NETWORKS AND HETEROGENEOUS MEDIA
©American Institute of Mathematical Sciences
Volume **12**, Number **1**, March **2017**

doi:10.3934/nhm.2017005
pp. **113–145**

A MATHEMATICAL FRAMEWORK FOR DELAY ANALYSIS IN SINGLE SOURCE NETWORKS

Samitha Samaranayake*

School of Civil and Environmental Engineering
Cornell University
Ithaca, NY 14853, USA

Axel Parmentier

CERMICS, École des Ponts Paristech
Université Paris Est
6 et 8 Avenue Blaise Pascal, 77420 Champs sur Marne, France

Ethan Xuan

Facebook Inc.
New York, NY 10003, USA

Alexandre Bayen

Department of Electrical Engineering & Computer Science and
Department of Civil and Environmental Engineering
University of California Berkeley
Berkeley, CA 94702, USA

(Communicated by Benedetto Piccoli)

Abstract. This article presents a mathematical framework for modeling heterogeneous flow networks with a single source and multiple sinks with no merging. The traffic is differentiated by the destination (i.e. Lagrangian flow) and different flow groups are assumed to satisfy the first-in-first-out (FIFO) condition at each junction. The queuing in the network is assumed to be contained at each junction node and spill-back to the previous junction is ignored. We show that this model leads to a well-posed problem for computing the dynamics of the system and prove that the solution is unique through a mathematical derivation of the model properties. The framework is then used to analytically prescribe the delays at each junction of the network and across any sub-path, which is the main contribution of the article. This is a critical requirement when solving control and optimization problems over the network, such as system optimal network routing and solving for equilibrium behavior. In fact, the framework provides analytical expressions for the delay at any node or sub-path as a function of the inflow at any upstream node. Furthermore, the model can be solved numerically using a very simple and efficient feed forward algorithm. We demonstrate the versatility of the framework by applying it to two example networks, a single path of multiple bottlenecks and a diverge junction with complex junction dynamics.

2010 *Mathematics Subject Classification.* Primary: 34B45.
Key words and phrases. Delay analysis, traffic flow, flow networks.

* Corresponding author.

1. **Introduction.**

Modeling and analysing the dynamics of network flows is an important problem that has applications in many different areas such as transportation planning [7, 25, 42], air traffic control [32, 48], communication networks [1, 6, 12, 22], processor scheduling [49] and supply chain optimization [34, 17]. Flow models are crucial for understanding the response of networked systems under different boundary conditions, estimating the state of the system, measuring system performance under different tunable parameters and devising the appropriate control strategies for efficient operation of the system. For example, in transportation networks, flow models are used for traffic estimation [53], dynamic traffic assignment or demand response assessment [30], traffic signal control [28], ramp-metering control [43] and rerouting [47]. This article focuses on modeling heterogeneous (multi-path) traffic flow through a network with a single source and multiple sinks with the objective of analytically expressing the delays at each node of the network as a function of the boundary flows at the source. This is a critical requirement when solving certain control and optimization problems over the network, where flow entering the network is one of the direct or indirect control parameters [46]. We present this model in the context of a transportation network modeling, but the results can be applied to any single origin network with no merge junctions that satisfies the following properties; 1) link flows are capacity restricted, 2) the flow through each junction satisfies the first-in-first-out (FIFO) condition, and 3) there is no holding of flow, i.e. the flow through a junction is maximized subject to the FIFO condition.

There are many types of traffic dynamics models, including microsimulation [18], cellular automaton [33], first-order or higher-order macroscopic models [19, 26, 44, 51] and point queue models [52], listed roughly in decreasing model complexity. Microsimulation models simulate the movement of individual vehicles by applying car following models and lane changing models. This approach enables the modeling of very specific and detailed characteristics of driver behavior and networks dynamics [18] such as route choice modeling and the implications of the precise network geometry (e.g. road curvature). Therefore, microscopic models are very popular in transportation planning at an urban scale, since planners have the ability to observe the behavior on the entire network due to modifications such as road closures and changes in traffic signal plans. However, these models can not be expressed in closed form, are very difficult to calibrate [20] and have an extremely high computational overhead associated with them.

Another related approach is to use cellular automaton models to simulate traffic [33]. Typically, cellular automaton models divide the roadway into cells, each of which can be empty or occupied by only one vehicle. This approach as also been extended to model heterogeneous flow [21, 23], where the behavior around freeway off-ramps are considered. Cellular automation models are similar to microscopic models in the sense that they both scale very poorly with the problem size and are generally computationally intractable since they track individual vehicles.

The third and perhaps most common approach is the use of macroscopic traffic flow models. Among these models, the most well known is the continuum or fluid-dynamic model based on the Lighthill-Williams-Richards (LWR) partial differential equation [26, 27, 44]. This model uses first-order partial differentiation equations to describe the relationship between aggregate traffic flow variables (flux and density), under the principle of mass conservation. Due to its simplicity, the LWR model does not capture individual driver behavior and cannot be used to model heterogeneous flows where different groups of drivers exhibit different behaviors. However, there have been extensions to the LWR model that can account for such heterogeneous flow [8, 9].

The most popular solution method to the LWR PDE is the Godunov scheme, also known as the cell transmission model (CTM) [7, 25]. The CTM discretizes space into cells and keeps track of the vehicle density in each cell instead of individual vehicles. Another approach for solving the LWR PDE is using the Moskowitz function. The Moskowitz function numbers vehicles as they pass a reference location and it describes the evolution of this number over time and space. The Moskowitz function can be interpreted as the cumulative vehicle count at a reference location, also known as cumulative curves, which is similar to our approach. Cumulative curves have been proposed together with triangular shaped fundamental diagram (relationship between flow and density) to simplify the traffic analysis based on the LWR model [35, 36, 37], because the adoption of cumulative curves automatically guarantees conservation. However, while cumulative curves do provide a nice graphical interpretation of the traffic flow, they do not present the ability to describe the network behavior as a function of variable boundary flows.

The final approach, which is also the approach used in this work, is to model the traffic via a network of point queues that are typically represented using ordinary differential equations (ODE's). One of the first point queue models was presented in the seminal work of Vickrey [52] to study a roadway with a single bottleneck. This approach has since been extended to handle networks with multiple commodities [11]. One of the main limitations of this ODE-based approach is that it does not capture the spatial propagation of traffic along a roadway (unlike the PDE-based macroscopic models) and assumes that all delayed vehicles wait at the point of the bottleneck, as if vehicles were packets in a data network. As a result, point queue models are generally unable to capture queue spillback. However, some recent work [41, 40] shows how point queue models can be adapted to capture queue spillback using two queues for each bottleneck.

There is a vast literature in the transportation community on network flow models based on both ordinary differential equation (ODE) based point queues and the LWR PDE based spatio-temporal propagation. These models are also commonly referred to as network loading models [2] in dynamic traffic assignment (DTA) [31]. While the existing literature provides a rich portfolio of modeling techniques, of which some recent works we summarize below, this work presents a new technique that adds to the existing literature by enabling the analytical expression of delays within the network as a function of the inflows at the origin and the bottlenecks that are active. This however comes at the cost of some restrictions in terms of the types of networks that can be analyzed, as we will explain in detail below.

There have been a number of recent works on extending point queue models for network loading in DTA. In [3, 4], the authors take a similar approach to ours in terms of extending the Vickrey point queue model (with no spillback). A time discretized iterative numerical scheme is provided to obtain a dynamic user equilibrium solution and the existence of the solution is proven under the assumptions of the model. The convergence rate of the iterative scheme is not explicitly addressed. Similarly, [11] considers a point-queue model with no spillback, but is able to model congestion delay during the free-flow phase of the fundamental diagram for hyperbolic fundamental diagrams. In contrast, [29] uses a point queue network with multiple queues to capture spillback, based on the double queue technique in [41]. The existence and uniqueness of the solution to the network loading problem is not addressed, but an existence result for the dynamic system optimal solution is provided. While the work in [11] allows for merge junctions and the model in [29] captures some queue spillback, none of the works above can analytically express the delays within the network as a function of the input flows[1].

A series of articles by Han et al. [14, 15, 16] reformulates the Vickrey model using a PDE-based approach and extends the results to general networks. This reformulation enables more rigorous mathematical analysis of the network loading model, better numerical methods for solving the network loading problem and a proof of existence for a route and departure time user equilibrium. In a follow up work, [5] shows the existence and well-posedness of this PDE-based queuing model. Furthermore, [13] presents a very important contribution to the literature by providing a continuous time network loading model with both existence and well-posedness guarantees. This model can be considered a continuous time counterpart of the Link Transmission Model [39, 54]. The results on existence and well-posedness are however limited to certain junction models. In particular, the results require fixed turning ratios at junctions, which does hold in dynamic traffic assignment.

In addition, there is a rich literature on ODE-based network flow propagation for various packet networks [1, 6, 12, 22], but a large majority of these dynamics models violate the FIFO and no holding requirements that are essential in phyisical flow networks. The models proposed for transportation network flows, as described above, can in fact satisfy these physical requirements, but cannot analytically prescribe the internal delays of the network as a function of the boundary flows. Therefore, a new framework is required for the problem that we consider.

Our approach can be summarized as follows. We assume that the traffic flow is differentiated by the destination of the flow (i.e. Lagrangian flow) and that the different flow groups satisfy the FIFO condition at each junction. The network is limited to a contain a single origin and no merge junctions. The input flows are assumed to be piecewise continuous in time and the link capacities are assumed to be piecewise constant in time. The queuing in the network is assumed to be contained at each junction node and spill-back to the previous junction is ignored[2]. We show that this model leads to a well-posed ordinary differential equation for computing the dynamics of the network as a function of the boundary flows and prove that the solution is unique through a rigorous mathematical derivation of the model properties. The main benefit of this framework is the ability to analytically prescribe the delays at any junction in the network and across any sub-path as a function of the the boundary flows, which can be an important requirement when solving certain control and optimization problems, such as demand allocation problems, where the flow entering the network is one of the direct or indirect control parameters. This is achieved via the creation of a time mapping operator that maps the traffic flow at a given node at a given time to the corresponding flow at the origin of the network

[1] We wish to incorporate the approach in [41] for capturing queue spillback into the proposed model as future work.

[2] This is a restrictive assumption in the case where the queue length at a given node exceeds the length of the incoming link because the queue would have interrupted the flow at the previous junction. The model allows for these cases of queue spill-back to be identified by comparing the queue length to the queue capacity, and for control actions to be taken to mitigate the spill-back. However, in general, it may not be possible to prevent spill-back under certain input flow profiles.

when that flow entered the network. We also show that this model can be solved numerically using a simple and efficient forward simulation approach. Finally, we demonstrate the application of the model by applying it to two example networks, a single path of multiple bottlenecks and a diverge junction with complex junction dynamics including time varying capacities.

The article is organized as follows. Section 2 introduces the network properties and junction dynamics. Then section 3 formalizes the time mapping operator, shows the well-posedness of the problem and proves the uniqueness of the solution to this model. Section 4 demonstrates the practical application of the mathematical framework by showing that the off-ramp model posed by Newell [38] can be modeled using this framework. Section 5 concludes the article.

2. **A point queue model for network flow.**

The traffic network with a single source is modeled as an arborescence[3]. The congestion at each bottleneck is modeled as a vertical queue that is located at the start of the bottleneck. Thus, the physical propagation of the queue forming at the bottleneck is not modeled. This modeling choice is only restrictive when the queue propagates upstream to the preceding junction, as the change in dynamics at the junction due to the queue is not taken into account, but the model is equivalent to a horizontal queuing model otherwise.

2.1. **Network definitions.**

A node v denotes a junction in the network and V is the set of all nodes. A link $l = (v_l^{\text{in}}, v_l^{\text{out}})$ is a couple consisting of an origin node v_l^{in} and a destination node v_l^{out}, and L is the set of all links.

The congestion-free travel time on link l is denoted by T_l, an agent that enters link l at time t will exit link l at time $t+T_l$. The congestion-free travel time between nodes v_1 and v_2 is denoted by $T_{(v_1,v_2)}$, an agent that enters node v_1 at time t will reach node v_2 at time $t + T_{(v_1,v_2)}$

The set of incoming links to node v is denoted by L_v^{in}, the set of outgoing links from node v is denoted by L_v^{out} and the set of all links l connected to node v is denoted by L_v.

$$L_v^{\text{in}} = \{l \in L | v_l^{\text{out}} = v\},\ L_v^{\text{out}} = \{l \in L | v_l^{\text{in}} = v\} \tag{1}$$

A node v is a source if it admits no incoming link ($L_v^{\text{in}} = \emptyset$). A node v is a sink if it admits no exiting link ($L_v^{\text{out}} = \emptyset$). The set of sinks is denoted by S.

The set of nodes V and the set of links L compose a network. Due to the network being an arborescence, it contains a unique source indexed by v_0. For all nodes $v \in V \backslash \{v_0\}$, L_v^{in} is a singleton. The element of this singleton is called the *parent* node and is denoted by π_v: $L_v^{\text{in}} = \{(\pi_v, v)\}$.

We define a path $p_{(v_{\text{orig}},v_{\text{dest}})}$ as a finite sequence of distinct nodes from an origin node v_{orig} to a destination node v_{dest} such that there is a link connecting each pair of subsequent nodes.

$$p_{(v_{\text{orig}},v_{\text{dest}})} = (v_{\text{orig}}, \cdots, v_{\text{dest}}) \text{ s.t. } (\pi_{v_i}, v_i) \in L \quad \forall i \in p \backslash v_{\text{orig}} \tag{2}$$

There is a unique path from any source to any destination since the network is tree structured. For each sink s, let p_s be the path starting at the origin v_{orig} and ending at node $v_s = s$, and V_{p_s} be the sequence of nodes on path p_s. The set of

[3] An arborescence is a directed rooted tree where all edges point away from the root.

paths P_v is the set of all paths p for which $v \in p$. The set of paths P_l is the set of all paths p for which $l \in p$.

$$P_v = \{p | v \in V_p\} \ ; \ P_l = \{p | v_l^{\text{in}} \in V_p \text{ and } v_l^{\text{out}} \in V_p\} \tag{3}$$

Remark 1. The path sets P_l where l is a link in L_v^{out} form a partition of P_v

$$P_v = \cup_{l \in L_v^{\text{out}}} P_l \tag{4}$$

2.2. **Modeling the flow of agents.** The traffic flow at a node is measured by counting the number of agents that pass through the node between an arbitrary initial time t_{initial} and any given time t.

For a node $v \in V \backslash v_0$ (that is not the source) and path $p \in P_v$, the arrival curve $A_v^p(t)$ gives the total number of agents on path p that arrive at node v during the time interval $(t_{\text{initial}}, t]$. Similarly, for a node $v \in V \backslash S$ (that is not a sink) and $p \in P_v$, the departure curve $D_v^p(t)$ gives the total number of agents on path p that leave node v during the time interval $(t_{\text{initial}}, t]$.

Remark 2. The arrival curve $A_v^p(t)$ (resp. departure curve $D_v^p(t)$) also gives the agent number of the last agent on path p to arrive at (resp. leave) node v by t. Arrival and departure curves are non-decreasing: if $t_1 < t_2$, $A(t_2) - A(t_1)$ (resp. $D_p(t_2) - D_p(t_1)$) is the total number of agents who arrive at (resp. pass) node v in the interval $(t_1, t_2]$, and is therefore non-negative.

Definition 2.1. *Acceptable cumulative arrival and departure curves* $\mathbf{A}(t_{initial}, t_{final}]$, $\mathbf{D}(t_{initial}, t_{final}]$

Given times t_{initial} and t_{final}, a function on $(t_{\text{initial}}, t_{\text{final}}]$ is an acceptable cumulative curve on $(t_{\text{initial}}, t_{\text{final}}]$ if it is continuous, piecewise C^1, and strictly increasing functions on $(t_{\text{initial}}, t_{\text{final}}]$.

The assumption that the cumulative curves are strictly increasing is made for mathematical convenience, but can be relaxed[4]. Cumulative curves are required to be piecewise C^1 in order to be able to define flows.

The outgoing flow λ_p^v at a node v is the piecewise continuous derivative of the departure curve D_p^v

$$\lambda_p^v = \frac{dD_p^v}{dt} \tag{5}$$

Remark 3. *Zero congestion-free travel time*

Let π_v, v be two consecutive nodes on path p. Agents on path p leaving node v at time t arrive at node v at $t + T_{(\pi_v, v)}$. For all links (π_v, v) and paths $p \in P_v$, without loss of generality we set the congestion-free travel time $T_{(\pi_v, v)}$ to be zero: $T_{(\pi_v, v)} = 0$. This implies that:

$$D_p^{\pi_v} = A_p^v \quad \forall l = (\pi_v, v) \in L, p \in P_v \tag{6}$$

This modeling choice is made purely for mathematical convenience, since the goal of this framework is to analyze delays in the network. The total travel time for each agent can be easily reconstructed a posteriori by adding the actual congestion-free travel time for each link of the path traveled by the agent.

[4] We could relax the assumption that the cumulative curves are strictly increasing and allow for non-decreasing curves. However, this results in the time mapping function $T^{(' \pi_v)} v$ introduced in section 3.2 being a correspondence instead of a function and makes the analysis significantly more complicated. Therefore, for mathematical convenience, we make the assumption that the cumulative curves are strictly increasing.

Thus, for all links $(\pi_v, v) \in L$ and paths $p \in P$ we have:

$$\frac{dA_p^v}{dt} = \frac{dD_p^{\pi_v}}{dt} = \lambda_p^{\pi_v} \tag{7}$$

$$\frac{dD_p^v}{dt} = \lambda_p^v. \tag{8}$$

2.3. **Queuing and diverge model.** This section defines the model dynamics for queuing and the flow propagation through a junction, which will then lead to a definition of the feasible departure curves that the model admits.

The capacity $\mu_l(t)$ of a link l is the maximum flow that can enter the link from its input node v_l^{in} at time t. Road capacity may vary with time due to weather conditions, accidents, or other factors. Thus, capacity is a time varying quantity.

Requirement 1. *Capacity constrained flows*

The inflow entering a link is always no greater than the links capacity.

$$\sum_{p \in P_l} \lambda_p^{v_l^{in}}(t) \leq \mu_l(t) \qquad \forall t, l \in L \tag{9}$$

If the flows arriving at a node v are larger than available outflow capacity, a queue will form at node v.

Definition 2.2. *Queue length $n_{v,p}(t)$*

We define the path queue length $n_{v,p}(t)$ at node v as the number of agents on path p that arrive at node v by time t and are yet to depart node v

$$n_{v,p}(t) = A_p^v(t) - D_p^v(t) \tag{10}$$

The total queue length $n_v(t)$ at node v is the sum of the path queue lengths.

$$n_v(t) = \sum_{p \in P_v} n_{v,p}(t) \tag{11}$$

Remark 4. Let $[D^v]^{-1}$ be the inverse of the departure curve D^v. Since D^v is strictly increasing, $t_k = [D^v]^{-1}(k)$ gives the time at which agent number k leaves node v.

Definition 2.3. *Delay in queue v*

We define $\delta_{v,p}(t)$ as the delay encountered in queue v by the agent that entered the queue at time t.

$$\begin{aligned} \delta_{v,p}(t) \qquad &= [D_p^v]^{-1}\left(A_p^v(t)\right) - t \\ &= [D_p^v]^{-1}\left(D_p^{\pi_v}(t)\right) - t \end{aligned} \tag{12}$$

As D_p^v is continuous, piecewise C^1, and strictly increasing, its inverse is continuous, piecewise C^1 and strictly increasing. Since the composition of functions preserves continuity (see [45] for the C^0 case and [10, 24] for the C^k case), the function $[D_p^v]^{-1} \circ D_p^{\pi_v}$ is continuous and piecewise C^1, and therefore, the delay $\delta_{v,g}$ is also continuous and piecewise C^1.

Remark 5.

If $n_{v,p}(t) = 0 \; \forall p \in P_v$, then $D_p^v(t) = A_p^v(t) \; \forall p \in P_v \implies \delta_{v,p}(t) = 0 \; \forall p \in P_v$. If $\exists p' \in P_v : n_{v,p'}(t) > 0$, then $D_{p'}^v(t) < A_{p'}^v(t) \implies [D_p^v]^{-1}\left(A_p^v(t)\right) > t$ and $\delta_{v,p}(t) > 0 \; \forall p \in P_v$.

Therefore,

$$\forall t, \delta_{v,p}(t) > 0 \Leftrightarrow \exists p' \in P_v : n_{v,p'}(t) > 0 \tag{13}$$

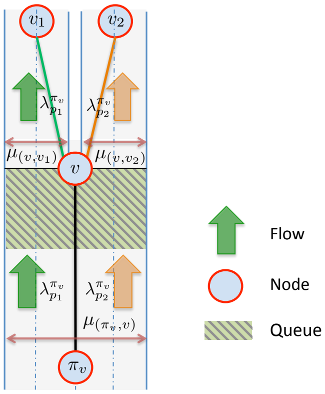

FIGURE 1. Diverge model

We now describe the diverge model with a graphical illstration of a one to two diverge junction given in Figure 1.

Requirement 2. *First-in-first-out (FIFO) property*

The model satisfies the FIFO property. The delay encountered in queue v at time t is identical for all paths p in P_v.

$$\delta_v(t) = \delta_{v,p}(t) = [D_p^v]^{-1}\left(D_p^{\pi_v}(t)\right) - t \qquad \forall t, \forall p \in P_v \tag{14}$$

FIFO property implies that agents exit the queue in the same order that they enter the queue regardless of which path they belong to.

$$t_1 < t_2 \Leftrightarrow [D_{p_1}^v]^{-1}\left(D_{p_1}^{\pi_v}(t_1)\right) < [D_{p_2}^v]^{-1}\left(D_{p_2}^{\pi_v}(t_2)\right) \tag{15}$$

Interpreting A_p^v (resp D_p^v) as the identifier of the agent which arrives in (resp. leaves) queue v at time t, we can see that the queues respect the FIFO rule for each path p. Let x_1 and x_2 be two agents: agent x_1 enters queue v at time t_1^{in} such that $A_p^v\left(t_1^{in}\right) = x_1$ and leaves queue v at time t_1^{out} such that $D_p^v\left(t_1^{out}\right) = x_1$, agent x_2 entered in queue v at time t_2^{in} such that $A_p^v\left(t_2^{in}\right) = x_2$ and leaves queue v at time t_2^{out} such that $D_p^v\left(t_2^{out}\right) = x_2$. As A_p^v and D_p^v are both strictly increasing functions, $t_1^{in} \leq t_2^{in} \Rightarrow x_1 \leq x_2 \Rightarrow t_1^{out} \leq t_2^{out}$, which means that if x_1 is enters queue v before x_2, it will leave v before x_2.

Proposition 1. *FIFO implies conservation of the ratio of flows*

If p_1 and p_2 are two paths in P_v such that $\lambda_{p_1}^{\pi_v}, \lambda_{p_2}^{\pi_v} > 0$, then the ratio of their flows is conserved when exiting node v

$$\frac{\lambda_{p_1}^v\left(t + \delta_v(t)\right)}{\lambda_{p_2}^v\left(t + \delta_v(t)\right)} = \frac{\lambda_{p_1}^{\pi_v}(t)}{\lambda_{p_2}^{\pi_v}(t)}, \qquad \forall t \in (t_{init}, t_{final}] \tag{16}$$

Proof. Let t be an arbitrary time. The FIFO assumption gives $\delta_{v,p}(t) = \delta_v(t)$. By definition of delay $\delta_{v,p}(t)$,

$$D_p^{\pi_v}(t) \quad = \quad D_p^v\left(t + \delta_{v,p}(t)\right) \ \forall p \in P_v$$

Taking the derivative with respect to t and using $\delta_{v,p}(t) = \delta_v(t)$,

$$\frac{dD_p^{\pi_v}(t)}{dt} \quad = \quad \left(1 + \frac{d\delta_v(t)}{dt}\right) \cdot \left.\frac{dD_p^v}{dt}\right|_{t+\delta_v(t)}$$

Using equation (8) we obtain,

$$\lambda_p^{\pi_v}(t) \quad = \quad \left(1+\frac{d\delta_v(t)}{dt}\right)\cdot\lambda_p^v(t+\delta_v(t))\ \ \forall p\in P_v \tag{17}$$

Therefore, it follows that

$$\frac{\lambda_{p_1}^v(t+\delta_v(t))}{\lambda_{p_2}^v(t+\delta_v(t))} \quad = \quad \frac{\lambda_{p_1}^{\pi_v}(t)}{\lambda_{p_2}^{\pi_v}(t)}$$

□

Definition 2.4. *Queue state η_v - state transitions*

We define queue state as the boolean valued function $\eta_v(t)$:

$$\eta_v(t)=\begin{cases}1 & \text{if } \delta_v(t)>0\\ 0 & \text{otherwise}\end{cases} \tag{18}$$

If $\eta_v=1$, queue v is said to be active, or in active state

If $\eta_v=0$, queue v is said to be inactive, or in inactive state

A queue state transition happens at time t if

$$\exists\epsilon>0 \text{ s.t. } \forall\theta\in\{(-\epsilon,0)\cup(0,\epsilon)\},\ \eta_v(t-\theta)=1-\eta_v(t+\theta) \tag{19}$$

When queue v is inactive, $D^v=D^{\pi_v}$.

Definition 2.5. *Link constraint $c_{v,l}(t)$*

Let $v\in V\backslash\{v_0\cup S\}$ be a node which is not a source or a sink. For all links $l\in L_v^{\text{out}}$, we define the link constraint $c_{v,l}(t)$ as the ratio of arriving flows at time t on capacity at queue v when this flow leaves queue v[5].

$$c_{v,l}(t)=\frac{\sum_{p\in P_l}\lambda_p^{\pi_v}(t)}{\mu_l(t+\delta_v(t))} \tag{20}$$

Definition 2.6. *Active link $\gamma_v(t)$ and set of active paths $\Gamma_v(t)$ of a node*

We define the active link $\gamma_v(t)$ of a node v at time t as the most constrained link [6] in L_v^{out}:

$$\gamma_v(t)\in\underset{l\in L_v^{\text{out}}}{\arg\max}\, c_{v,l}(t) \tag{21}$$

We define the set of active paths $\Gamma_v(t)$ in queue v as the set of paths in the most constrained link $\gamma_v(t)$

$$\Gamma_v(t)=P_{\gamma_v(t)} \tag{22}$$

Remark 1 gives $\Gamma_v\subset P_v$.

Requirement 3. *Full capacity discharge property*

The model satisfies the full capacity discharge property. For each node v and time t, if queue v is active at t, then the active link $\gamma_v(t)$ discharges at full capacity.

$$\eta_v(t)=1\Rightarrow\sum_{p\in\Gamma_v(t)}\lambda_p^v(t+\delta_v(t))=\mu_{\gamma_v(t)}(t+\delta_v(t)) \tag{23}$$

With this last property, we complete the definition of the dynamics model.

[5] The dissipation rate of the point queue at the node is only governed by the capacities of the outgoing links. This model can be extended to also impose a discharge rate constraint based on the capacity of the incoming link, but increases the complexity of the notation and the proofs.

[6] When there is a tie, one of them is chosen arbitrarily.

Definition 2.7. Feasible flows

A feasible flow λ_p^v at a node v is a flow that satisfies the FIFO, capacity constraint and full capacity discharge properties from requirements 1, 2 and 3.

The definition of the initial conditions on the network completes the definition of the model.

Definition 2.8. *Initial times for each non-source node*

Given a set of initial delays at each node $\delta_v\left(t_{\text{initial}}\right) \geq 0, \forall v \in V \backslash (S \cup \{v_0\})$ and an initial time t_{initial}, we define the set of initial times over which the departure curves are defined for each non-source node recursively as follows:

$$\begin{cases} t_{0,\text{initial}} & = & t_{\text{initial}} \text{ for node } v_0 \\ t_{v,\text{initial}} & = & t_{\pi_v,\text{initial}} + \delta_v\left(t_{\pi_v,\text{initial}}\right) \end{cases} \tag{24}$$

2.4. **Existence and uniqueness of the solution to the model.** Now that we have fully defined the model dynamics, we consider the well-posedness of the model. In other words, given a network, link capacities and the departure functions at the source, we want to know whether the dynamics of the model admits a unique solution.

Problem 1: General network problem

Input. An arborescence (V, L) with source v_0 and sink set S, capacities $\mu_l\left(t\right), \forall l \in L,\ t \in [t_{\text{initial}}, t_{\text{final}}]$, acceptable departure functions from the source $D_p^{v_0} \in \mathbf{D}(t_{\text{initial}}, t_{\text{final}})\ \forall p \in P_{v_0}$ and initial delays $\delta_v\left(t_{\text{initial}}\right) \geq 0,\ \forall v \in V \backslash (S \cup \{v_0\})$

Question. Does a corresponding set of feasible flows exist for all internal nodes $v \in V \backslash v_0$ and are they unique?

Theorem 1 states that the solution to problem 1 both exists and that the solution is unique, under certain conditions on the departure curves at the origin and the link capacities of the network.

Theorem 1. ***Existence and uniqueness of the solution to problem 1***

Problem 1 admits a unique solution under the following conditions.

1) the path flows at the origin $\lambda_p^0\left(t\right)$ are piecewise polynomial,

2) link capacities μ_l are piecewise constant over time.

Note that neither of the assumptions of the theorem are restrictive in a practical sense[7].

The next section is devoted to a constructive proof of Theorem 1. The general flow of the proof is as follows. Sections 3.1-3.3 first develop a set of differential equations for delays in the network. In section 3.4, we then prove that a unique solution to differential equation on delays also implies a unique solution to problem 1. Section 3.5 proves that the differential equations on the delay at each node always admit an unique solution, which finally leads to the proof of Theorem 1.

3. **A solution based on time mapping.** This section builds a constructive proof of Theorem 1. Throughout sections 3.1-3.3, we require that the flows at the origin are acceptable departure curves as defined in definition 2.1 and that the outflows

[7] Neither of these assumptions are restrictive in a practical sense, because any piecewise continuous function on a closed interval can be approximated to an arbitrary accuracy by a polynomial of appropriate degree (Stone-Weierstrass theorem [50]) and link capacities do not evolve in a continuous manner. Link capacities are typically subject to discrete changes due to incidents such as accidents and changes in weather.

at each node satisfy the model requirements (i.e. result in feasible flows as defined in definition 2.7).

3.1. **Local study of point queues.** We begin by proving Proposition 2, which gives an analytical expression for the derivative of the delay at node as a function of its downstream capacities and outgoing flow at its parent nodes.

Proposition 2. *Evolution law of a single queue*
If queue v is active at time t,

$$\left.\frac{d\delta_v}{dt}\right|_t = \frac{\sum_{p\in\Gamma_v(t)} \lambda_p^{\pi_v}(t)}{\mu_{\gamma_v(t)}(t+\delta_v(t))} - 1 \tag{25}$$

Proof. From Equation (17),

$$\begin{aligned}
\lambda_p^{\pi_v}(t) &= \left(1 + \frac{d\delta_v(t)}{dt}\right)\cdot\lambda_p^v(t+\delta_v(t)) && \forall p\in\Gamma_v(t)\\
\Rightarrow \frac{d\delta_v(t)}{dt} + 1 &= \frac{\lambda_p^{\pi_v}(t)}{\lambda_p^v(t+\delta_v(t))} && \forall p\in\Gamma_v(t)\\
\Rightarrow \frac{d\delta_v(t)}{dt} + 1 &= \frac{\sum_{p\in\Gamma_v(t)} \lambda_p^{\pi_v}(t)}{\sum_{p\in\Gamma_v(t)} \lambda_p^v(t+\delta_v(t))}
\end{aligned}$$

Using the full capacity discharge property from Requirement 3,

$$\begin{aligned}
\frac{d\delta_v(t)}{dt} + 1 &= \frac{\sum_{p\in\Gamma_v(t)} \lambda_p^{\pi_v}(t)}{\mu_{\gamma_v(t)}(t+\delta_v(t))}\\
\Rightarrow \left.\frac{d\delta_v}{dt}\right|_t &= \frac{\sum_{p\in\Gamma_v(t)} \lambda_p^{\pi_v}(t)}{\mu_{\gamma_v(t)}(t+\delta_v(t))} - 1
\end{aligned}$$

3.2. **Time mapping.** The evolution law stated above for any given node v depends on the outgoing flows $\lambda_p^{\pi_v}$ at the parent node. However, this it not an input of Problem 1. In this section, we introduce the notion of time mapping to obtain a modified law for the delay evolution that replaces the outgoing flows at the parent node with the outgoing flow at the origin.

3.2.1. *Definition of time mapping functions.* The evolution law from Proposition 2 gives a non-linear *ordinary differential equation* (ODE) that governs the evolution of $\delta_v(t)$. The evolution of delay encountered by an agent x entering queue v at time t depends on the flows entering the queue at t and the capacity of the active link(s) γ_v at time $t+\delta_v(t)$ when agent x leaves the queue. The non-linearity of the ODE makes directly computing the dynamics along a path algebraically complex. Therefore, we introduce a time mapping function.

Let v be an internal node of the network and its parent node be π_v. An agent leaving node π_v at time t will leave node v at time $t+\delta_v(t)$. We now introduce the following time mapping function:

Definition 3.1. *Node time mapping function T^{v,π_v}*

We define the time mapping function T^{v,π_v} by

$$T^{v,\pi_v} : t \mapsto t + \delta_v(t) \tag{26}$$

an agent leaving node π_v at time t will leave node v at time $T^{v,\pi_v}(t)$

The notation T^{v,π_v} (variable ordering) is chosen for mathematical convenience with respect to the derivatives of the function, as will be apparent in the rest of the discussion. In equation (26), T^{v,π_v} takes a time with a physical meaning at the exit of node π_v on its right hand side, and gives back a time with a physical meaning at the exit of node v on its left hand side.

Proposition 3. *T^{v,π_v} is strictly increasing and bijective*

The function T^{v,π_v} is strictly increasing and thus bijective from its domain to its image. Its derivative is

$$\left.\frac{dT^{v,\pi_v}}{dt}\right|_t = \frac{\sum_{p\in\Gamma_v(t)} \lambda_p^{\pi_v}(t)}{\mu_l\left(t+\delta_v(t)\right)} > 0 \tag{27}$$

Physically, this means that the FIFO assumption is respected: i.e. an agent x_2 entering queue v after another agent x_1 will also leave the queue after x_1

Proof. Taking the derivative of equation (26) and applying equation (25) in Proposition 2 gives,

$$\left.\frac{dT^{v,\pi_v}}{dt}\right|_t = \frac{\sum_{p\in\Gamma_v(t)} \lambda_p^{\pi_v}(t)}{\mu_l\left(t+\delta_v(t)\right)}. \tag{28}$$

The departure curves at the origin are strictly increasing since they must be acceptable departure curves. The full capacity discharge property from requirement 3 requires that one outgoing link at each node discharges at full capacity. Finally, these properties combined with Proposition 1, which states that the outflows at a node are proportional to the inflows, give us the result that $\left.\frac{dT^{v,\pi_v}}{dt}\right|_t > 0$. □

Thus T^{v,π_v} is invertible and its inverse is an increasing function[8].

Definition 3.2. *Node time mapping function $T^{\pi_v,v}$*

Given an internal node v, we define the function $T^{\pi_v,v}$ as the inverse of T^{v,π_v}

$$T^{\pi_v,v} \circ T^{v,\pi_v} = \mathbb{1} \text{ and } T^{v,\pi_v} \circ T^{\pi_v,v} = \mathbb{1} \tag{29}$$

We now consider the unique path $(v_0, v_1, \cdots, v_{n-1}, v_n)$ which leads from the source v_0 to some node v_n. As each node has a unique parent, we can recursively trace the path from node v back to the source node v_0. Let t^{v_n} be a fixed time. If an agent x leaves node v_n at the time t^{v_n}, we can recursively define the following:
1) $t^{v_{n-1}}{=}T^{v_{n-1},v_n}(t^{v_n})$ is the time that agent x left v_{n-1}, $t^{v_n}{=}t^{v_{n-1}} + \delta_v(t^{v_{n-1}})$
2) $t^{v_{n-2}}{=}T^{v_{n-2},v_{n-1}}(t^{v_{n-1}})$ is the time that agent x left v_{n-2},
$t^{v_n}{=}t^{v_{n-2}} + \delta_{v_{n-1}}(t^{v_{n-2}}) + \delta_v(t^{v_{n-2}} + \delta_{v_{n-1}}(t^{v_{n-2}}))$
3) $t^{v_{n-3}}{=}T^{v_{n-3},v_{n-2}}(t^{v_{n-2}})$ is the time that agent x left v_{n-3}, $\cdots$
As T^{v,π_v} and $T^{\pi_v,v}$ are bijective for all internal nodes v, we can give the following definition

Definition 3.3. *Time mapping function from and to the origin T^{v,v_0} and $T^{v_0,v}$*

Let v_n be a node, and $(v_0, v_1, v_2, \cdots, v_{\pi_n}, v_n)$ be a path from the origin v_0 to node v. We define the time mapping function to the origin as the composition of the node time mapping function on the path between the source and v_n

$$T^{v_0,v_n} = T^{v_0,v_1} \circ T^{v_1,v_2} \circ \cdots \circ T^{v_{\pi_n},v_n} \tag{30}$$

[8] If the acceptable set of departure curves $\mathbf{D}$ is relaxed to allow non-decreasing instead of strictly increasing functions, T^{v,π_v} becomes a correspondence, and the mathematical treatment would be more involved.

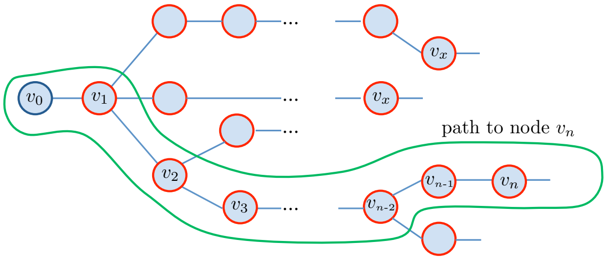

FIGURE 2. Time mapping nodes

an agent that leaves node v_n at time t left the origin v_0 at time $T^{v_0,v_n}(t)$.

$$T^{v_n,v_0} = T^{v_n,v_{\pi_n}} \circ \cdots \circ T^{v_2,v_1} \circ T^{v_1,v_0} \tag{31}$$

an agent that leaves the origin at time t will leave node v_n at time $T^{v_n,v_0}(t)$

A sample path from the origin v_0 to a node v_n is illustrated in figure 2. We can now define the time mapping function between any arbitrary pair of nodes.

Definition 3.4. *Time mapping function between two arbitrary nodes*

We define the time mapping function $T^{i,j}$ between node i and node j as follows. 1) There exists a path between nodes i and j (for example nodes v_2 and v_n in figure 2),

$$T^{i,j} = \begin{cases} T^{i,i+1} \circ T^{i+1,i+2} \circ \cdots \circ T^{j-2,j-1} \circ T^{j-1,j} & \text{if } i \prec j \\ T^{i,i-1} \circ T^{i-1,i-2} \circ \cdots \circ T^{j+2,j+1} \circ T^{j+1,j} & \text{if } i \succ j \end{cases} \tag{32}$$

Let x be an agent that leaves node j at time t. $T^{i,j}(t)$ is the time that agent x leaves node j.

2) There does not exist a path between nodes i and j (for example nodes v_2 and v_x in figure 2),

$$T^{i,j} = T^{i,v_0} \circ T^{v_0,j} \tag{33}$$

Let x_j be an agent that leaves node j at t. From definition 3.3 we know that x_j leaves the origin at time $T^{0,j}(t)$. Let x_i be an agent that also leaves the origin at time $T^{0,j}(t)$. Then $T^{i,j}(t)$ is the time that agent x_i leaves node i.

Definition 3.5. *Time mapping operator* $\mathbf{T}^{i,j}$

We define the time mapping operator $\mathbf{T}^{i,j}$ on the set F of time dependent functions as follows:

$$\mathbf{T}^{i,j}: \begin{array}{ccc} F & \rightarrow & F \\ f & \mapsto & f \circ T^{j,i} \end{array} \tag{34}$$

We now consider the physical interpretation of $T^{i,j}$.

3.2.2. *Time mapping of model quantities.* This section first studies the relationship between departure curves at different nodes and the time mapping function. We then define the time mapped versions of the other quantities in the model. The time mapping operators allow for mapping any quantity from one node to the other. This definition of a time mapped quantities thus allows any quantity to be defined with respect to the source node of the network.

Proposition 4. *Physical interpretation of the time mapping function*

Let p be a path, and $(v_0, v_1, v_2, \cdots, v_n)$ be a sequence of consecutive nodes on the path.

$$D_p^{v_i} = D_p^{v_0} \circ T^{v_0, v_i} \qquad \forall v_i \in p \tag{35}$$

Let $x = D_p^{v_0}\left(t^{v_0}\right)$ be an agent on path p that leaves the origin at time t^{v_0} and $t^{v_i} = T^{v_i,0}\left(t^{v_0}\right) \forall v_i \in p$.

$$D_p^{v_0}\left(t^{v_0}\right) = D_p^{v_1}\left(t^{v_1}\right) = \cdots = D_p^{v_i}\left(t^{v_i}\right) = \cdots = D_p^{v_n}\left(t^{v_n}\right) \tag{36}$$

Proof. Proof by induction on the length of the sequence k. If $k = 0$, the result is trivial. Let $k \in [1, i]$ be an integer. By the induction hypothesis, we assume that the result is true for to $k = i - 1$, i.e. $D_p^{v_{i-1}} = D_p^{v_0} \circ T^{0, v_{i-1}}$. By the definition of path delay $\delta_{v,p}, D_p^{v_i}\left(t + \delta_{v_i,p}(t)\right) = D_p^{v_{i-1}}(t), \forall t$, which means $D_p^{v_{i-1}} = D_p^{v_i} \circ T^{v_i, v_{i-1}}$. Composing both sides of the equality with T^{v_{i-1}, v_i} we get $D_p^{v_i} = D_p^{v_{i-1}} \circ T^{v_{i-1}, v_i}$. Substituting the induction hypothesis and simplifying the results completes the proof.

$$\begin{aligned} D_p^{v_i} &= D_p^{v_{i-1}} \circ T^{v_{i-1}, v_i} \\ &= D_p^{v_0} \circ T^{0, v_{i-1}} \circ T^{v_{i-1}, v_i} \\ &= D_p^{v_0} \circ T^{v_0, v_i} \end{aligned}$$

Equation (36) follows directly from equation (35) □

Remark 6. As function $T^{i,j}$ is the inverse of $T^{j,i}$, the operator $\mathbf{T}^{j,i}$ is the inverse of $\mathbf{T}^{i,j}$.

We can now reformulate the first equation of Proposition 4 as follows:

Proposition 5. *Time mapping of departure curve D_p^v*

Let i and j be two nodes on path p.

$$D_p^i = \mathbf{T}^{i,j}(D_p^j) \tag{37}$$

Proof. Using definition 3.5 we have,

$$\begin{aligned} \mathbf{T}^{i,j}\left(D_p^j\right) &= D_p^j \circ T^{j,i} = D_p^j \circ T^{j,0} \circ T^{0,i} \\ &= D_p^0 \circ T^{0,i} = D_p^i \end{aligned}$$

□

Proposition 6. *Time mapping and flows*

Let v be a node on path p.

$$\lambda_p^i = \mathbf{T}^{i,j}\left(\lambda_p^j\right) \cdot \frac{dT^{j,i}}{dt} \tag{38}$$

Proof. From the definition of flow, $\lambda_p^v = \dfrac{dD_p^v}{dt}$. The result is obtained by simply taking the derivative of the equation $D_p^i = D_p^j \circ T^{j,i}$ (from Proposition 5) with respect to time. □

Remark 7. The time mapping and derivative operators do not commute.

Definition 3.6. *Time mapping of delay δ_j^i*

Let v be an internal node[9]. We define the time mapped delay in queue v at node π_v, $\delta_v^{\pi_v}$ as the delay encountered in queue v by an agent leaving node π_v:

$$\delta_v^{\pi_v} \doteq \delta_v \tag{39}$$

Let i be an arbitrary node and j be an internal node. We define the time mapped delay in queue j at node i, δ_j^i as

$$\delta_j^i \doteq \mathbf{T}^{i,\pi_j}\left(\delta_j^{\pi_j}\right) = \delta_j^{\pi_j} \circ T^{\pi_j,i} \tag{40}$$

Physically, if nodes i and j are on the same branch with $i \prec j$ (resp. $i \succ j$), then $\delta_j^i(t)$ is the time that an agent which leaves queue i at time t will be (resp. has been) delayed at in queue j.

Definition 3.7. *Time mapping for capacity*

We define the time mapped capacity of a link l, $\mu_l^{v_l^{\text{in}}}$ as the capacity encountered by an agent at queue v_l^{in} in link l

$$\mu_l^{v_l^{\text{in}}} \doteq \mu_l \tag{41}$$

Let l be an arbitrary link and v an internal node. We define the time mapped capacity of link l at node v as

$$\mu_l^v \doteq \mathbf{T}^{v,v_l^{\text{in}}}\left(\mu_l^{v_l^{\text{in}}}\right) = \mu_l^{v_l^{\text{in}}} \circ T^{v_l^{\text{in}},v} \tag{42}$$

Physically, if link l and node v are on the same branch with $v_l^{\text{in}} \prec v$ (resp. $v_l^{\text{in}} \succ v$), then $\mu_v^l(t)$ is the capacity an agent that leaves queue v at time t encountered (resp. encounters) at link l.

Proposition 7. *Physical interpretation of mapped delay and mapped capacity*

Let v_j be an arbitrary node, p be a path, and $(v_0, v_1, v_2, \cdots, v_n)$ be a sequence of consecutive nodes on the path p. Also, let $t^{v_i} = T^{v_i,0}(t^{v_0}), \forall v_i \in p$.

$$\delta_{v_j}^{v_0}(t^{v_0}) = \delta_{v_j}^{v_1}(t^{v_1}) = \cdots = \delta_{v_j}^{v_i}(t^{v_i}) = \cdots = \delta_{v_j}^{v_n}(t^{v_n}) \tag{43}$$

Let l be an arbitrary link.

$$\mu_l^{v_0}(t^{v_0}) = \mu_l^{v_1}(t^{v_1}) = \cdots = \mu_l^{v_i}(t^{v_i}) = \cdots = \mu_l^{v_n}(t^{v_n}) \tag{44}$$

Proof. Let i be an arbitrary node and j be an internal node. From definition (3.6) for time mapped delay we have.

$$\begin{aligned} \delta_j^i\left(t^i\right) &\doteq \delta_j^{j-1}\left(T^{j-1,i}\left(t^i\right)\right) \\ &= \delta_j^{j-1}\left(t^{j-1}\right) \end{aligned}$$

Therefore, $\delta_{v_j}^{v_i}(t^{v_i}) = \delta_{v_j}^{v_{j-1}}(t^{v_{j-1}}), \forall v_i \in p$, which proves equation (43). The proof for equation (44) is identical. □

Definition 3.8. *Time mapping of active link and active paths*

Let v be an internal node. We define mapped active link $\gamma_v^{\pi_v}$ as the active link for flow exiting node π_v at queue v, and mapped active paths $\Gamma_v^{\pi_v}$ as the active paths for flow exiting node π_v at queue v.

$$\gamma_v^{\pi_v} \doteq \gamma_v \quad ; \quad \Gamma_v^{\pi_v} \doteq \Gamma_v \tag{45}$$

[9] An internal node is a node v which is neither a sink nor the source $k \in K \backslash (\{0\} \cup S)$.

Let j be an arbitrary node, we define the mapped active link and mapped paths for flow exiting queue v at node j as

$$\gamma_v^j = \mathbf{T}^{j,\pi_v}\left(\gamma_v^{\pi_v}\right) \quad ; \quad \Gamma_j^v = \mathbf{T}^{j,\pi_v}\left(\Gamma_v^{\pi_v}\right) \tag{46}$$

Physically, if node j and node v are on the same branch with $j \prec v$ (resp. $j \succ v$), then $\gamma_v^j(t)$ is the active link that an agent leaving node j at time t will encounter (resp. encountered) at queue v, and $\Gamma_j^v(t)$ are the corresponding active paths.

Definition 3.9. *Time mapped link constraint*

Let v be a internal node and $l \in L_v^{\text{out}}$. We define the mapped link constraint $c_{v,l}^{\pi_v}$ as the link constraint at link l for an agent leaving node π_v.

$$\begin{aligned} c_{v,l}^{\pi_v}(t) &\doteq \frac{\sum_{p\in P_l} \lambda_p^{\pi_v}(t)}{\mu_l\left(t+\delta_v(t)\right)} && (47)\\ &= \frac{\sum_{p\in P_l} \lambda_p^{\pi_v}(t)}{\mu_l^v\left(t+\delta_v(t)\right)} \\ &= \frac{\sum_{p\in P_l} \lambda_p^{\pi_v}(t)}{\mu_l^{\pi_v}(t)} && (48)\end{aligned}$$

Let j be an arbitrary node, we define the mapped link constraint for link l at node j as

$$c_{v,l}^j \doteq \mathbf{T}^{j,\pi_v}\left(c_{v,l}^{\pi_v}\right) = c_{v,l}^{\pi_v} \circ T^{\pi_v,j} \tag{49}$$

$$c_{v,l}^j(t) = \frac{\sum_{p\in P_l} \lambda_p^j(t)}{\mu_l^j(t)} \cdot \frac{dT^{\pi_v,j}}{dt} \tag{50}$$

Physically, if node j and node v are on the same branch with $j \prec v$ (resp. $j \succ v$), then $c_{v,l}^j(t)$ is the link constraint that an agent leaving node j at time t will encounter (resp. encountered) at link l.

Remark 8. The notation of the link constraint can be simplified for convenience as follows when time mapped.

$$c_{v,l}^j = c_l^j \tag{51}$$

We use the simplified notation in the rest of the discussion.

Proposition 8. *The mapping of link constraints and active links is coherent*

For all non-sink nodes $j \in V\backslash S$, internal nodes $v \in V\backslash(S \cup \{0\}$ and time $t \in (t_{initial}, t_{final}]$, we have

$$\gamma_v^j(t) \in \underset{l\in L_v^{out}}{\arg\max}\, c_l^j(t) \tag{52}$$

Proof. Let v be an internal node and let t^j be a time. Let $t^{\pi_v} = T^{\pi_v,j}\left(t^j\right)$. Proving the proposition is equivalent to proving the following set equality

$$\underset{l\in L_v^{\text{out}}}{\arg\max}\, c_{v,l}\left(t^{\pi_v}\right) = \underset{l\in L_v^{\text{out}}}{\arg\max}\, c_l^j\left(t^j\right) \tag{53}$$

From the definition of the link constraint in equation (20) we have

$$c_{v,l}\left(t^{\pi_v}\right) \quad \doteq \quad \frac{\sum_{p\in P_l} \lambda_p^{\pi_v}\left(t^{\pi_v}\right)}{\mu_l\left(t^{\pi_v}+\delta_v\left(t^{\pi_v}\right)\right)} \tag{54}$$

By definition of μ_l^v in equation (41), we have $\mu_l\left(t^{\pi_v}+\delta_v\left(t^{\pi_v}\right)\right)=\mu_l^v\left(t^{\pi_v}+\delta_v\left(t^{\pi_v}\right)\right)$ and defining $t^v \doteq T^{v,\pi_v}\left(t^{\pi_v}\right)=t^{\pi_v}+\delta_v\left(t^{\pi_v}\right)$, we obtain
$\mu_l\left(t^{\pi_v}+\delta_v\left(t^{\pi_v}\right)\right)=\mu_l^v\left(T^{v,\pi_v}\left(t^{\pi_v}\right)\right)=\mu_l^v\left(t^v\right)$. Equation (44) finally gives

$$\mu_l\left(t^{\pi_v}+\delta_v\left(t^{\pi_v}\right)\right)=\mu_l^j\left(t^j\right) \tag{55}$$

Moreover, using equation (38) gives $\lambda_p^{\pi_v}\left(t^{\pi_v}\right)\cdot \left.\frac{dT^{\pi_v,j}}{dt}\right|_{t^j}=\lambda_p^j\left(t^j\right)$. Summing on all paths p in P_l, we obtain

$$\sum_{p\in P_l}\lambda_p^{\pi_v}\left(t^{\pi_v}\right)=\frac{1}{\left.\frac{dT^{\pi_v,j}}{dt}\right|_{t^j}}\cdot\sum_{p\in P_l}\lambda_p^j\left(t^j\right) \tag{56}$$

Substituting equations (55) and (56) in the right hand side of equation (54) and using the time mapped link constraint from equation (50), we obtain

$$\begin{aligned} c_{v,l}\left(t^{\pi_v}\right) &= \frac{1}{\left.\frac{dT^{\pi_v,j}}{dt}\right|_{t^j}}\cdot\left[\frac{\sum_{p\in P_l}\lambda_p^j\left(t^j\right)}{\mu_l^j\left(t^j\right)}\right] && (57)\\ &= \frac{1}{\left.\frac{dT^{\pi_v,j}}{dt}\right|_{t^j}}\cdot c_l^j\left(t^j\right) && (58)\end{aligned}$$

For all $l\in L_v^{\text{out}}$, $c_{v,l}\left(t^{\pi_v}\right)$ and $c_l^j\left(t^j\right)$ are proportional (and the proportionality ratio is independent from l). Therefore, the arg max in equation (53) are the same. which concludes the proof. □

Definition 3.10. *Capacity of the active link*

For notational simplicity we denote the capacity of the active link of an agent that enters queue v at time t as follows:

$$\begin{aligned} Q_v(t) &\doteq \mu_{\gamma_v(t)}^{\pi_v}(t) && (59)\\ &= \mu_{\gamma_v(t)}^{v}\left(t+\delta_v(t)\right) \\ &= \mu_{\gamma_v(t)}\left(t+\delta_v(t)\right) && (60)\end{aligned}$$

Definition 3.11. *Time mapped capacity of the active link*

Let v be a internal node. We define the time mapped active link capacity $Q_v^{\pi_v}$ as the capacity of link γ_v as seen by an agent at node π_v.

$$Q_v^{\pi_v}\doteq Q_v \tag{61}$$

Let j be an arbitrary node, we define the mapped active link capacity for link $\gamma_v(t)$ as seen by an agent at node j as

$$Q_v^j\doteq \mathbf{T}^{j,\pi_v}\left(Q_v^{\pi_v}\right)=Q_v^{\pi_v}\circ T^{\pi_v,j}=Q_v\circ T^{\pi_v,j} \tag{62}$$

Physically, if node j and node v are on the same branch with $j\prec v$ (resp. $j\succ v$), then $Q_v^j(t)$ is the active link capacity that an agent leaving node j at time t will encounter (resp. encountered) at link $\gamma_v^j(t)$.

Definition 3.12. *Time mapping of queue state*

Let i be an arbitrary node and j be an internal node. We define the time mapped queue state of queue j at node i, η_j^i as the queue state at queue j as seen by an agent at queue i

$$\eta_j^i \doteq \mathbf{T}^{i,\pi_v}\left(\eta_j^{\pi_v}\right) = \eta_j^{\pi_v} \circ T^{\pi_v,i} = \eta_j \circ T^{\pi_v,i} \tag{63}$$

Physically, if queue i and node j are on the same branch with $i \prec j$ (resp. $i \succ j$), then $\eta_j^i\,(t)$ is the queue state an agent that leaves node i at time t encounters (resp. encountered) at queue j.

3.3. **Global evolution of delay.** We now have the necessary tools to define the evolution of delays at any node of the network with respect to the flows at any upstream node in the network.

Definition 3.13. *First active upstream node*

Let v be an internal node. We define the first active upstream node of v as

$$\Upsilon_v^j\,(t) = \max_{\succ}\left\{u | u \prec v, \eta_u^j\,(t) = 1\right\} \tag{64}$$

For notational convenience we also define the following:

$$\begin{aligned}
\hat{\gamma}_v^j\,(t) &\doteq \gamma^j_{\Upsilon_v^j(t)}\,(t) & (65)\\
\hat{\Gamma}_v^j\,(t) &\doteq \Gamma^j_{\Upsilon_v^j(t)}\,(t) & (66)\\
\hat{Q}_v^j\,(t) &\doteq Q^j_{\Upsilon_v^j(t)}\,(t) & (67)\\
\hat{\eta}_v^j\,(t) &\doteq \eta^j_{\Upsilon_v^j(t)}\,(t) & (68)
\end{aligned}$$

Theorem 2. ***Evolution law for delay at an arbitrary internal node*** v ***mapped to any node*** j

Given an arbitrary internal node $v \in V \backslash (S \cup \{0\})$ such that queue v is active, if the flows at the origin are acceptable departure curves and the model requirements are satisfied, the evolution law for delay mapped to any upstream node $j \in V \backslash S$ is

$$\left.\frac{d\delta_v^j}{dt}\right|_t = \begin{cases} \dfrac{\sum_{p\in\Gamma_v^j(t)} \lambda_p^j\,(t)}{Q_v^j\,(t)} - \left.\dfrac{dT^{0,j}}{dt}\right|_t \\ \text{if } v \text{ is the first active queue} \in p \\ \dfrac{\sum_{p\in\Gamma_v^j(t)} \lambda_p^j\,(t)}{Q_v^j\,(t)} - \dfrac{\sum_{p\in\hat{\Gamma}_p^j(t)} \lambda_p^j\,(t)}{\hat{Q}_v^j\,(t)} & \text{otherwise} \end{cases} \tag{69}$$

Proof. Let t be a time and v be a node. Evolution law (25) in Proposition 2 gives

$$\left.\frac{d\delta_v}{dt}\right|_t = \frac{\sum_{p\in\Gamma_v(t)} \lambda_p^{\pi_v}\,(t)}{Q_v\,(t)} - 1 \tag{70}$$

By the definition of the time mapping functions we have, $\delta_v^{\pi_v}\,(t) \doteq \delta_v\,(t)\,, Q_v^{\pi_v}\,(t) \doteq Q_v\,(t)\,, \Gamma_v^{\pi_v}\,(t) \doteq \Gamma_v\,(t)$. Thus, equation (70) becomes:

$$\left.\frac{d\delta_v^{\pi_v}}{dt}\right|_t = \frac{\sum_{p\in\Gamma_v^{\pi_v}(t)} \lambda_p^{\pi_v}\,(t)}{Q_v^{\pi_v}\,(t)} - 1 \tag{71}$$

Case 1: If node v is not the first active node of path p and $\Upsilon_v\,(t)$ exists.

Let for an arbitrary node j, $t^j = T^{j,\pi_v}(t)$. Since all the nodes between $\Upsilon_v(t)$ and π_v are inactive by the definition of $\Upsilon_v(t)$ (note that $\Upsilon_v(t)$ may be equal to π_v), we have

$$t^{\Upsilon_v(t^{\pi_v})} = t^{\pi_v} = t \tag{72}$$

Furthermore, since $\hat{\eta}_v^{\pi_v}(t) = 1$, and the full capacity discharge of active links (assumption 3), we have

$$\sum_{p \in \hat{\Gamma}_v^{\pi_v}(t)} \lambda_p^{\Upsilon_v(t)}(t) = \hat{Q}_v^{\pi_v}(t) \tag{73}$$

$$\sum_{p \in \hat{\Gamma}_v^{\pi_v}(t)} \lambda_p^{\pi_v}(t) = \hat{Q}_v^{\pi_v}(t) \tag{74}$$

Thus:

$$\frac{\sum_{p \in \hat{\Gamma}_v^{\pi_v}(t)} \lambda_p^{\pi_v}(t)}{\hat{Q}_v^{\pi_v}(t)} = 1 \tag{75}$$

By replacing the constant 1 in equation (71) with the above result we get,

$$\left.\frac{d\delta_v^{\pi_v}}{dt}\right|_t = \frac{\sum_{p \in \Gamma_v^{\pi_v}(t)} \lambda_p^{\pi_v}(t)}{Q_v^{\pi_v}(t)} - \frac{\sum_{p \in \hat{\Gamma}_v^{\pi_v}(t)} \lambda_p^{\pi_v}(t)}{\hat{Q}_v^{\pi_v}(t)} \tag{76}$$

This gives us the result for $j = \pi_v$. We will now map this result to any node $j \in V \backslash S$. By definition of time mapping, we have

$$\delta_v^j = \delta_v^{\pi_v} \circ T^{\pi_v,j} \tag{77}$$

Taking its derivative with respect to time, we obtain

$$\frac{d\delta_v^j}{dt} = \left[\frac{d\delta_v^{\pi_v}}{dt} \circ T^{\pi_v,j}\right] \cdot \frac{dT^{\pi_v,j}}{dt} \tag{78}$$

$$\left.\frac{d\delta_v^j}{dt}\right|_t = \left[\frac{\sum_{p \in \Gamma_v^{\pi_v} \circ T^{\pi_v,j}(t)} \lambda_p^{\pi_v} \circ T^{\pi_v,j}(t)}{Q_v^{\pi_v} \circ T^{\pi_v,j}(t)} - \right.$$

$$\left.\frac{\sum_{p \in \Gamma_{\Upsilon_v t}^{\pi_v} \circ T^{\pi_v,j}(t)} \lambda_p^{\pi_v} \circ T^{\pi_v,j}(t)}{Q_{\Upsilon_v(t)}^{\pi_v} \circ T^{\pi_v,j}(t)}\right] \cdot \left.\frac{dT^{\pi_v,j}}{dt}\right|_t \tag{79}$$

Equation (38) on flow mapping gives

$$\left(\lambda_p^{\pi_v} \circ T^{\pi_v,j}(t)\right) \cdot \left.\frac{dT^{\pi_v,j}}{dt}\right|_t = \lambda_p^j(t) \tag{80}$$

Substituting this result and the simple time mapping transformations of λ and Q into equation (79) gives the final result

$$\left.\frac{d\delta_v^j}{dt}\right|_t = \frac{\sum_{p \in \Gamma_v^j(t)} \lambda_p^j(t)}{Q_v^j(t)} - \frac{\sum_{p \in \Gamma_{\Upsilon_v(t)}^j(t)} \lambda_p^j(t)}{Q_{\Upsilon_v(t)}^j(t)} \tag{81}$$

$$= \frac{\sum_{p \in \Gamma_v^j(t)} \lambda_p^j(t)}{Q_v^j(t)} - \frac{\sum_{p \in \hat{\Gamma}_v^j(t)} \lambda_p^j(t)}{\hat{Q}_v^j(t)} \tag{82}$$

Case 2: If node v is the first active node of path p, we leave the constant 1 in equation (71) and follow the same remaining steps as in case 1 to obtain the result.

$$\left.\frac{d\delta_v^j}{dt}\right|_t = \frac{\sum_{p\in\Gamma_v^j(t)} \lambda_p^j(t)}{Q_v^j(t)} - \left.\frac{dT^{0,j}}{dt}\right|_t \tag{83}$$

□

Applying Theorem 2 with $j = 0$, we see that the delays with respect to the flows at the origin δ_v^0 are solutions to the ordinary differential equations in definition 3.14.

Definition 3.14. *Time mapped delay evolution differential equation*

- If v is not an active node and the flow on its active link γ_0^v is within capacity, then $\frac{d\delta_v^0}{dt} = 0$.

comes directly from the definition of a segment of constant constrain

- If v is an active node or its active link γ_0^v is over capacity, then

$$\left.\frac{d\delta_v^0}{dt}\right|_t = \begin{cases} \frac{\sum_{p\in\Gamma_v^0(t)} \lambda_p^0(t)}{Q_v^0(t)} - 1 \\ \text{if } v \text{ is the first active queue} \in p \\ \frac{\sum_{p\in\Gamma_v^0(t)} \lambda_p^0(t)}{Q_v^0(t)} - \frac{\sum_{p\in\hat{\Gamma}_v^0(t)} \lambda_p^0(t)}{\hat{Q}_v^0(t)} \text{ otherwise} \end{cases} \tag{84}$$

where the time mapping functions are redefined from delays as follows:

$$T^{j,0} = \sum_{0 \prec i \preccurlyeq j} \delta_i^0 \tag{85}$$

Proposition 9. *Delay evolution does not depend on departure curves*

All the time mapped quantities in equations (84) *can be computed using only the initial delays, departure curve at the origin and the link capacities. It does not require the departure curves for any internal nodes $v \in V\backslash v_0$.*

Proof. The time mapping function only depends on the delay functions from definition 3.1. The time mapped flows can be obtained using the time mapping function using Proposition 6. The other time mapped quantities are by definition constructed using the time mapping function as given in section 3.2.2. □

3.4. **Equivalence of departure curves and delays.**

We prove Theorem 1 on the existence and uniqueness of Problem 1 by first showing the equivalence between Problem 1 and Problem 2 (defined below), and then proving the existence and uniqueness of Problem 2 in the next section.

Problem 2: General delay problem
Input. An arborescence (V, L) with source v_0 and sink set S, capacities $\mu_l(t), \forall l \in L,\ t \in [t_{\text{initial}}, t_{\text{final}}]$, departure functions from the source $D_p^{v_0} \in \mathbf{D}(t_{\text{initial}}, t_{\text{final}})$ $\forall p \in P_{v_0}$ and initial delays $\delta_v(t_{\text{initial}}) \geq 0,\ \forall v \in V\backslash(S \cup \{v_0\})$
Question. Does a solution to the time mapped delay function from definition 3.14 for each node $v \in V\backslash v_0$ exist and is it unique?

Theorem 3. *Problem (1) and problem (2) are equivalent*

Proof. The inputs to both problems are identical. Therefore, we only need to prove that the existence of a solution to one problem implies a unique and feasible corresponding solution to the other problem.

(⇒) Suppose first that Problem 1 admits a solution.
By the definition of delay,

$$\delta_v\left(t\right) = \left[D_p^v\right]^{-1}\left(D_p^{\pi_v}\left(t\right)\right) - t, \tag{86}$$

By the definition of time mapped delay we obtain,

$$\begin{aligned}\delta_v^0\left(t\right) &= \delta_v^{\pi_v} \circ T^{\pi_v,0}\left(t\right) &(87)\\ &= \delta_v^0\left(\left[D_p^v\right]^{-1}\left(D_p^0\left(t\right)\right)\right) &(88)\\ &= \delta_v\left(\left[D_p^v\right]^{-1}\left(D_p^0\left(t\right)\right)\right), &(89)\end{aligned}$$

which can be made a function of only D_p^v by equation (86).

Theorem 2 then ensures that the delay functions thus defined satisfy the time mapped delay evolution from definition 3.14, i.e. a feasible solution to problem 2. Furthermore, the solution is unique from equation (89), since D_p^v is a strictly increasing function.

(⇐) Suppose now that Problem 2 admits a solution $\delta_v^0\left(t\right)$. We can build the corresponding departure curves $D_p^v\left(t\right)$ as follows.

$$D_p^0\left(t\right) = \delta_0^0\left(t\right) \tag{90}$$

The inverse departure curve $\left[D_p^0\right]^{-1}\left(x\right)$ can be constructed from $D_p^0\left(t\right)$, since the departure curve is strictly increasing.

$$\left[D_p^v\right]^{-1}\left(x\right) = \left[D_p^0\right]^{-1}\left(x\right) + T^{v,0}\left(\left[D_p^0\right]^{-1}\left(x\right)\right) \tag{91}$$

The departure curve $D_p^v\left(t\right)$ can also be constructed from $\left[D_p^v\right]^{-1}\left(x\right)$ due to the strictly increasing nature of the functions.

We now show that the departure curves thus defined are feasible departure curves, i.e. a feasible solution to problem 1.

1. D_p^0 is continuous and piecewise C^1 because λ_p^0 is piecewise continuous. Furthermore, since $T^{j,0}$ is strictly increasing for all nodes j, D_p^v is continuous and piecewise C^1.
2. The capacity constraint on links is imposed by equation (84) due to Proposition 8.
3. The FIFO condition is satisfied by construction since the delay δ_v^0 is not a function of the path p.
4. The full capacity discharge of the active queues is also imposed by by equation (84) due to Proposition 8.

□

3.5. **Existence and uniqueness of the time mapped delay evolution.** This section proves Theorem 4 on the existence and uniqueness of the solution to Problem 2.

Theorem 4. ***Existence and uniqueness of the solution to problem (2)***

The solution to problem (2) exists and is unique on the time interval of the problem $[t_{initial}, t_{final}]$, *if the following conditions are satisfied.*

1. *the path flows at the origin* $\lambda_p^0(t)$ *are piecewise polynomial,*
2. *link capacities* μ_l *are piecewise constant over time.*

The proof of this theorem is fairly technical and requires several definitions and lemmas. Theorem 1 is a direct corollary of this result due to to Theorem 3 on the equivalence of the two problems.

The main goal of the proof of Theorem 4 is to show that there are a finite number of possible transitions, and to integrate equation (84) across the transitions. The next definitions and lemmas enables to establish these properties.

Definition 3.15. *Depth of a node* $d(v)$

We define the depth $d(v)$ of a node v as the number of links on the unique path from the origin v_0 to node v

Definition 3.16. *Link constraint comparators* $B_{(c_{l_1}, c_{l_2})}(t)$ *and* $B_{c_l}(t)$

Given a node v and two distinct links (l_1, l_2), we define the boolean comparator $B_{(c_{l_1}, c_{l_2})}(t)$ as follows:

$$B_{(c_{l_1}, c_{l_2})}(t) = \begin{cases} 1 & \text{if } \frac{\sum_{p \in P_{l_1}} \lambda_p^0(t)}{\mu_{l_1}^0(t)} > \frac{\sum_{p \in P_{l_2}} \lambda_p^0(t)}{\mu_{l_2}^0(t)} \\ 0 & \quad\quad\quad \textit{otherwise} \end{cases} \tag{92}$$

Given a node v and link $l \in L_v^{\text{out}}$, we define the boolean comparator $B_{c_l}(t)$ as follows:

$$B_{c_l}(t) = \begin{cases} 1 & \text{if } \frac{\sum_{p \in P_l} \lambda_p^0(t)}{\mu_l^0(t)} > 1 \\ 0 & \quad \textit{otherwise} \end{cases} \tag{93}$$

Definition 3.17. *Time segment of constant link constraint* J

A time segment J is a segment of constant link constraint if and only if

1. for each $l \in L$, the boolean $B_{c_l}(t)$ is constant on J,
2. for each pair of nodes $(l_1, l_2) \in L$, the boolean $B_{(c_{l_1}, c_{l_2})}(t)$ is constant on J.
3. for each $l \in L$, the time mapped link capacity $\mu_l^0(t)$ is constant on J.

Lemma 10. *Under the assumptions on flows and capacities, there are a finite number of segments of constant link constraint*

Proof. Consider a pair of links (l_1, l_2). Since capacities are piecewise constant and flows are piecewise polynomial, there are a finite number of segments on which the capacities are constant and flows are polynomial. On any such a segment, $\frac{\sum_{p \in P_{l_1}} \lambda_p^0(t)}{\mu_{l_1}^0(t)}$ and $\frac{\sum_{p \in P_{l_1}} \lambda_p^0(t)}{\mu_{l_1}^0(t)} - \frac{\sum_{p \in P_{l_2}} \lambda_p^0(t)}{\mu_{l_2}^0(t)}$ are polynomials. Therefore, the number of times each expression crosses zero is bounded by the degree of the polynomial, which implies that there are a finite number of segments of constant link constraint. □

Lemma 11. *Constant active link*

If J *is a segment of constant link constraint, the active link* γ_v^0 *of any node* v *is constant on* J.

Proof. The result comes directly from the definition of a segment of constant constraint. □

Definition 3.18. *Solution of depth n*

A solution of problem (2) for depth n is a set of solutions δ_v for all nodes v such that $d(v) < n$. It can be rigorously defined because the equations for δ_v only depend on variables associated with nodes of depth less than n.

Definition 3.19. *Elementary time segment $T^e\left(v\right)$*

Given a node v and a solution of depth $d(v) - 1$ (if v is not the origin), an elementary segment for node v is a time segment $T^e\left(v\right)$ such that

- $T^e(v)$ is a segment of constant constraint,
- If v is not the origin, for each node $j \in V$ such that $d\left(j\right) < d\left(v\right)$, the node state $\eta_j\left(t\right)$ is constant on $T^e(v)$.

Lemma 12. *Single transition of node state on an elementary segment*

If there exists a solution to problem (2) up to depth $d(v-1)$, and if $T^e(v) = [t_0, t_f]$ is an elementary segment for node v, then there is a solution δ_v^0 of the problem and node v admits at most one transition in $T^e(v)$.

Proof. As for each node $j \in V$ such that $d\left(j\right) < d\left(v\right)$, the node state $\eta_j\left(t\right)$ is constant on $T^e(v)$, the first active upstream node Υ_v is constant over time. Moreover, as $T^e(v)$ is a segment of constant constraint, Lemma 11 gives that active link γ_v and first active upstream link $\hat{\gamma}_v$ are constant on $T^e(v)$, and the sign of $\dfrac{\sum_{p\in\Gamma_v^0(t)}\lambda_p^0\left(t\right)}{Q_v^0\left(t\right)} - \dfrac{\sum_{p\in\hat{\Gamma}_v^0(t)}\lambda_p^0\left(t\right)}{\hat{Q}_v^0\left(t\right)}$ is constant on $T^e(v)$.

Let us now consider the following four cases:

1. $B_{(c_{\hat{\gamma}_v}, c_{\gamma_v})}\left(t_0\right) = 1, \eta_v^0\left(t\right) = 1 \implies \left.\dfrac{d\delta_v^0}{dt}\right|_t > 0$ and since the queue state is already active no transition will occur.
2. $B_{(c_{\hat{\gamma}_v}, c_{\gamma_v})}\left(t_0\right) = 1, \eta_v^0\left(t\right) = 0 \implies \left.\dfrac{d\delta_v^0}{dt}\right|_t > 0$ and the queue state will immediately transition to being active $\eta_v^0\left(t\right) = 1$. No further transitions will occur as shown above.
3. $B_{(c_{\hat{\gamma}_v}, c_{\gamma_v})}\left(t_0\right) = 0, \eta_v^0\left(t\right) = 1 \implies \left.\dfrac{d\delta_v^0}{dt}\right|_t \leq 0$ and the queue at node v starts dissipating. There will be a transition in the queue state to inactive $\eta_v^0\left(t\right) = 0$ if the queue dissipates by time t_f and the queue state will remain active otherwise.
4. $B_{(c_{\hat{\gamma}_v}, c_{\gamma_v})}\left(t_0\right) = 0, \eta_v^0\left(t\right) = 0 \implies \left.\dfrac{d\delta_v^0}{dt}\right|_t \leq 0$ and the only possibility is the strict equality case and the queue state remains inactive.

□

Lemma 13. *Unique solution on an elementary segment*

Let $T^e(v)$ be an elementary segment for node v. Assuming a solution of depth $d(v) - 1$ (if v is not the origin), then solution of equation (84) *for node v exists is unique on $T^e(v)$.*

Proof. By Lemma 12, there can be at most one state transition of node v in $T^e(v)$. This splits $T^e(v)$ into at most two sub-segments where $\eta_v = 0$ or $\eta_v = 1$. From Lemma 11 we have that active link γ_v and the first active upstream link Υ_v are constant on $T^e(v)$. Therefore, the quantities $\Gamma_v, \hat{\Gamma}_v, Q_v$ and $\hat{Q}_v$ are constant on $T^e(v)$. Equation (84) states that

- If v is not an active node ($\eta_v = 0$) and the flow on its active link γ_v is within capacity, then $\dfrac{d\delta_v^0}{dt} = 0$.
- If v is an active node ($\eta_v = 1$) or it's active link γ_v is over capacity,

$$\left.\frac{d\delta_v^0}{dt}\right|_t = \begin{cases} \dfrac{\sum_{p\in\Gamma_v^0(t)} \lambda_p^0\,(t)}{Q_v^0\,(t)} - 1 \\ \text{if } v \text{ is the first active queue} \in p \\ \dfrac{\sum_{p\in\Gamma_v^0(t)} \lambda_p^0\,(t)}{Q_v^0\,(t)} - \dfrac{\sum_{p\in\hat{\Gamma}_v^0(t)} \lambda_p^0\,(t)}{\hat{Q}_v^0\,(t)} \text{ otherwise} \end{cases} \tag{94}$$

As all the variables in equation (94) other than the flow $\lambda_p^0\,(t)$ are constant during an elementary segment $T^e(v)$ and the flow $\lambda_p^0\,(t)$ is continuous in t for all $t \in T^e(v)$, we can show that equation (94) admits a unique solution on the interval $T^e(v)$ by the Picard-Lindelöf theorem. □

We have now all the ingredients to prove Theorem 4.

Proof of Theorem 4: We will prove the following proposition: *The time interval of interest $(t_{initial}, t_{final}]$ can be partitioned into a finite set of elementary segments, and the solution to problem (2) exists and is unique*

The proof is done inductively over the depth of the network. If the network contains a single node v_0, $[t_{\text{initial}}, t_{\text{final}}]$ is an elementary segment for v_0, $(t_{\text{initial}}, t_{\text{final}}] \in T^e\,(v_0)$ and there is a unique solution by Lemma 13. By the induction hypothesis, let us now assume that $(t_{\text{initial}}, t_{\text{final}}]$ can be partitioned into a finite number of elementary segments with respect to all nodes of depth n and that the solution exists and is unique. Let $t_0, t_1, \cdots, t_m$ be times such that $E_n = \{(t_i, t_{i+1}], \forall i \in [0, m-1]\}$ is the set of elementary segments for nodes of depth n, and let δ_v for all $v \in \{V | d(v) \leq n\}$ be the unique solution of depth n.

Let K_n be the non-empty set of nodes of depth n, and let $v \in K_n$ be a node in this set. Lemma 12 gives that for each $v \in K_n$, there is at most one state transition on $(t_i, t_{i+1}]$. Let $F_n(v)$ be the set of times at which these transitions occur for node v. Since there are m elementary segments, there can at most be $|F_n\,(v)| \leq m$ transitions. If F_n is the set of times at which the transitions for all nodes of depth n happen, $|F_n| \leq m \cdot K_n$.

Let $\{t'_0, t'_1, \cdots, t'_{m'}\} = \{t_0, t_1, \cdots, t_m\} \cup F_n$ be the m' segments created by splitting E_n at each of the state transitions for nodes of depth n. The total number of segments m' satisfies $m' \leq m \cdot (K_n + 1)$, since $|F_n| \leq m \cdot K_n$. By the definition of the t'_i, for each $i \in [0, m']$ we have

- for all $v \in K_n$, η_v is constant on $(t'_i, t'_{i+1}]$,
- $(t'_i, t'_{i+1}]$ is a segment of constant constraint J, since it is subset of an elementary segment, which is already by definition a segment of constant constraint.

Thus, $[t'_i, t'_{i+1}]$ is an elementary segment for all nodes of depth $(n+1)$. Furthermore, by Lemma 13, this implies that there is an unique solution to all nodes of depth $n+1$, which concludes the proof. □

This also completes the proof of Theorem 1.

Proof of Theorem 1. Problem 1 is equivalent to Problem 2 by Theorem 3 and Problem 2 admits a unique solution by Theorem 4. □

In some applications, it is also important to be able to computing the total delay experienced by an agents that takes a particular path. A provides analytical expressions for the total delay along a path.

4. **Applications.** The solution to problem (1) models the flows in the network given the departure time functions at the origin and the initial delays by providing the departure time functions for all the other nodes in the network. The solution can be obtained by first solving problem (2), which provides the agent delay function at each node. Practically, problem (2) easier to directly solve than problem (1), because it corresponds to an explicit automaton that is easy to implement for numerical simulations.

Given a discretization time step Δt and the initial conditions $\delta_v(0)$, algorithm 1 gives a numerical solution to the discretized problem (2). The algorithm numerically integrates the ordinary differential equation (ODE) given in equation (84) over time to obtain the solution. The algorithm relies on the fact that each discretized time step is an elementary segment, because the path flows and capacities are assumed to be constant (discrete approximation) during each time step.

Algorithm 1 Calculate approximate solution of problem (2)

```
solveDelays(sourceFlow: $\lambda^0$, initialDelays: $\delta^0[0]$, capacities: $\mu$)
  for $l \in L_0^{\text{out}}$ do
    for $t = 1$ to $T$ do
      update($v_l^{\text{out}}, t, 1, 0$)
    end for
  end for

  update(node: $v$, timeStep: $t$, lastActiveConstraint: $\hat{\omega}$)
  if $v \notin S$ then
    $\Delta_{0,v}^0[t] = \Delta_{0,\pi_v}^0[t] + \delta_v^0[t-1]$
    for $l \in L_v$ do
      $\mu_l^0[t] = \mu_l(t + \Delta_{0,v}^0[t])$
      $c_l^0[t] = \frac{\sum_{p \in P_l} \lambda_p^0[t]}{\mu_l^0[t]}$
    end for
    $\gamma_v[t] = \arg\max_{l \in L_v^{\text{out}}} c_{v,l}(t)$
    $\Gamma_v[t] = P_{\gamma_v(t)}$
    $\omega_v[t] = \frac{\sum_{p \in \Gamma_v[t]} \lambda_p^0[t]}{\mu_l^0[t]}$
    $\delta_v^0[t] = max\,(0,\ (\omega_v - \hat{\omega}) \cdot \Delta t)$
    for $l \in L_v^{out}$ do
      if $\delta_v^0[t] > 0$ then
        update($v_l^{\text{out}},\ t,\ \omega_v$)
      else
        update($v_l^{\text{out}},\ t,\ \hat{\omega}$)
      end if
    end for
  end if
```

4.1. **Single route with multiple bottlenecks.** The first case we will study is that of a simple single path network with multiple queues due to several capacity bottlenecks, as illustrated in figure 3. This network can be modeled as a tree with a single sink, i.e. a single path. Thus, we will remove the path index from the notation in this section. Each internal node v has a unique child, thus the internal nodes can be indexed by the integers $v_0, \cdots, v_n$ and the unique path of the tree is $[v_0, v_1, \cdots, v_n, v_s]$. Moreover, as they model a succession of queues on the same road, we can assume that the capacity of each link (v_i, v_{i+1}) is constant and equal to capacity of the corresponding road segment $\forall v, \mu_{v_i,v_{i+1}} = \mu$. From Theorem 5, we know that the evolution of delay is given by

$$\left.\frac{d\Delta_p^0}{dt}\right|_t = \begin{cases} \dfrac{\sum_{p'\in\tilde{\Gamma}_p^0(t)} \lambda_{p'}^0(t)}{\tilde{Q}_p^0(t)} - 1 & \text{if p has an active queue} \\ 0 & \text{otherwise} \end{cases} \tag{95}$$

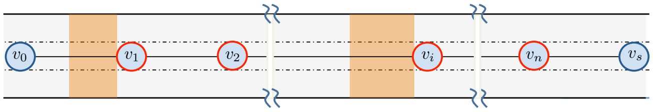

FIGURE 3. Multiple bottlenecks on a road.

Since, the link with the smallest capacity will always be the last active link $\tilde{\mu} = \min(\mu_{v_i,v_{i+1}})$:

$$\left.\frac{d\Delta^0}{dt}\right|_t = \begin{cases} \dfrac{\lambda^0(t)}{\tilde{\mu}} - 1 & \text{if there is an active queue} \\ 0 & \text{otherwise} \end{cases} \tag{96}$$

Thus, the evolution of total delay is equivalent to the evolution law for one queue of capacity $\tilde{\mu}$, and the network can be simplified to a unique internal node v followed by a link of capacity $\tilde{\mu}$.

If the capacity of the links is time varying and $\bar{\mu}(t)$ is the capacity of the most constrained link that the agent entering the network at time t is subjected to,

$$\left.\frac{d\Delta^0}{dt}\right|_t = \begin{cases} \dfrac{\lambda^0(t)}{\bar{\mu}(t)} - 1 & \text{if p has an active queue} \\ 0 & \text{otherwise} \end{cases} \tag{97}$$

4.2. **Off-Ramp bottleneck.** The next application is to compute the the dynamics of a congested freeway off-ramp, using the off-ramp model presented by Newell [38]. This example shows the versatility of the proposed framework, since Newell's the model includes non-FIFO dynamics at the off-ramp. This is accommodated by introducing an additional node and state dependent capacities on two links. The description of the model is as follows. As seen in figure 4(a), there are two flows λ_h and λ_e that enter the network, which has a capacity of μ_h. Therefore, $\lambda_h(t) + \lambda_e(t) \leq \mu_h$. The exiting flow λ_e is restricted by a capacity constraint of μ_e at the exit. There are four possible states of queuing dynamics that can occur based on the flow values. Figure 5 illustrates the transitions between the states.

Case 1: $\lambda_e \leq \mu_e$. If $\lambda_e(t) \leq \mu_e$, no queues will form in the network and there will be no delay.

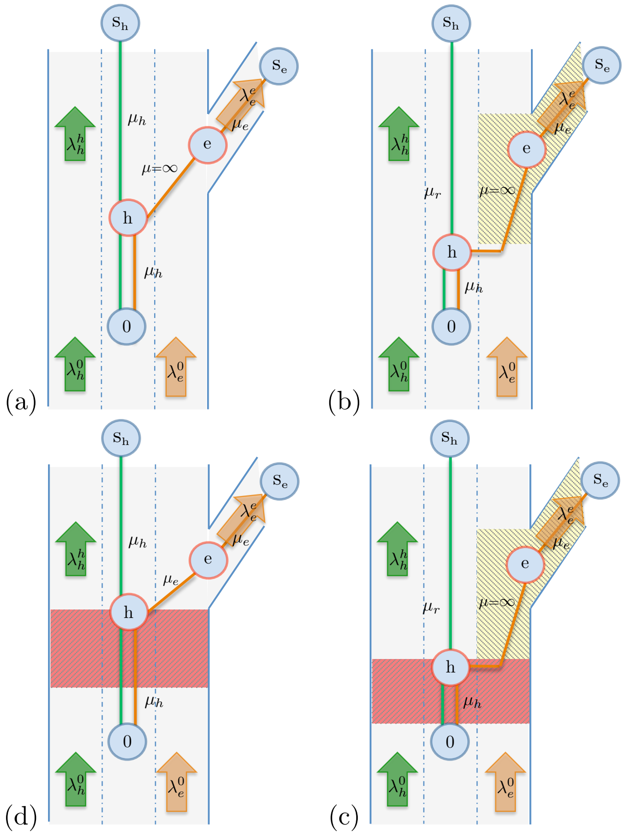

FIGURE 4. Off-Ramp model with its four states - (a) state $\emptyset$ (b) state Q_e (c) state (Q_e, Q_h) (d) state Q_h. See figure 5 for a illustration of the state transitions.

Case 2: $\lambda_e > \mu_e$ and $\lambda_h \leq \mu_r$. If $\lambda_e(t) > \mu_e$, an exit queue will start forming at the entrance to the exit as seen in figure 4(b), which will then restrict the capacity of the freeway from μ_h to μ_r.
Case 3: $\lambda_e > \mu_e$, $\lambda_h > \mu_r$ and $\frac{\mu_r}{\lambda_h} \cdot \lambda_e \geq \mu_e$. If the freeway flow $\lambda_h > \mu_r$, then a second freeway queue will start forming behind the exiting agent queue, as seen in figure 4(c), since the freeway demand is greater than the new reduced freeway capacity μ_r. This second freeway queue will contain both freeway and exiting agents and therefore the flow exiting the queue will be subject to the *first-in-first-out* (FIFO) condition. As a result, since the freeway flow λ_h is restricted to a rate of μ_r, the exiting agent flow at the freeway queue will be restricted to $\lambda'_e = \frac{\mu_r}{\lambda_h} \cdot \lambda_e$.
Case 4: $\lambda_e > \mu_e$, $\lambda_h > \mu_r$ and $\frac{\mu_r}{\lambda_h} \cdot \lambda_e < \mu_e$. Now, if $\lambda'_e < \mu_e$, then the off-ramp queue will start decreasing since the flow is less than the capacity and the queue will disappear. Thus, in this case, an off-ramp bottleneck created a second bottleneck that in turn removed the off-ramp bottleneck, which is an unstable equilibrium. Therefore, as explained in [38], there will be a single queue of both freeway and exiting agents that occurs at the off-ramp, as seen in figure 4(d), and the freeway flow through the bottleneck will be $\lambda_h^{out} = \frac{\mu_e}{\lambda_e} \cdot \lambda_h$ according to the FIFO condition.

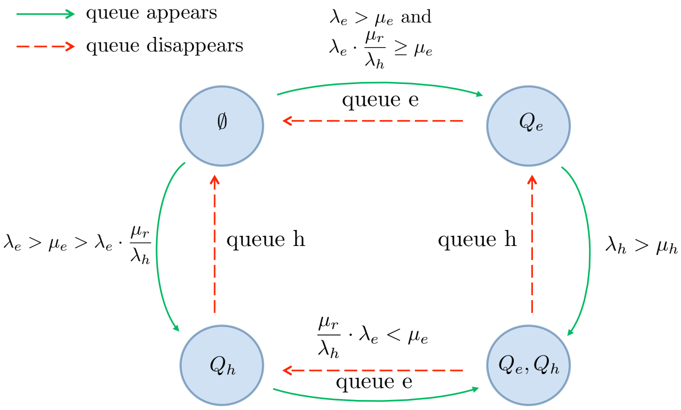
queue appears
queue disappears
$\lambda_e > \mu_e$ and
$\lambda_e \cdot \frac{\mu_r}{\lambda_h} \geq \mu_e$
queue e
$\emptyset$
Q_e
$\lambda_e > \mu_e > \lambda_e \cdot \frac{\mu_r}{\lambda_h}$
queue h
queue h
$\lambda_h > \mu_h$
$\frac{\mu_r}{\lambda_h} \cdot \lambda_e < \mu_e$
Q_h
Q_e, Q_h
queue e

FIGURE 5. State transitions in the off-ramp model. The four states $\emptyset$, Q_e, (Q_e, Q_h) and Q_h correspond respectively to the cases (a), (b), (c) and (d) from figure 4.

The uniqueness and existing properties hold even with the state dependent capacities, since the flows are assumed to be piecewise polynomial and therefore lead to a finite number of state transitions. This implies that the link capacities are piecewise constant. Therefore, we can solve for the delays in this network using algorithm 1. Furthermore, this subnetwork can be part of a larger network over which we wish to compute the system delays.

Figure 6 shows the flow and delay profiles for a numerical example of the off ramp network with the following link capacities: $\mu_E = 5, \mu_H = 30$ and $\mu = 45$. We can observe the following state transitions during the simulated time window.

- At `t=92` Appearance of exiting agent queue.
- At `t=121` Appearance of freeway queue.
- At `t=222` Disappearance of exiting agent queue.
- At `t=372` Disappearance of freeway queue.

One interesting observation is that freeway congestion caused by the exiting agent bottleneck persists well beyond the time at which the exiting agent queue disappears.

5. **Conclusion.**

This article presented a mathematical framework for modeling traffic flow through a network with a single source and multiple sinks. The model satisfies the standard laws of flow dynamics and is shown to lead to a well-posed dynamics problem with an unique solution. The main benefit of this framework is the ability to analytically prescribe the delays at each junction as a function of the boundary flows at any other upstream junction and the delay over any sub-path with respect to the boundary flow at the source node of the sub-path. This is a critical requirement when solving control and optimization problems over the network, since solving an optimization problem over simulation models is generally intractable in terms of computational complexity. The versatility of computing the delays as a function of the inflow at any point in the network is achieved though a mathematical framework for time mapping the delays. An algorithm for computing a discrete approximation of the system numerically is also provided. The application of the framework is illustrated using two examples, a single path consisting of multiple

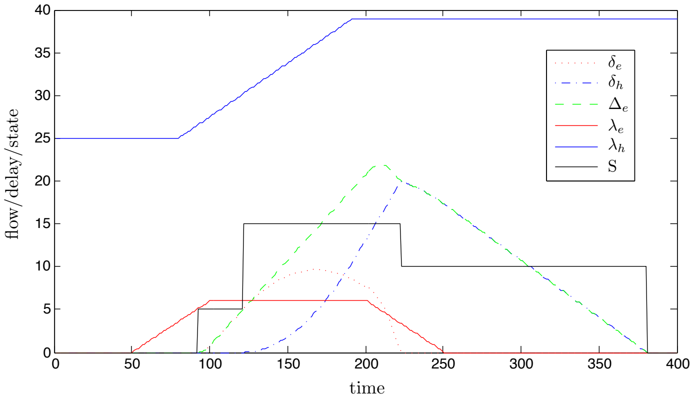

FIGURE 6. Simulation of states and delays (δ_E, δ_H) as functions of time t, given the incoming flows at the off ramp, and road parameters: $\mu_E = 5, \mu_H = 30$ and $\mu = 45$

bottlenecks and a diverge junction with complex junction dynamics. The main limitation of this framework is the requirement that the network only have a single source. The time mapping framework presented can however be generalized to any non-cyclic (tree) network. Thus, the next step would be to introduce merging dynamics into the model to obtain a more general network model.

REFERENCES

[1] A. Adas, Traffic models in broadband networks, *Communications Magazine, IEEE,* **35** (1997), 82–89.

[2] V. Astarita, A continuous time link model for dynamic network loading based on travel time function, In *13th International Symposium on Transportation and Traffic Theory,* (1996), 79–102, Lyon, France.

[3] X. J. Ban, J.-S. Pang, H. X. Liu and R. Ma, Continuous-time point-queue models in dynamic network loading, *Transportation Research Part B: Methodological,* **46** (2012), 360–380.

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

[5] A. Bressan and K. Han, Existence of optima and equlibria for traffic flow on networks, *Networks & Heterogeneous Media,* **8** (2013), 627–648.

[6] C.-S. Chang, *Performance Guarantees in Communication Networks*, Springer, 2000.

[7] C. F. Daganzo, The cell transmission model, Part II network traffic, *Transportation Research Part B: Methodological,* **29** (1995), 79–93.

[8] C. F. Daganzo, A continuum theory of traffic dynamics for freeways with special lanes, *Transportation Research Part B: Methodological*, **31** (1997), 83–102.

[9] C. F. Daganzo, W.-H. Lin and J. M. Del Castillo, A simple physical principle for the simulation of freeways with special lanes and priority vehicles, *Transportation Research Part B: Methodological*, **31** (1997), 103–125.

[10] H. M. Edwards, *Advanced Calculus: A Differential Forms Approach*, Springer Science & Business Media, New York, 2014.

[11] T. L. Friesz, K. Han, P. A. Neto, A. Meimand and T. Yao, Dynamic user equilibrium based on a hydrodynamic model, *Transportation Research Part B: Methodological,* **47** (2013), 102–126.

[12] V. S. Frost and B. Melamed, Traffic modeling for telecommunications networks, *Communications Magazine, IEEE,* **32** (1994), 70–81.

[13] K. Han, B. Piccoli, T. L. Friesz and T. Yao, A continuous-time link-based kinematic wave model for dynamic traffic networks, arXiv preprint, `arXiv:1208.5141`, 2012.

[14] K. Han, T. L. Friesz and T. Yao, Existence of simultaneous route and departure choice dynamic user equilibrium, *Transportation Research Part B: Methodological,* **53** (2013a), 17–30.

[15] K. Han, T. L. Friesz and T. Yao, A partial differential equation formulation of Vickrey's bottleneck model, part i: Methodology and theoretical analysis, *Transportation Research Part B: Methodological,* **49** (2013b), 55–74.

[16] K. Han, T. L. Friesz and T. Yao, A partial differential equation formulation of Vickrey's bottleneck model, part ii: Numerical analysis and computation, *Transportation Research Part B: Methodological,* **49** (2013c), 75–93.

[17] M. Herty, A. Klar and B. Piccoli, Existence of solutions for supply chain models based on partial differential equations, *SIAM Journal on Mathematical Analysis,* **39** (2007), 160–173.

[18] P. Hidas, Modelling vehicle interactions in microscopic simulation of merging and weaving, *Transportation Research Part C: Emerging Technologies,* **13** (2005), 37–62.

[19] H. Holden and N. H. Risebro, A mathematical model of traffic flow on a network of unidirectional roads, *SIAM J. Math. Anal.,* **26** (1995), 999–1017.

[20] Y. Hollander and R. Liu, The principles of calibrating traffic microsimulation models, *Transportation,* **35** (2008), 347–362.

[21] D.-W. Huang, Effects of ramps in the nagel-schreckenberg traffic model, *International Journal of Modern Physics C,* **13** (2002), 739–749.

[22] S. Jain, K. Fall and R. Patra, Routing in a delay tolerant network, In *Proceedings of the 2004 Conference on Applications,* Technologies, Architectures, and Protocols for Computer Communications, ACM SIGCOMM '04, **34** (2004), 145–158, New York, NY, USA.

[23] B. Jia, R. Jiang and Q.-S. Wu, Traffic behavior near an off ramp in the cellular automaton traffic model, *Phys. Rev. E,* **69** (2004), 056105.

[24] J. Lafontaine, *An Introduction to Differential Manifolds*, 2015.

[25] J. Lebacque, The godunov scheme and what it means for first order traffic flow models, In *Proceedings of the 13th International Symposium on Transportation and Traffic Theory*, (1996a), 647–678.

[26] M. J. Lighthill and G. B. Whitham, On kinematic waves I. flood movement in long rivers, *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, **229** (1955), 281–316.

[27] M. J. Lighthill and G. B. Whitham, On kinematic waves II. a theory of traffic flow on long crowded roads, *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, **229** (1955), 317–345.

[28] H. K. Lo, E. Chang and Y. C. Chan, Dynamic network traffic control, *Transportation Research Part A: Policy and Practice,* **35** (2001), 721–744.

[29] R. Ma, X. J. Ban and J.-S. Pang, Continuous-time dynamic system optimum for single-destination traffic networks with queue spillbacks, *Transportation Research Part B: Methodological,* **68** (2014), 98–122.

[30] D. K. Merchant and G. L. Nemhauserm, A model and an algorithm for the dynamic traffic assignment problems, *Transportation science*, **12** (1978), 183–199.

[31] D. K. Merchant and G. L. Nemhauser, Optimality conditions for a dynamic traffic assignment model, *Transportation Science*, **12** (1978), 200–207.

[32] I. M. Mitchell, A. M. Bayen and C. J. Tomlin, A time-dependent hamilton-jacobi formulation of reachable sets for continuous dynamic games, *IEEE Transactions on Automatic Control*, **50** (2005), 947–957.

[33] K. Nagel and M. Schreckenberg, A cellular automaton model for freeway traffic, *J. Phys. I France,* **2** (1992), 2221–2229.

[34] A. Nagurney, J. Dong and D. Zhang, A supply chain network equilibrium model, *Transportation Research Part E: Logistics and Transportation Review*, **38** (2002), 281–303.

[35] G. Newell, A simplified theory of kinematic waves in highway traffic, part I: General theory, *Transportation Research Part B: Methodological,* **27** (1993a), 281–287.

[36] G. Newell, A simplified theory of kinematic waves in highway traffic, part II: Queueing at freeway bottlenecks, *Transportation Research Part B: Methodological,* **27** (1993b), 289–303.

[37] G. Newell, A simplified theory of kinematic waves in highway traffic, part III: Multi-destination flows, *Transportation Research Part B: Methodological,* **27** (1993c), 305–313.

[38] G. F. Newell, Delays caused by a queue at a freeway exit ramp, *Transportation Research Part B: Methodological,* **33** (1999), 337–350.

[39] N. Nezamuddin and S. D. Boyles, A continuous due algorithm using the link transmission model, *Networks and Spatial Economics*, **15** (2015), 465–483.

[40] C. Osorio and G. Flötteröd, Capturing dependency among link boundaries in a stochastic dynamic network loading model, *Transportation Science,* **49** (2014), 420–431.

[41] C. Osorio, G. Flötteröd and M. Bierlaire, Dynamic network loading: A stochastic differentiable model that derives link state distributions, *Transportation Research Part B: Methodological,* **45** (2011), 1410–1423.

[42] S. Peeta and A. K. Ziliaskopoulos, Foundations of dynamic traffic assignment: The past, the present and the future, *Networks and Spatial Economics,* (2001), 233–265.

[43] J. Reilly, S. Samaranayake, M. L. D. Monache, W. Krichene, P. Gaotin and A. M. Bayen, An efficient method for coordinated ramp metering using the discrete adjoint method, *Journal of Optimization Theory and Applications,* **167** (2015), 733–760.

[44] P. I. Richards, Shock waves on the highway, *Operations research,* **4** (1956), 42–51.

[45] W. Rudin, *Principles of Mathematical Analysis - Volume 3*, McGraw-Hill New York, 1964.

[46] S. Samaranayake, A. Parmentier, Y. Xuan and A. M. Bayen, Congestion reduction at an off-ramp via incentives for demand shift, In *Proceedings of the IEEE European Control Conference,* (2015), 3465–3471.

[47] S. Samaranayake, J. Reilly, W. Kirchenbe. M. L. D. Monache, P. Gaotin and A. M. Bayen, System optimal dynamic traffic assignment with partial compliance (SO-DTA-PC), In *Proceedings of the IEEE American Control Conference,* (2015), 663–670.

[48] I. Simaiakis and H. Balakrishnan, Queuing models of airport departure processes for emissions reduction, In *Proceedings of the AIAA Guidance, Navigation and Control Conference and Exhibit*, (2009).

[49] H. S. Stone, Multiprocessor scheduling with the aid of network flow algorithms, *IEEE Transactions on Software Engineering,* **SE-3** (1997), 85–93.

[50] M. H. Stone, The generalized weierstrass approximation theorem, *Mathematics Magazine,* **21** (1948), 237–254.

[51] C. M. Tampere, R. Corthout, D. Cattrysse and L. H. Immers, A generic class of first order node models for dynamic macroscopic simulation of traffic flows, *Transportation Research Part B: Methodological,* **45** (2011), 289–309.

[52] W. S. Vickrey, Congestion theory and transport investment, *The American Economic Review,* **59** (1969), 251–260.

[53] D. Work, S. Blandin, O.-P. Tossavainen, B. Piccoli and A. M. Bayen, A traffic model for velocity data assimilation, *AMRX Applied Mathematics Research eXpress,* **1** (2010), 1–35.

[54] I. Yperman, *The Link Transmission Model for Dynamic Network Loading*, Ph.D. Thesis, Katholieke Universiteit Leuven, Belgium, 2007.

A. **Total path delay.** In some applications, it is also important to be able to computing the total delay experienced by an agents that takes a particular path. In this section, we provide analytical expressions for the total delay along a path.

Definition A.1. *Total delay of a path p*
We define the total delay Δ_p^0 encountered on a path p at time t as the total delay encountered by agent on path p that enters on the network at t throughout its entire path to the sink node.

$$\Delta_p^0\left(t\right) = [D_p^{v_{p_N}}]^{-1}\left(D_p^0\left(t\right)\right) - t \tag{98}$$

where v_{p_N} is the last non-sink node on path p. We define the time mapped total delay Δ_p^j as the total delay in path p as seen by an agent that is at node j at time t.

$$\Delta_p^j = \mathbf{T}^{j,0}\left(\Delta_p^0\right) \tag{99}$$

Proposition 14. *Total delay Δ_p^j as a function of queue delay δ*
The time mapped total delay Δ_p^j encountered on a path is equal to the sum of delay encountered by the agent on its path.

$$\Delta_p^j\left(t^j\right) = \sum_{v\in V_p\setminus(\{0\}\cup S)} \delta_v^j\left(t^j\right) \tag{100}$$

where t^j is the time that the agent is at node j.

Proof. Let $t^i = T^{i,j}\left(t^j\right)$. We obtain the result as follows using the definition of delay and a series of time mappings.

$$\begin{aligned}
LHS &= \Delta_p^j\left(t^j\right) \\
&= \mathbf{T}^{j,0}\left(\Delta_p^0\left(t^j\right)\right) \\
&= \Delta_p^0\left(T^{0,j}\left(t^j\right)\right) \\
&= \Delta_p^0\left(t^0\right) \\
&= [D_p^{v_{p_N}}]^{-1}\left(D_p^0\left(t^0\right)\right) - t^0
\end{aligned}$$

$$\begin{aligned}
RHS &= \sum_{v\in V_p\setminus(\{0\}\cup S)} \delta_v^j\left(t^j\right) \quad = \sum_{v\in V_p\setminus(\{0\}\cup S)} \delta_v^{\pi_v}\left(t^{\pi_v}\right) \\
&= \sum_{v\in V_p\setminus(\{0\}\cup S)} [D_p^v]^{-1}\left(D_p^{\pi_v}\left(t^{\pi_v}\right)\right) - t^{\pi_v} \\
&= \sum_{v\in V_p\setminus(\{0\}\cup S)} [D_p^v]^{-1}\left(D_p^{\pi_v}\left(t^{\pi_v}\right)\right) - [D_p^{\pi_v}]^{-1}\left(D_p^{\pi_v}\left(t^{\pi_v}\right)\right) \\
&= \sum_{v\in V_p\setminus(\{0\}\cup S)} [D_p^v]^{-1}\left(D_p^0\left(t^0\right)\right) - [D_p^{\pi_v}]^{-1}\left(D_p^0\left(t^0\right)\right) \\
&= [D_p^{v_{p_N}}]^{-1}\left(D_p^0\left(t^0\right)\right) - [D_p^0]^{-1}\left(D_p^0\left(t^0\right)\right) \\
&= [D_p^{v_{p_N}}]^{-1}\left(D_p^0\left(t^0\right)\right) - t^0
\end{aligned}$$

□

Definition A.2. *Active link of the last active queue of a path p at time t $\left(a_p\left(t\right)\right)$*
Let p be a path and t be the time that an agent departs node j. We define the last active queue of the path p time mapped to passing node j at time t as

$$a_p^j\left(t\right) = \max_{\succ}\left\{v \in V_p | \eta_v^j\left(t\right) = 1\right\} \tag{101}$$

For notational convenience we also define the following:

$$\tilde{\gamma}_p^j\,(t) = \gamma^j_{a_p^j(t)}\,(t) \tag{102}$$

$$\tilde{\Gamma}_p^j\,(t) = \Gamma^j_{a_p^j(t)}\,(t) \tag{103}$$

$$\tilde{Q}_p^j\,(t) = \mu^j_{\tilde{\gamma}_p^j(t)}\,(t) \tag{104}$$

Theorem 5. ***Evolution law for total delay*** Δ_p^0
Let p be a path, t be a time. The evolution law for total delay at time t is

$$\left.\frac{d\Delta_p^0}{dt}\right|_t = \begin{cases} \dfrac{\sum_{p'\in\tilde{\Gamma}_p^0(t)} \lambda_{p'}^0\,(t)}{\tilde{Q}_p^0\,(t)} - 1 & \textit{if } p \textit{ has an active queue} \\ 0 & \textit{otherwise} \end{cases} \tag{105}$$

Proof. Taking the derivative of equation (100) for $j = 0$, we obtain

$$\left.\frac{d\Delta_p^0}{dt}\right|_t = \sum_{v\in V_p\setminus(S\cup\{0\})} \left.\frac{d\delta_v^0}{dt}\right|_t \tag{106}$$

$$= \sum_{\{v|v\in V_p\setminus(S\cup\{0\}),\gamma_v^0(t)=1\}} \left.\frac{d\delta_v^0}{dt}\right|_t \tag{107}$$

Note that $\gamma_v^0\,(t) = 1$ implies node v is active when the source flow at time t reaches node v. From Theorem 2 with $j = 0$ we have,

$$\left.\frac{d\delta_v^0}{dt}\right|_t = \begin{cases} \dfrac{\sum_{p\in\Gamma_v^0(t)} \lambda_p^0\,(t)}{Q_v^0\,(t)} - 1 \\ \text{if } v \text{ is the first active queue} \in p \\ \dfrac{\sum_{p\in\Gamma_v^0(t)} \lambda_p^0\,(t)}{Q_v^0\,(t)} - \dfrac{\sum_{p\in\hat{\Gamma}_v^0(t)} \lambda_p^0\,(t)}{\hat{Q}_v^0\,(t)} \text{ otherwise} \end{cases} \tag{108}$$

Plugging this into equation (107) gives a telescopic series, since it only considers the active nodes of the path and $\tilde{Q}_p^0\,(t)$ gives the capacity of the last active link of path p. Thus, we obtain

$$\left.\frac{d\Delta_p^0}{dt}\right|_t = \frac{\sum_{p'\in\tilde{\Gamma}_p^0(t)} \lambda_{p'}^0\,(t)}{\tilde{Q}_p^0\,(t)} - 1 \tag{109}$$

If p does not contain an active queue there is no queuing in the path, which means there is no change in the queue length and therefore no change in the delay. □

Remark 9. Note that this theorem can be extended to any subpath $p_{ij} \in p$ such that

$$\left.\frac{d\Delta_{p_{i,j}}^i}{dt}\right|_t = \frac{\sum_{p'\in\tilde{\Gamma}_{p_{ij}}^i(t)} \lambda_{p'}^i\,(t)}{\tilde{Q}_{p_{ij}}^i\,(t)} - 1 \tag{110}$$

Received June 2016; revised December 2016.

E-mail address: samitha@cornell.edu
E-mail address: axel.parmentier@enpc.fr
E-mail address: xuan.yiguang@path.berkeley.edu
E-mail address: bayen@berkeley.edu