \max_{k=1,\cdots,K'} \max_{u \geq 0, \pi \geq 0} (\tau_a^k(u) + \sum_{l \leq e_a^k(u)/\Delta < l+1} \lambda_a^{3,k,l}(u) \cdot \pi_{h_a s}^l + [1 - \lambda_a^{3,k,l}(u)] \cdot \pi_{h_a s}^{l+1}) \tag{26c}$$

These three scalars exist because of the three boundedness assumptions and the fact that $0 < \lambda_a^{2,k',k} \leq 1$, $\forall a, k', k; e_a^{k'}(u) \leq k\Delta < e_a^{k'+1}(u)$ and $0 < \lambda_a^{3,k,l} \leq 1, \forall a, k, l; l \leq e_a^k(u)/\Delta < l+1$. For the same reason, they are all positive. We further define a set $E = \{y = (u, \pi) \geq 0 \,|\, u \leq \alpha_u 1,\ \pi \leq \alpha_\pi 1\}$ and a function $F = (F_u(u, \pi),\ F_\pi(u, \pi))$. Here **1** is a vector with all components being 1 with the proper dimension. Set *E* and function *F* constitute a VI which *tries to find* $y^* = (u^*, \pi^*) \in E$ *such that*

$$F(y^*)^T (y - y^*) \geq 0, \forall y \in E. \tag{27}$$

Since *E* is compact and convex and further *F* is continuous with respect to (u, π), according to Lemma 1, the above VI has a nonempty and compact solution set, denoted as $\Gamma = \{y^* = (u^*, \pi^*)\}$. We next show that the solution set of *NCPDUE*, denoted as Ξ, coincides with Γ.

First of all, for each solution $y^* = (u^*, \pi^*) \in \Gamma$, since *E* is a polyhedral set, there must exist multipliers γ and η such that

$$\begin{cases} 0 \leq u^* \perp F_u(u^*, \pi^*) + \gamma \geq 0 & (28a) \\ 0 \leq \pi^* \perp F_\pi(u^*, \pi^*) + \eta \geq 0 & (28b) \\ 0 \leq \gamma \perp \alpha_u 1 - u^* \geq 0 & (28c) \\ 0 \leq \eta \perp \alpha_\pi 1 - \pi^* \geq 0 & (28d) \end{cases}$$

In order to prove $y^* = (u^*, \pi^*)$ solves *NCPDUE*, we need to show that $\gamma = 0$ and $\eta = 0$. Suppose $\gamma_{bs}^k > 0$ for some *b*, *s*, *k*. We then must have ${u_{bs}^k}^* = \alpha_u > 0$ from (28c). This implies

$\rho_1 = \tau_b^k(u^*) + \sum_{l \le e_b^k(u^*)/\Delta < l+1} \lambda_b^{3,k,l}(u^*) \cdot \pi_{h_b s}^{l\ *} + [1-\lambda_b^{3,k,l}(u^*)] \cdot \pi_{h_b s}^{l+1\ *} - \pi_{l_b s}^{k\ *} + \gamma_{bs}^k = 0$ due to (28a). This also means that for the tail node of link b, denoted as node i, $\sum_{a \in A(i)} u_{as}^{k\ *} - d_{is}^k - \sum_{a \in B(i)} \sum_{k': e_a^{k'}(u^*) \le k\Delta < e_a^{k'+1}(u^*)} [\lambda_a^{1,k'}(u^*) \cdot \lambda_a^{2,k',k}(u^*) \cdot u_{as}^{k'\ *} + \lambda_a^{1,k'+1}(u^*) \cdot (1-\lambda_a^{2,k',k}(u^*)) \cdot u_{as}^{k'+1\ *}] + \eta_{is}^k \ge$ $\alpha_u - (d_{is}^k + \sum_{a \in B(i)} \sum_{k': e_a^{k'}(u^*) \le k\Delta < e_a^{k'+1}(u^*)} [\lambda_a^{1,k'}(u^*) \cdot \lambda_a^{2,k',k}(u^*) \cdot u_{as}^{k'\ *} + \lambda_a^{1,k'+1}(u^*) \cdot (1-\lambda_a^{2,k',k}(u^*)) \cdot u_{as}^{k'+1\ *}]) + \eta_{is}^k > 0$ based on (26b). Hence $\pi_{is}^{k^*} = \pi_{l_b s}^{k\ *} = 0$ according to (28b). Therefore, we have $0 < \tau_b^k(u^*) \le \rho_1 = \tau_b^k(u^*) + \sum_{l \le e_b^k(u^*)/\Delta < l+1} \lambda_b^{3,k,l}(u^*) \cdot \pi_{h_b s}^l + [1-\lambda_b^{3,k,l}(u^*)] \cdot \pi_{h_b s}^{l+1} + \gamma_{bs}^k = 0$ which is a contradiction.

Similarly if $\eta_{is}^k > 0$ for some i, s, k, we have $\pi_{is}^{k^*} = \alpha_\pi > 0$ based on (28d). According to (28a), we must have $\rho_2 = \tau_a^k(u^*) + \sum_{l \le e_a^{k'}(u^*)/\Delta < l+1} \lambda_a^{3,k,l}(u^*) \cdot \pi_{h_a s}^{l\ *} + [1-\lambda_a^{3,k,l}(u^*)] \cdot \pi_{h_a s}^{l+1\ *} - \pi_{l_a s}^{k\ *} + \gamma_{as}^k \ge 0$ for any link $a \in A(i)$. However since $\pi_{l_a s}^{k\ *} = \pi_{is}^{k^*} = \alpha_\pi$ and $\gamma_{as}^k = 0$, $\rho_2 = \tau_a^k(u^*) + \sum_{l \le e_a^k(u^*)/\Delta < l+1} \lambda_a^{3,k,l}(u^*) \cdot \pi_{h_a s}^{l\ *} + [1-\lambda_a^{3,k,l}(u^*)] \cdot \pi_{h_a s}^{l+1\ *} - \alpha_\pi < 0$ due the definition in (26c). This is a contradiction.

So far we have proved that $\Gamma \subseteq \Xi$. In order to prove $\Xi \subseteq \Gamma$, we notice that for any solution $y^* = (u^*, \pi^*) \in \Xi$, (28) has to be satisfied with the two set of multipliers $\gamma = 0$ and $\eta = 0$. Due to the equivalence between (28) and VI (27), y^* must also be a solution to VI (27). This implies $y^* \in \Gamma$ and further $\Xi \subseteq \Gamma$. Therefore, $\Xi = \Gamma$ and since Γ is nonempty and compact, so is the solution set of *NCPDUE*. □

4. Solution algorithm

The solution algorithm in this section is based on the fact that for a given flow $\bar{u}$ (denoted as base inflow), the link travel time vector τ will be fixed by a network loading procedure. Hence, all $\lambda^1, \lambda^2, \lambda^3$ and e will be fixed as well. As a result, the originally non-close-form NCP model (22) will become close-form as in (29):

$$\begin{cases} 0 \le u_{as}^k \perp \{\tau_a^k(u) + \sum_{l \le e_a^k(\bar{u})/\Delta < l+1} \lambda_a^{3,k,l}(\bar{u}) \cdot \pi_{h_a s}^l + [1-\lambda_a^{3,k,l}(\bar{u})] \cdot \pi_{h_a s}^{l+1} - \pi_{l_a s}^k\} \ge 0, \forall a \in A, s \in S, k = 1, \cdots K', & \text{(29a)} \\ 0 \le \pi_{is}^k \perp (\sum_{a \in A(i)} u_{as}^k - d_{is}^k - \sum_{a \in B(i)} \sum_{k': e_a^{k'}(\bar{u}) \le k\Delta < e_a^{k'+1}(\bar{u})} [\lambda_a^{1,k'}(\bar{u}) \cdot \lambda_a^{2,k',k}(\bar{u}) \cdot u_{as}^{k'} + \lambda_a^{1,k'+1}(\bar{u}) \cdot (1-\lambda_a^{2,k',k}(\bar{u})) \cdot u_{as}^{k'+1}] \ge 0, \forall i \in N, s \in S, i \ne s, k = 1, \cdots K'. & \text{(29b)} \end{cases}$$

Note the only difference between model (22) and (29) is that $\lambda^1, \lambda^2, \lambda^3$, e and τ in (29) are evaluated at the base inflow $\bar{u}$, while in (22) they are functions of u. In other words, (29) is a close-form NCP which is much easier to solve. We denote (29) as the "relaxed" NCP of (22).

The above observation outlines an iterative algorithm to solve model (22). It is heuristic in the sense that the convergence can not be established using regularity conditions. This is mainly due to the fact that (22) can not be expressed in a close form. The algorithm is listed as below.

Algorithm DUE

Step 1. Initialization. Assign an initial feasible base inflow $(\bar{u})^0$.
Step 2. Main Loop. Set *n=0*.
Step 2.1 Construct current relaxed NCP at $(\bar{u})^n$ by a network loading procedure.
Step 2.2 Solve the relaxed NCP and denote its solution $(\tilde{u})^n, (\tilde{\pi})^n$ as a "candidate" solution.
Step 2.3 Convergence Test. If certain convergence criterion is satisfied at the candidate solution, go to Step 3; else, go to Step 2.4.
Step 2.4 Update and Move. Set $(\bar{u})^{n+1} = (\bar{u})^n + \theta \cdot ((\tilde{u})^n - (\bar{u})^n)$, *n=n+1* and go to Step 2.1.
Step 3. Find an optimal solution $(\tilde{u})^n, (\tilde{\pi})^n$.

In the above algorithm, $0 \le \theta \le 1$ is a pre-defined step size. In addition, there are several options for the convergence test in Step 2.3. The most commonly used and also simplest way is to check whether the base inflows stabilize between two consecutive iterations, i.e.,

$$Gap\_u = |(\bar{u})^n - (\bar{u})^{n+1}| \le \varepsilon_1 . \quad (30)$$

A more rigorous way is to check whether the complementarity condition in equation (1) holds:

$$Gap\_DUE = u^T F_u(u, \pi) \le \varepsilon_2 . \quad (31)$$

In (30) and (31), ε_1 and ε_2 are chosen as small positive scalars.

The next two sections will further discuss the network loading procedure and the method for solving the relaxed NCP model (29).

4.1 Link-Based Network Loading Procedure

The network loading procedure in Algorithm DUE is link-based, i.e., for a given based inflow $\bar{u}$, it is to load/propagate $\bar{u}_{as}^k$ to link *a* for each link *a* at any time interval *k* towards a given destination *s*. This will generate other traffic measurements such as $\bar{v}, \bar{x}, \tau(\bar{u}), e(\bar{u})$. In turn, $\lambda^1(\bar{u}), \lambda^2(\bar{u}), \lambda^3(\bar{u})$ can also be determined. In Carey and Ge (*33*) and Nie and Zhang (*28*), this loading procedure is also called the "solution algorithm for link travel time model." In this paper, we intend to call it a "loading procedure" since it indeed generates not only travel times but also link flows and exit flows.

To be consistent with the discretization scheme in Section 3.1, we need to adopt the loading process in Astarita (*24*). However, since the original algorithm by Astarita (*24*) may have numerical problems (*28*), in this paper, we apply the improved algorithm by Nie and Zhang (*28*) which performs the loading based on cumulative departure curves (Algorithm D2). Furthermore, in order to construct the relaxed NCP model (29), one needs to track the relation between $\bar{u}_{as}^{k'}$ and $\bar{v}_{as}^k$ for any *a* and *s* and appropriate (k', k) pairs. Due to the fact that inflows to any destination *s* will experience the same travel time at each entrance time *k*, we can first use a three dimensional matrix $V(a,k',k)$ to represent this relation. In particular, $V(a,k',k)=\rho$ means that a proportion of ρ ($0 \le \rho \le 1$) of the inflows $\bar{u}_{as}^{k'}\Delta$ will exit the link *a* at time *k*, i.e., become part of $\bar{v}_{as}^{k}\Delta$. It turns out that we can introduce a "stack" for this tracking purpose. The revised loading procedure, denoted as Algorithm DL, is listed as follows. Note that since the loading procedure is the same for each link, we only show it for link *a*. Also, we omit the "bar" symbol on each variable to simplify the notation.

Algorithm DL

Step 0 Initialization. Set $l=\lfloor \alpha_a/\Delta \rfloor, x_{as}^1=0, v_{as}^k=0, \forall s\in S, k=1,\cdots,l; e_a^1=\alpha_a; R_{as}^u=0, \forall s \in S$. Create an empty stack *SR*. Set *k*=1.

Step 1 Move. Set *k*=*k*+1. Compute $x_{as}^k = x_{as}^{k-1}+(u_{as}^{k-1}-v_{as}^{k-1})\Delta, \forall s\in S$ and $x_a^k=\sum_{\forall s\in S} x_{as}^k, u_a^k=\sum_{\forall s\in S} u_{as}^k, v_a^k=\sum_{\forall s\in S} v_{as}^k$.

Then compute τ_a^k. Set $e_a^k=(k-1)\Delta+\tau_a^k$ and $n_l=\lfloor e_a^k/\Delta - l\rfloor$. If $n_l<1$, go to Step 1.1; otherwise, go to Step 1.2.

Step 1.1: Update $R_{as}^u=R_{as}^u+u_{as}^{k-1}, \forall s\in R$. Push pair $(k-1,1)$ into stack *SR*.

Step 1.2: Set *l*=*l*+1; $\rho_k=(l\Delta-e_a^{k-1})/(e_a^k-e_a^{k-1})$; $v_{as}^l=R_{as}^u+\rho_k u_{as}^{k-1}, \forall s\in S$. Push pair $(k-1,\rho_k)$ to stack *SR*. Then pop each entry (i,ρ) from stack *SR* and set $V(a,i,l)=\rho$.

For *j*=2 to n_l: set *l*=*l*+1, $\rho_k=\Delta/(e_a^k-e_a^{k-1})$, $v_{as}^l=\rho_k u_{as}^{k-1}, \forall s\in S$, and $V(a,k-1,l)=\rho_k$.

Set $\rho_k=(e_a^k-l\Delta)/(e_a^k-e_a^{k-1})$ and update $R_{as}^u=\rho_k u_{as}^{k-1}, \forall s\in S$. Push the pair $(k-1,\rho_k)$ to stack SR.

Step 2. If $k<T'$, go to Step 1; otherwise, stop.

In Algorithm DL, $\lfloor x \rfloor$ denotes the integral part of a real value *x*, α_a is the free flow travel time for link *a*, and $R_{as}^u\Delta$ is the undistributed inflows for destination *s*. The term "undistributed" here represents inflows that have not yet been used to determine exit flows (*28*). The stack *SR* is used to track the time interval and proportion of those inflows that are undistributed. Therefore, each entry in SR is a pair of (*k*, *ρ*) where *k* is the time interval and *ρ* is the proportion, both for inflows. Besides tracking the relation between *u* and *v*, Algorithm DL above also extends Algorithm D2 in Nie and Zhang (*28*) by performing the loading for each individual destination. It also worth noting that the matrix *V* represents λ^1 and λ^2 in (29) and λ^3 can be computed using equation (20). Consequently, NCP (29) can be constructed readily using *V* and λ^3.

4.2 Solving the Relaxed NCP

Since the relaxed model (29) is a well-defined NCP with continuous and close-form defining functions, it can be readily solved using existing solution techniques. Facchinei and Pang (*19*) provides a comprehensive review of solution methods for NCPs. In particular, the projection-based methods play a central role in solving NCPs because calculating the projection on the nonnegative orthant, the defining set of an NCP, is extremely easy and efficient compared with that on a general convex set. Based on this observation, Dirkse and Ferris (*34*) developed a path search algorithm which, under certain regularity conditions, is proved to be globally convergent with quadratic convergence rate (near the solution). The algorithm was later evolved to the

PATH solver which is now available in GAMS (General Algebraic Modeling System, see *35*). In this paper, we directly adopt the PATH solver which has been shown to be effective to solve NCP (29). For detailed descriptions of the solver, one can refer to Ferris and Munson (*36*). Using the PATH solver requires developing GAMS codes for the relaxed NCP (29) which is straightforward and hence the details are omitted here.

5. Numerical examples

In this section, a case study is provided to demonstrate the model and solution algorithm proposed in the paper. We start with the link travel time function that is actually used.

5.1 Link Travel Time Function

In this paper, we choose the following linear form for the link travel time function:

$$\tau_a^k = \alpha_a (1 + \beta_a^u u_a^k + \beta_a^x x_a^k), \tag{32}$$

where α_a is the free flow travel time for link *a*, β_a^u and β_a^x are constants that are shown in Table 1. Here we introduce the aggregated link inflow into the link travel time function by the constant β_a^u. The purpose is to make the solution algorithm for relaxed NCP (29) numerically stable. The reason is as follows. First, (29) can be rewritten in a matrix notation as

$$\begin{cases} 0 \le u_s \perp [\tau(u) + \Omega_s \pi_s] \ge 0 \\ 0 \le \pi_s \perp [\Lambda_s u_s - d_s] \ge 0 \end{cases}, \forall s \in S \tag{33}$$

with Ω_s and Λ_s are fixed matrices computed using the base inflow. Here $u_s = (u_{as}^k)_{\forall a,k}$, $\pi_s = (\pi_{is}^k)_{\forall i,k;i \ne s}$, and $d_s = (d_{is}^k)_{\forall i,k;i \ne s}$ denote destination-based variables.

Model (33) has a very special structure such that it can be easily decomposed to individual destinations. The interactions between variables of different destinations only exist in the link travel time vector $\tau(u)$. If we compute the Jacobian matrix of (33), denoted as *M*, we will have

$$M = \begin{array}{c} \begin{matrix} \pi_1 & \cdots & \pi_{|S|} & u_1 & \cdots & u_{|S|} \end{matrix} \\ \begin{bmatrix} 0 & \cdots & 0 & \Lambda_1 & 0 & 0 \\ \vdots & \ddots & \vdots & 0 & \ddots & 0 \\ 0 & \cdots & 0 & 0 & 0 & \Lambda_{|S|} \\ \Omega_1 & 0 & 0 & \partial\tau/\partial u_1 & \cdots & \partial\tau/\partial u_{|S|} \\ 0 & \ddots & 0 & \vdots & \ddots & \vdots \\ 0 & 0 & \Omega_{|S|} & \partial\tau/\partial u_1 & \cdots & \partial\tau/\partial u_{|S|} \end{bmatrix} \end{array} \begin{matrix} \pi_1 \\ \cdots \\ \pi_{|S|} \\ u_1 \\ \cdots \\ u_{|S|} \end{matrix} . \tag{34}$$

In (34), each variable on the right side of the matrix indicates the corresponding row is computed by taking the partial derivative of the function perpendicular to the particular variable over all variables; while each variable on the top indicates that the column is computed by taking partial derivative of each function to the particular variable. If β_a^u=0, the diagonal of $\partial\tau/\partial u_s$ will be all zero since x_a^k only includes inflows up to time (*k*-1) and not u_a^k. This implies that the diagonals of the *M* are all zero. Such a matrix can be proved to be not positive definite and thus may cause problems to solve the relaxed NCP. Adding a small positive β_a^u to the diagonal turns out to be helpful to stabilize the solution process. It worth noting that $\beta_a^u << \beta_a^x$, therefore, the possibility of FIFO violations due to β_a^u is expected to be small which will be verified in later numerical studies.

5.2 Case Study

The case study is a small example tested on the hypothetical network depicted in Fig. 2, denoted as the D3 network. In the DUE literature, this network was first used by Chen and Hsuen (*13*). In this paper, we use slightly different specifications for the D3 network, as shown in Table 1. The network has two origins: node 1 and 2, and one destination: node 3. Further, the length of each time interval is set as $\Delta = 0.25$ minutes (15 seconds).

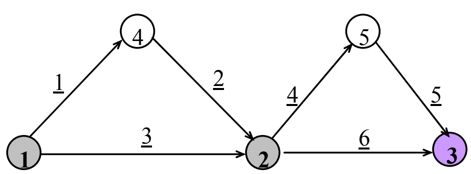

Fig. 2 Test network

Table 1 Configuration of the test network

Link	α_a (min)	β_a^u (min²/vehicle)	β_a^x (min/vehicle)
1	1.2	0.00125	0.01
2	1.2	0.00125	0.01
3	2.16	0.00125	0.0056
4	1.2	0.00125	0.01
5	1.2	0.00125	0.01
6	2.4	0.00125	0.005

To simulate the fluctuation of traffic volumes during peak hours, we employ a parabolic-shaped curve to represent the OD demands between each OD pair. In particular, the demand rate between any OD pair *rs* is assumed to be calculated through equation (35):

$$d^{rs}(k) = 40 + 120 * (1 - \left(\frac{k - K/2}{K/2}\right)^2), \forall 1 \le k \le K, \forall r \in R, s \in S, \quad (35)$$

where *K* denotes the total number of intervals during which OD trips will be generated.

For the case study, we set *K*=120 which is equivalent to 30 minutes. Algorithm DUE can solve successfully the proposed model. Fig. 3 first depicts the convergence performance of the algorithm for the first 25 iterations. In this figure, although the units for the two gaps are different as indicated, we plot them together in order to show their slight different performances. We can easily observe that both gaps decrease monotonically, whereas *Gap_u* decreases faster. When close to 10^{-4}, *Gap_DUE* is stabilized. Furthermore, after 25 iterations, the absolute difference of inflow rates between two consecutive iterations is close to 10^{-5} vehicles/minute. This should be accurate enough for most transportation related applications. It is also worth noting that due to the exact solve of each relaxed NCP by the PATH solver (usually to 10^{-6}), the proposed Algorithm DUE requires much less number of major iterations than previous DUE solution methods based on FW.

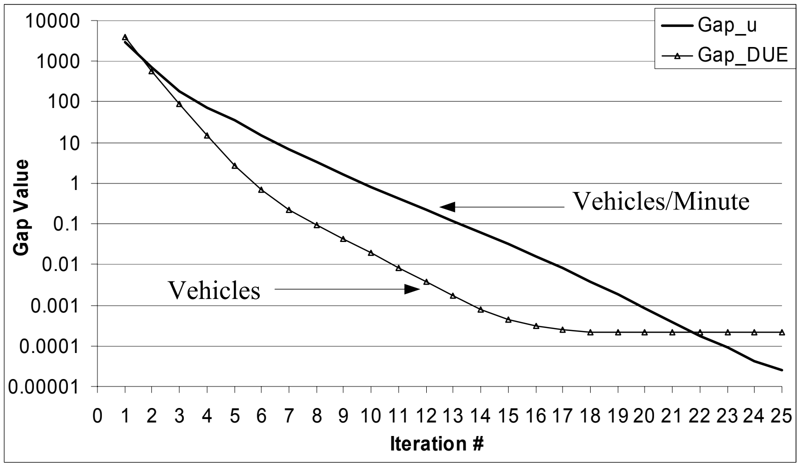

Fig. 3 Convergence of the algorithm

Fig. 4(a) below shows the aggregated inflow rate on each link. We can see that the inflow rates to link 1 and 3 change more abruptly than those to link 4 and 6. This may be due to the fact that the choice of links at node 1 will be impacted by the inflows at node 2, and not vice versa. Notice that the inflows to link 2 and 5 are exactly the exit flows from link 1 and 4, respectively. We can therefore observe that the shapes of exit flows are similar to their corresponding inflows, but are smoother. We further plots in Fig. 4(b) the travel times from node 1 and 2 respectively to destination 3 via different links. This figure shows that at the beginning (k=1 to 15), the travel time from node 1 to 3 via link 1 is higher than that via link 3. Consequently, all vehicles choose link 3 during this period as depicted in Figure 4(b). This is also true for the period of k>=115. While from node 2 to 3, choosing either link 4 or link 6 will have almost equal travel times, hence both links will be selected. The time-dependent link flows are also presented in Fig. 4(c).

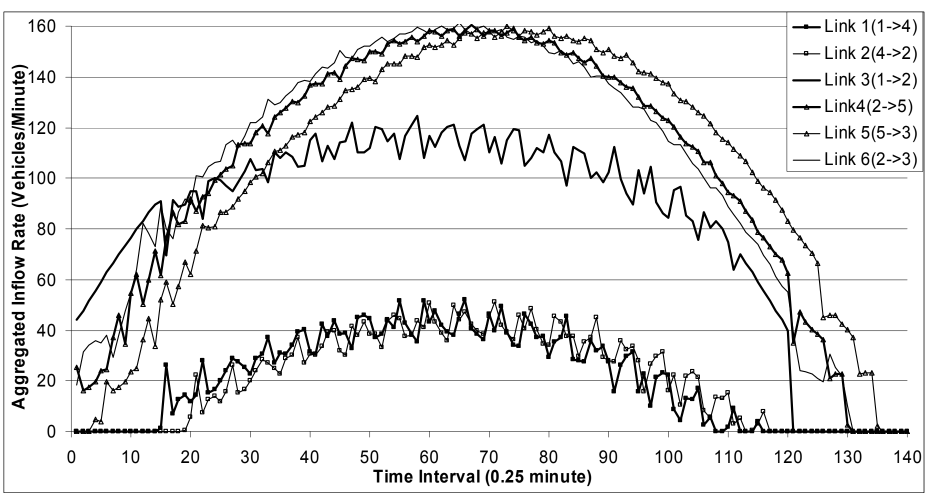

Fig. 4(a) Inflow Rates

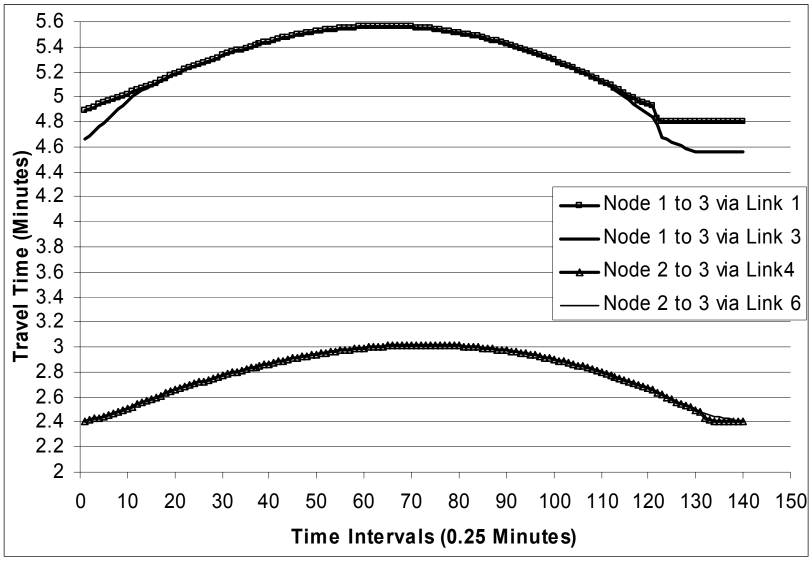

Fig. 4(b) Travel Times

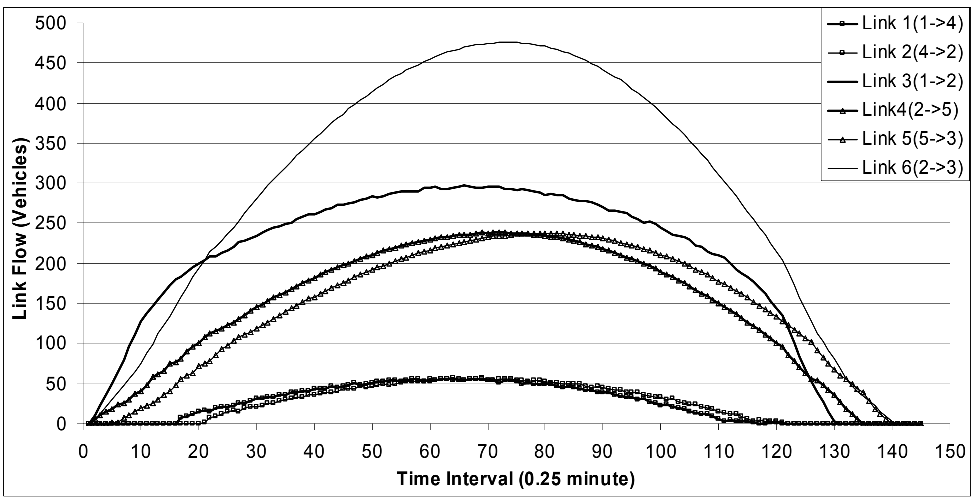

Fig. 4(c) Link Flows

We illustrate the discretized version of the derivative of link travel time (over time), i.e. $(\tau_a^{k+1} - \tau_a^k)/\Delta$, in Fig. 5. Clearly link travel times change rather abruptly, but the derivatives are within the range of (-0.1, 0.1). Therefore, FIFO is not violated for our case study since $(\tau_a^{k+1} - \tau_a^k)/\Delta > -1$ is held for all links at all time intervals.

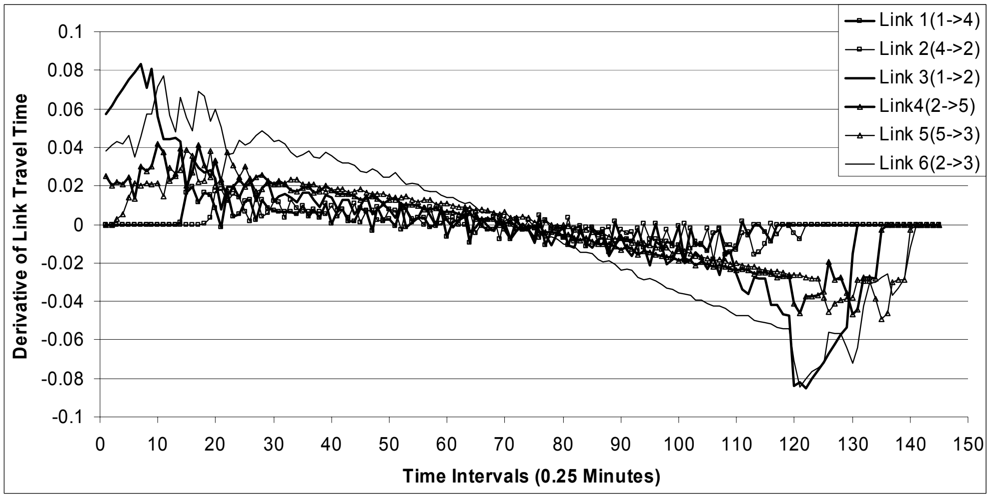

Fig. 5 Link Travel Time Changes

6. Conclusions and future research

We presented a link-node based NCP model for basic DUE problem and explicitly captured the exact flow propagations. The solution existence and compactness condition was established under mild assumptions. We also developed an iterative solution algorithm for the proposed model by solving a relaxed NCP in each iteration. The relaxed NCP can be solved very efficiently using existing solution technique and thus the entire algorithm only required fairly small number of iterations. The case study demonstrated that the proposed model and solution approach are effective for solving DUE problems.

For future studies, the proposed NCP model, especially its solution approach, merits further investigations. Especially, certain solution convergence condition needs to be established. Secondly, the travel time function used in this paper may be suitable only for mild congestion scenarios. In this regard, more sophisticated functional forms need to be developed to capture heavy traffic congestion. Lastly, the model and solution approach proposed in this paper need to be further tested on large scale DUE problems. Due to the special structure of the proposed NCP model, certain decomposition scheme based on individual destinations may be applied. The authors have developed such schemes for solving asymmetric and static user equilibrium problems (*37*), as well as some preliminary extensions to DUE (*38*). More rigorous extensions to DUE are currently under investigation and results will be reported in subsequent papers.

References

1. Astarita, V., 1996. A continuous time link model for dynamic network loading based on travel time function. In: Lesort, J.-B. (Ed.), *Transportation and Traffic Theory*. Pergamon, Oxford, 79-102.
2. Ban, X., 2005. *Quasi-variational Inequality Formulations and Solution Approaches for Dynamic User Equilibria*. Ph.D Thesis, University of Wisconsin-Madison.

3. Ban, **X**., Liu, H., and Ferris, M.C., 2006. A link-node based complementarity model and its solution algorithm for asymmetric user equilibria. In *Proceedings of the 85th Transportation Research Board Annual Meeting (CD-ROM)*.
4. Billups, S.C., Dirkse, S. P. and Ferris, M. C., 1997. A comparison of large scale mixed complementarity problem solvers. *Computational Optimization and Applications* 7, 3-25
5. Bliemer, M.C.J., 2001. *Analytical dynamic traffic assignment with interacting user-classes: Theoretical advances and applications using a variational inequality approach*. Ph.D. Thesis, Delft University of Technology, The Netherlands.
6. Bliemer , M.C.J. and Bovy, P.H.L., 2003. Quasi-variational inequality formulation of the multiclass dynamic traffic assignment problem. *Transportation Research* 37B, 501-519.
7. Brooke, A., Kendrick, D. and Meeraus, A., et al., 1998. *GAMS, a user's guide*. GAMS Development Corporation.
8. Cao, M., and Ferris, M. C., 1996. A pivotal method for affine variational inequalities. M*athematics of Operations Research* 21, 44-64.
9. Carey, M., 1987. Optimal Time-Varying Flows on Congested Networks. *Operations Research* 35, 58-69.
10. Carey, M., Ge, Y.E. and McCartney, M., 2003. A whole-travel time model with desirable properties. *Transportation Science* 37(1), 83-96.
11. Carey, M. and Ge, Y.E. (2004). Efficient discretisation for link travel time models. *Networks and Spatial Economics* 4, 269-290.
12. Chen, H.K. 1999. *Dynamic Travel Choice Models*. Springer-Verlag, New York.
13. Chen, H.K. and Hsuen, C.F., 1998. A model and an algorithm for the dynamic user-optimal route choice problem. *Transportation Research* 32B(3), 219-234.
14. Dafermos, S.C., 1980. Traffic equilibrium and variational inequality. *Transportation Science*, 14, 42-54.
15. Daganzo, C.F., 1994. The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory. *Transportation Research* 28B(4), 269-287.
16. Daganzo, C.F., 1995a. The cell transmission model part II: network traffic. *Transportation Research* 29B(2), 79-93.
17. Daganzo, C.F., 1995b. Properties of link travel time functions under dynamic loads. *Transportation Research*, 29B, 95-98.
18. Dirkse, S.P. and Ferris, M.C., 1995. The PATH solver: A non-monotone stabilization scheme for mixed complementarity problems. *Optimization Methods and Software* 5, 123-156.
19. Facchinei, F. and Pang J.S., 2003a. *Finite-Dimensional Variational Inequalities and Complementarity Problems (Vol. I and II)*. Springer-Verlag, New York
20. Ferris, M.C. and Munson, T.S., 1998. *PATH 4.6 User Manual*. University of Wisconsin-Madison.
21. Ferris, M. C. Fourer, R., and Gay, D. M., 1999. Expressing complementarity problems and communicating them to solvers. *SIAM Journal on Optimization* 9, 991-1009.
22. Friesz, T.L., Bernstein, D. and Smith, T.E., et al., 1993. A Variational Inequality Formulation of the Dynamic Network User Equilibrium Problem. *Operations Research* 41(1), 179-191.
23. Friesz, T.L., Bernstein, D., Suo, Z. and Tobin, R.L. (2001) Dynamic network user equilibrium with time-dependent time lags. *Networks and Spatial Economics* 1, 319-347.

24. Friesz, T.L. and Mookherjee, R., 2005 Solving the dynamic network user equilibrium problem with state-dependent time shifts. *Transportation Research, Part B*. In Press.
25. Han, S. and Heydecker, B.G., 2006. Consistent objectives and solution of dynamic user equilibrium models. *Transportation Research* 40B, 16-34.
26. Heydecker, B.G. and Verlander, N., 1999. Calculation of dynamic traffic equilibrium assignments. Proceedings of the European Transport Conferences, Seminar F, 79-91.
27. Lo, H.K., Ran, B. and Hongola, B., 1996. Multiclass Dynamic Traffic Assignment Model: Formulation and Computational Experiences. *Transportation Research Record 1537*, 74-82.
28. Lo, H. and Szeto, W.Y., 2002. A cell-based variational inequality formulation of the dynamic user optimal assignment problem. *Transportation Research* 36B, 421-443.
29. Nie, X.J. and Zhang, H.M., 2005. Delay-function-based link models: their properties and computational issues. *Transportation Research*, Part B, in press.
30. Peeta, S. and Ziliaskopoulos, A.K., 2001. Foundations of Dynamic Traffic Assignment: the Past, the Present and the Future. *Networks and Spatial Economics* 1, 233–265.
31. Ran, B. and Boyce, D.E., 1994. *Dynamic Urban Transportation Network Models: Theory and Implications for Intelligent Vehicle Highway Systems, Lecture Notes in Economics and Mathematical Systems* 417. Springer-Verlag, New York.
32. Ran, B. and Boyce, D.E, 1996. *Modeling Dynamic Transportation Networks: An Intelligent Transportation Systems Oriented Approach,* 2nd revised edn. Springer-Verlag, New York.
33. Ran, B., Lo, H. and Boyce, D.E., 1996. A Formulation and Solution Algorithm for A Multi- class Dynamic Traffic Assignment Problem *Transportation and Traffic Theory*. Edited by Lesort, J.B., Elsevier Science, UK, 195-216.
34. Smith, M.J., 1979. The Existence, Uniqueness and Stability of Traffic Equilibria. *Transportation Research* 13B, 295-304.
35. Ulbrich, M., 2002. *Nonsmooth Newton-like Methods for Variational Inequalities and Constrained Optimization Problems in Function Spaces*. Ph.D thesis, Technische Universitat Munchen, Germany.
36. Wardrop, J.G., 1952. Some theoretical aspects of road traffic research. In *Proceedings of the Institution of Civil Engineers*, Part II, 1, 325-378.
37. Wie, B.W., Tobin, R.L. and Carey, M., 2002. The existence, uniqueness and computation of an arc-based dynamic network user equilibrium formulation. *Transportation Research* 36B, 897-918.
38. Xu, Y.W., Wu, J.H., Florian, M., Marcotte, P., and Zhu, D.L., 1999. Advances in the continuous dynamic network loading problem. *Transportation Science* 33, 341-353.