A mathematical framework for delay analysis in single source networks

Axel Parmentier, Samitha Samaranayake, Yiguang (Ethan) Xuan, Alexandre Bayen

***Abstract*— This article presents a mathematical framework for modeling heterogeneous flow networks with a single source and multiple sinks. The traffic is differentiated by its destination (i.e. Lagrangian flow) and different flow groups are assumed to satisfy the first-in-first-out (FIFO) condition at each junction. We show that our 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 one of the contributions 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 a diverge junction with complex junction dynamics.**

I. Introduction

Modeling and analysing the dynamics of network flows is an important problem that has applications in many different areas such as transportation planning [3], [7], [14], air traffic control [10], [16], communication networks [1], [2], [4], [6], processor scheduling [17] and supply chain optimization [11]. 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 [19], dynamic traffic assignment or demand response assessment [9], traffic signal control [8], ramp-metering control [5] and incident rerouting [15]. This article focuses on modeling heterogeneous (multi-path) physical flows through a network with a single source and multiple sinks with the specific objective of expressing the delays at each node of the network as a function of the boundary flows at the source. This can be a critical requirement when solving control and optimization problems over a network in cases where the flow entering the network is one of the direct or indirect control parameters of the system. For example, when trying to eliminate congestion at a critical node of the network by manipulating the boundary flows. We present our framework in the context of physical flow networks and particularly freeway transportation networks, which have the following physical requirements, but our results can be applied to any network that satisfies these 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.

While there is a vast literature on network flow propagation, particularly for various packet networks, a large majority of these dynamics models violate the FIFO and no holding requirements listed above, which are essential requirements in phyisical flow networks. Many models proposed for transportation network flows do infact satisfy these physical requirements [3], [7], but none of these models 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 queuing in the network is contained at each junction node, i.e. spill-back to the previous junction if occurs is ignored[1]. We show that our 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 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. 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 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 a diverge junction with complex junction dynamics.

A. Parmentier is with the Centre d'Enseignement et de Recherche en Mathmatiques et Calcul Scientifique at cole des Ponts ParisTech e-mail: axel.parmentier@enpc.fr

S. Samaranayake is with the Department of Civil and Environmental Engineering at the UC Berkeley e-mail: samitha@berkeley.edu

Y. Xuan is with California Partners for Advanced Transportation Technology at UC Berkeley e-mail: xuan.yiguang@path.berkeley.edu

A. Bayen is with the Department of Electrical Engineering and Computer Science, and Civil and Environmental Engineering at UC Berkeley e-mail: bayen@berkeley.edu

[1]Spill back to the previous junction can be observed and flagged when it occurs. The primary goal of this model is for being used in optimization problems where (in most cases) a good solution will eliminate long spill backs.

II. A POINT QUEUE MODEL FOR NETWORK FLOW

A. 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.

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. 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 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 $l \in p$.

B. 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 1. 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 monotonically increasing: 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 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[2]. Cumulative curves are required to be 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

$$\lambda_p^v = \frac{dD_p^v}{dt} \tag{1}$$

Remark 2. *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:

$$D_p^{\pi_v} = A_p^v \qquad \forall l = (\pi_v, v) \in L, p \in P_v \tag{2}$$

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

$$\frac{dA_p^v}{dt} = \frac{dD_p^{\pi_v}}{dt} = \lambda_p^{\pi_v} \tag{3}$$

$$\frac{dD_p^v}{dt} = \lambda_p^v. \tag{4}$$

C. Queuing and diverge model

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.

$$\sum_{p \in P_l} \lambda_p^{v_l^{\text{in}}}(t) \leq \mu_l(t) \qquad \forall t, l \in L \tag{5}$$

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

Definition 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

$$n_{v,p}(t) = D_p^v(t) - A_p^v(t) \tag{6}$$

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

$$n_v(t) = \sum_{p \in P_v} n_{v,p}(t) \tag{7}$$

[2]We could relax the assumption that the cumulative curves are strictly increasing and allow for monotonically increasing curves. However, this results in the time mapping function $T^{(,\pi_v)}v$ introduced in section III-B 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.

Remark 3. 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 3. *Delay in queue* v. We define $\delta_{v,p}(t)$ as the delay encountered in queue v by the agent which entered the queue at time t.

$$\begin{aligned} \delta_{v,p}(t) \qquad &= [D_p^v]^{-1}\left(A_p^v(t)\right) - t \\ &= [D_p^v]^{-1}\left(D_p^{\pi_v}(t)\right) - t \end{aligned} \tag{8}$$

As D_p^v is continuous, piecewise C^1, and strictly increasing, its inverse is continuous, piecewise C^1 and strictly increasing. Thus, as $D_p^{\pi_v}$ is also continuous, piecewise C^1 and strictly increasing, the function $[D_p^v]^{-1} \circ D_p^{\pi_v}$ is continuous and piecewise C^1, and delay $\delta_{v,g}$ is continuous and piecewise C^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.

$$\delta_v(t) = \delta_{v,p}(t) = [D_p^v]^{-1}\left(D_p^{\pi_v}(t)\right) - t \quad \forall t, \forall p \in P_v \tag{9}$$

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

$$t_1 < t_2 \Leftrightarrow [D_{p_1}^v]^{-1}\left(D_{p_1}^{\pi_v}(t_1)\right) < [D_{p_2}^v]^{-1}\left(D_{p_2}^{\pi_v}(t_2)\right) \tag{10}$$

The FIFO condition also implies the conservation of the ratio flows across paths through the junction.

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

$$\frac{\lambda_{p_1}^v(t+\delta_v(t))}{\lambda_{p_2}^v(t+\delta_v(t))} = \frac{\lambda_{p_1}^{\pi_v}(t)}{\lambda_{p_2}^{\pi_v}(t)}, \quad \forall t \in (t_{init}, t_{\text{final}}] \tag{11}$$

See appendix I in [13] for the proof.

Definition 4. *Queue state* η_v *- state transitions.* We define queue state as the boolean valued function $\eta_v(t)$:

$$\eta_v(t) = \begin{cases} 1 & \text{if } \delta_v(t) > 0 \\ 0 & \text{otherwise} \end{cases} \tag{12}$$

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

$$\exists \epsilon > 0 \text{ s.t. } \forall \theta \in [-\epsilon, \epsilon], \quad \eta_v(t-\theta) = 1 - \eta_v(t+\theta) \tag{13}$$

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

Definition 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[3].

$$c_{v,l}(t) = \frac{\sum_{p \in P_l} \lambda_p^{\pi_v}(t)}{\mu_l(t+\delta_v(t))} \tag{14}$$

[3]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.

Definition 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 [4] in L_v^{out}:

$$\gamma_v(t) \in \underset{l \in L_v^{\text{out}}}{\arg\max}\, c_{v,l}(t) \tag{15}$$

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

$$\Gamma_v(t) = P_{\gamma_v(t)} \tag{16}$$

$\Gamma_v \subset P_v$ because $P_v = \cup_{l \in L_v^{\text{out}}} P_l$.

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.

$$\delta_v(t) > 0 \Rightarrow \sum_{p \in \Gamma_v(t)} \lambda_p^v(t+\delta_v(t)) = \mu_{\gamma_v(t)}(t+\delta_v(t)) \tag{17}$$

Definition 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.

Definition 8. *Initial times for each non-source node.* Given a set of initial delays at each node $\delta_v(t_{\text{initial}}) \geq 0, \forall v \in V \backslash (S \cup \{v_0\})$ 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:

$$\begin{cases} t_{0,\text{initial}} &= t_{\text{initial}} \qquad \text{for node } v_0 \\ t_{v,\text{initial}} &= t_{\pi_v,\text{initial}} + \delta_v(t_{\pi_v,\text{initial}}) \end{cases} \tag{18}$$

D. 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 $\mu_l(t), \forall l \in L$, $t \in [t_{\text{initial}}, t_{\text{final}}]$, 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 $\delta_v(t_{\text{initial}}) \geq 0$, $\forall v \in V \backslash (S \cup \{v_0\})$
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. 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 $\lambda_p^0(t)$ are piecewise polynomial,
2) link capacities μ_l are piecewise constant over time.

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

Note that neither of the assumptions of the theorem are restrictive in a practical sense[5].

III. A SOLUTION BASED ON TIME MAPPING

This section builds a constructive proof of theorem 1. Throughout sections III-A-III-C, we require that the flows at the origin are acceptable departure curves as defined in definition 1 and that the outflows at each node satisfy the model requirements (i.e. result in feasible flows as defined in definition 7).

A. 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,

$$\left.\frac{d\delta_v}{dt}\right|_t = \frac{\sum_{p\in\Gamma_v(t)} \lambda_p^{\pi_v}(t)}{\mu_{\gamma_v(t)}(t+\delta_v(t))} - 1 \qquad (19)$$

See appendix II in [13] for the proof.

B. 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. Furthermore, 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 9. Node time mapping function T^{v,π_v}. We define the time mapping function T^{v,π_v} by

$$T^{v,\pi_v} : t \mapsto t+\delta_v(t) \qquad (20)$$

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 (20), 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

$$\left.\frac{dT^{v,\pi_v}}{dt}\right|_t = \frac{\sum_{p\in\Gamma_v(t)} \lambda_p^{\pi_v}(t)}{\mu_l(t+\delta_v(t))} > 0 \qquad (21)$$

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. See appendix III in [13] for proof.

Thus T^{v,π_v} is invertible and its inverse is an increasing function[6].

Definition 10. 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}

$$T^{\pi_v,v} \circ T^{v,\pi_v} = \mathbb{1} \text{ and } T^{v,\pi_v} \circ T^{\pi_v,v} = \mathbb{1} \qquad (22)$$

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 11. 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

$$T^{v_0,v_n} = T^{v_0,v_1} \circ T^{v_1,v_2} \circ \cdots \circ T^{v_{\pi_n},v_n} \qquad (23)$$

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

$$T^{v_n,v_0} = T^{v_n,v_{\pi n}} \circ \cdots \circ T^{v_2,v_1} \circ T^{v_1,v_0} \qquad (24)$$

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 1. We can now define the time mapping function between any arbitrary pair of nodes.

Definition 12. Time mapping function between two arbitrary nodes. We define the time mapping function $T^{i,j}$ between

[5] 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 [18]) 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.

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

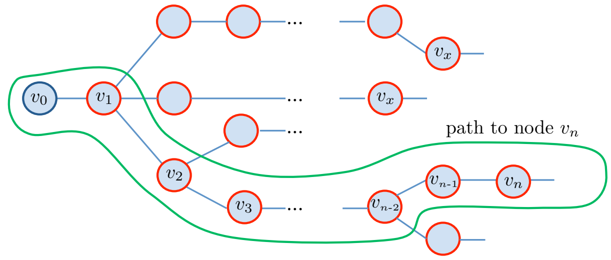

Fig. 1: Time mapping nodes

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 1),

$$T^{i,j} = \begin{cases} T^{i,i+1} \circ T^{i+1,i+2} \circ \ldots \circ T^{j-2,j-1} \circ T^{j-1,j} & \text{if } i \prec j \\ T^{i,i-1} \circ T^{i-1,i-2} \circ \ldots \circ T^{j+2,j+1} \circ T^{j+1,j} & \text{if } i \succ j \end{cases} \tag{25}$$

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 1),

$$T^{i,j} = T^{i,v_0} \circ T^{v_0,j} \tag{26}$$

Let x_j be an agent that leaves node j at t. From definition 11 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 13. *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:

$$\begin{array}{rccc} \mathbf{T}^{i,j}: & F & \rightarrow & F \\ & f & \mapsto & f \circ T^{j,i} \end{array} \tag{27}$$

We can now consider the physical interpretation of $T^{i,j}$ and derive the time mapping of each of the model quantities. This derivation is given in appendix IV of [13].

C. 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 14. *First active upstream node.* Let v be an internal node. We define the first active upstream node of v as

$$\Upsilon_v^j(t) = \max_{\succ} \left\{ u | u \prec v, \eta_u^j(t) = 1 \right\} \tag{28}$$

For notational convenience we also define the following:

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

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

$$\left. \frac{d\delta_v^j}{dt} \right|_t = \begin{cases} \dfrac{\sum_{p \in \Gamma_v^j(t)} \lambda_p^j(t)}{Q_v^j(t)} - \left. \dfrac{dT^{0,j}}{dt} \right|_t & \\ \qquad \text{if } v \text{ is the first active queue} \in p & \\ \dfrac{\sum_{p \in \Gamma_v^j(t)} \lambda_p^j(t)}{Q_v^j(t)} - \dfrac{\sum_{p \in \hat{\Gamma}_p^j(t)} \lambda_p^j(t)}{\hat{Q}_v^j(t)} & \text{otherwise} \end{cases} \tag{29}$$

See appendix V in [13] for proof.

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 15.

Definition 15. *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 $\dfrac{d\delta_v^0}{dt} = 0$.
If v is an active node or its active link γ_0^v is over capacity, then $\dfrac{d\delta_v^0}{dt}$ is given by equation (29), where the time mapping functions are redefined from delays as follows:

$$T^{j,0} = \sum_{0 \prec i \preccurlyeq j} \delta_i^0 \tag{30}$$

Proposition 4. Delay evolution does not depend on departure curves. All the time mapped quantities in equations (29) 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 9. The time mapped flows can be obtained using the time mapping function using proposition 9 in [13]. The other time mapped quantities are by definition constructed using the time mapping function as given in appendix IV of [13].

D. 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 15 for each node $v \in V \backslash v_0$ exist and is it unique?

Theorem 3. Problem (1) and problem (2) are equivalent.
See appendix VI in [13] for proof.

E. 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 $[t_{\text{initial}}, t_{\text{final}}]$, 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.

See appendix VII in [13] for proof.

In some applications, it is also important to be able to computing the total delay experienced by an agents that takes a particular path. Appendix VIII in [13] provides analytical expressions for the total delay along a path.

IV. Application

Given a discretization time step Δt and the initial conditions $\delta_v(0)$, a numerical solution to the discretized problem (2) can be computed by numerically integrating the ordinary differential equation (ODE) given in equation (29) over time to obtain the solution. Details are given in appendix IX of [13].

We use our framework to compute the the dynamics of a congested freeway off-ramp, using the model presented by Newell [12]. This example shows the versatility of our 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 2(a), there are two flows λ_h and λ_e that enter the network, which has a capacity of μ_h. Therefore, $\lambda_h(t) + \lambda_e(t) \le \mu_h$. 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 3 illustrates the transitions between the states.

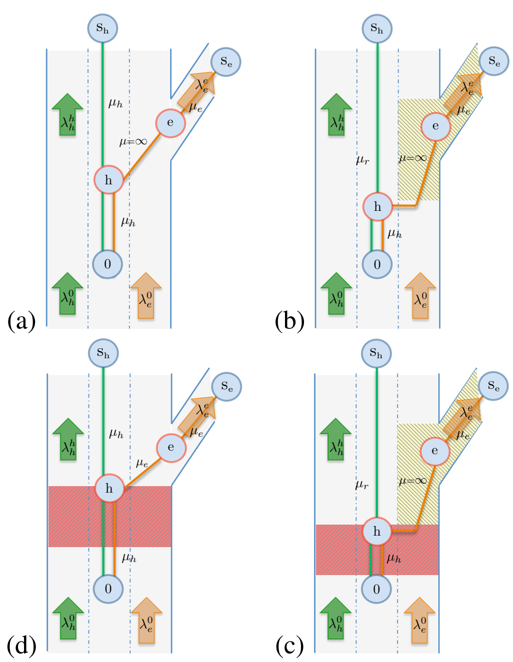

Fig. 2: Off-Ramp model with the four different queing states.

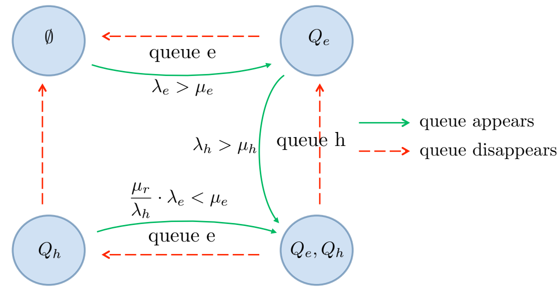

Fig. 3: State transitions in the off-ramp model. The four states $\emptyset$, Q_e, (Q_e, Q_h) and Q_h correspond respectively to the cases (a), (b), (c) and (d) from figure 2.

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 in [13]. Furthermore, this subnetwork can be part of a larger network over which we wish to compute the system delays.

Figure 4 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.

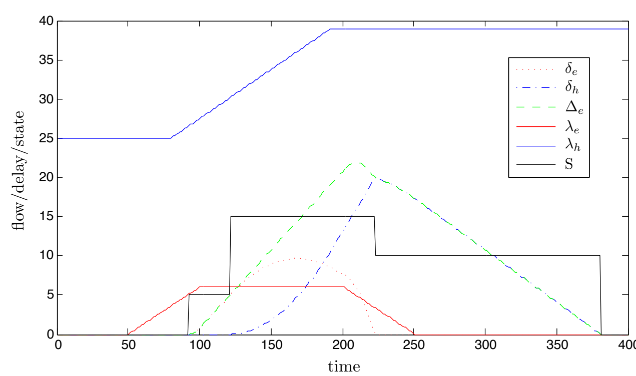

Fig. 4: Simulation of states and delays (δ_E, δ_H) as functions of time t, given the incoming flows at the off ramp, and road parameters: $\mu_E = 5, \mu_H = 30$ and $\mu = 45$.

V. 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 bottlenecks and a diverge junction with complex junction dynamics. The main limitation of this framework is the limitation to single source networks. 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.

A longer version of the article including all the proofs is available online for the reviewer's convenience at: `http://dx.doi.org/10.7922/G2RN35S6`.

REFERENCES

[1] Abdelnaser Adas. Traffic models in broadband networks. *Communications Magazine, IEEE*, 35(7):82–89, 1997.

[2] Cheng-Shang Chang. *Performance guarantees in communication networks*. Springer, 2000.

[3] Carlos F Daganzo. The Cell Transmission Model , Part II Network Traffic. 298(2):79–93, 1995.

[4] Victor S Frost and Benjamin Melamed. Traffic modeling for telecommunications networks. *Communications Magazine, IEEE*, 32(3):70–81, 1994.

[5] Gabriel Gomes and Roberto Horowitz. Optimal freeway ramp metering using the asymmetric cell transmission model. *Transportation Research Part C: Emerging Technologies*, 14(4):244–262, 2006.

[6] Sushant Jain, Kevin Fall, and Rabin Patra. Routing in a delay tolerant network. In *Proceedings of the 2004 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, SIGCOMM '04, pages 145–158, New York, NY, USA, 2004. ACM.

[7] Jean-Patrick Lebacque. The godunov scheme and what it means for first order traffic flow models. In *Internaional symposium on transportation and traffic theory*, pages 647–677, 1996.

[8] Hong K. Lo, Elbert Chang, and Yiu Cho Chan. Dynamic network traffic control. *Transportation Research Part A: Policy and Practice*, 35(8):721 – 744, 2001.

[9] Deepak K Merchant and George L Nemhauser. A Model and an Algorithm for the Dynamic Traffic Assignment Problems. *Transportation science*, 12(3):183–199, 1978.

[10] Ian M Mitchell, Alexandre M. Bayen, and Claire J Tomlin. A time-dependent hamilton-jacobi formulation of reachable sets for continuous dynamic games. *Automatic Control, IEEE Transactions on*, 50(7):947–957, 2005.

[11] Anna Nagurney, June Dong, and Ding Zhang. A supply chain network equilibrium model. *Transportation Research Part E: Logistics and Transportation Review*, 38(5):281–303, 2002.

[12] Gordon F. Newell. Delays caused by a queue at a freeway exit ramp. *Transportation Research Part B: Methodological*, 33(5):337 – 350, 1999.

[13] Axel Parmentier, Samitha Samaranayake, Yiyuang Xuan, and Alexandre M. Bayen. A mathematical framework for delay analysis in single source networks. *Technical report*, 2014. `http://dx.doi.org/10.7922/G2RN35S6`.

[14] Srinivas Peeta and Athanasios K. Ziliaskopoulos. Foundations of Dynamic Traffic Assignment : The Past , the Present and the Future. *Networks and Spatial Economics*, pages 233–265, 2001.

[15] Samitha Samaranayake, Jack Reilly, Walid Krichene Maria Laura Delle Monache, Paola Gaotin, and Alexandre M. Bayen. System Optimal Dynamic Traffic Assignment with Partial Compliance (SO-DTA-PC). *Technical report*, 2014. `http://dx.doi.org/10.7922/G23X84KV`.

[16] Ioannis Simaiakis and Hamsa Balakrishnan. Queuing models of airport departure processes for emissions reduction. In *AIAA Guidance, Navigation and Control Conference and Exhibit*, 2009.

[17] Harold S Stone. Multiprocessor scheduling with the aid of network flow algorithms. *Software Engineering, IEEE Transactions on*, (1):85–93, 1977.

[18] Marshall H Stone. The generalized weierstrass approximation theorem. *Mathematics Magazine*, 21(5):237–254, 1948.

[19] Dan Work, Sebastien Blandin, Olli-Pekka Tossavainen, Benedetto Piccoli, and Alexandre M. Bayen. A traffic model for velocity data assimilation. *AMRX Applied Mathematics Research eXpress*, 1:1–35, 2010.