A CONVEX FORMULATION OF TRAFFIC DYNAMICS ON TRANSPORTATION NETWORKS[*]

YANNING LI[†], CHRISTIAN G. CLAUDEL[‡], BENEDETTO PICCOLI[§], AND
DANIEL B. WORK[¶]

Abstract. This article proposes a numerical scheme for computing the evolution of vehicular traffic on a road network over a finite time horizon. The traffic dynamics on each link is modeled by the *Hamilton–Jacobi* (HJ) *partial differential equation* (PDE), which is an equivalent form of the *Lighthill–Whitham–Richards* PDE. The main contribution of this article is the construction of a single convex optimization program which computes the traffic flow at a junction over a finite time horizon and decouples the PDEs on connecting links. Compared to discretization schemes which require the computation of all traffic states on a time-space grid, the proposed convex optimization approach computes the boundary flows at the junction using only the initial condition on links and the boundary conditions of the network. The computed boundary flows at the junction specify the boundary condition for the HJ PDE on connecting links, which then can be separately solved using an existing semi-explicit scheme for single link HJ PDE. As demonstrated in a numerical example of ramp metering control, the proposed convex optimization approach also provides a natural framework for optimal traffic control applications.

Key words. traffic modeling, networks, junction solver, convex optimization, traffic control

AMS subject classifications. 35L50, 35R02, 35Q93, 90B20

DOI. 10.1137/16M1074795

1. Introduction. Efficient numerical schemes for solving traffic dynamics modeled by a conservation law on a transportation network are critical and the basis for traffic applications, such as real-time traffic estimation and optimal traffic control. This article proposes a numerical scheme which solves the traffic dynamics in a convex program without the discretization of the time-space domain and can be used as the framework for optimal control on transportation networks.

The road network is represented by a directed graph $\mathcal{G}(\mathcal{L}, \mathcal{V})$ consisting of links $l \in \mathcal{L}$ and vertices $v \in \mathcal{V}$. Each link l represents a road segment with spatial coordinates $x \in [a_l, b_l]$ and homogeneous physical parameters, such as the free flow speed and the capacity. Each vertex v represents a junction in the transportation network, consisting of at least one incoming link and one outgoing link. The link endpoints that are not connected to a junction are referred to as the *network boundaries* while endpoints at junctions are called *internal boundaries*.

Describing the dynamics of traffic on a network consists of two modeling components, namely a model for the traffic evolution on each link, and a model of traffic flow

---

[†]Department of Civil and Environmental Engineering, University of Illinois at Urbana Champaign, Urbana, IL 61801 (yli171@illinois.edu).

[‡]Department of Civil, Architectural and Environmental Engineering, University of Texas at Austin, Austin, TX 78712 (christian.claudel@utexas.edu).

[§]Department of Mathematical Science, Rutgers University, Camden, Camden, NJ 08102 (piccoli@camden.rutgers.edu).

[¶]Corresponding author. Department of Civil and Environmental Engineering, and Coordinated Science Laboratory, University of Illinois at Urbana Champaign, Urbana, IL 61801 (dbwork@illinois.edu).

through each junction. The standard first order model for traffic flow on a single link indexed by l is the *Lighthill–Whitham–Richards* (LWR) *partial differential equation* (PDE) [31, 38]:

$$(1) \qquad \frac{\partial \rho_l(t,x)}{\partial t} + \frac{\partial \psi_l\left(\rho_l(t,x)\right)}{\partial x} = 0,$$

which describes the evolution of the traffic density $\rho_l(t,x)$ in the time and space domain $[0, t_{max}] \times [a_l, b_l]$. In (1), $\psi_l(\cdot)$ is the flux function which describes the empirical relationship between the density and flow on each link. The initial and boundary conditions are respectively defined as $\rho_l(0,x) = \rho_{l,0}(x), x \in [a_l, b_l]$ and $\rho_l(t, a_l) = \rho_{a_l}(t)$, $\rho_l(t, b_l) = \rho_{b_l}(t), t \in [0, t_{max}]$.

There is an increasing interest in an equivalent representation of the LWR PDE (1) which is known as the *Hamilton–Jacobi* (HJ) PDE [2, 5, 6, 11, 12, 13]. By integrating (1) in space x, the following HJ PDE can be obtained:

$$(2) \qquad \frac{\partial \mathbf{M}_l(t,x)}{\partial t} - \psi_l\left(-\frac{\partial \mathbf{M}_l(t,x)}{\partial x}\right) = 0.$$

In formulation (2), the traffic state is described by a real scalar function $\mathbf{M}_l(t,x)$, known as the Moskowitz function [34, 35, 36, 37]. Intuitively, $\mathbf{M}_l(t,x)$ is a continuous analogue of a sequentially indexed vehicle ID (or cumulative vehicle count) and is related to the density by $\rho_l(t,x) = -\partial \mathbf{M}_l(t,x)/\partial x$. All vehicles (including vehicles on the road at $t = 0$) are labeled incrementally in the order they enter the link, where negative labels are assigned to vehicles initially on the link. As a convention, the continuous analogue of the vehicle ID at $(t,x) = (0, a_l)$ is set as $\mathbf{M}_l(0, a_l) = 0$. The initial and boundary conditions are respectively $\mathbf{M}_l(0,x) = \mathbf{M}_{l,0}(x)$, $\mathbf{M}_l(t, a_l) = \mathbf{M}_{a_l}(t)$, $\mathbf{M}_l(t, b_l) = \mathbf{M}_{b_l}(t)$, where the boundary Moskowitz data is related to the density data by $\mathbf{M}_{l,0}(x) = -\int_{\chi=a_l}^{x} \rho_{l,0}(\chi) \mathrm{d}\chi$, $\mathbf{M}_{a_l}(t) = \int_{\tau=0}^{t} \psi_l\left(\rho_{a_l}(\tau)\right) \mathrm{d}\tau$, and $\mathbf{M}_{b_l}(t) = \mathbf{M}_{l,0}(b_l) + \int_{\tau=0}^{t} \psi_l\left(\rho_{b_l}(t)\right) \mathrm{d}\tau$. The HJ PDE formulation (2) enables the use of variational theory [11] to compute the vehicle ID $\mathbf{M}_l$ at any point (t,x) by minimizing a functional given a concave Hamiltonian $\psi_l(\cdot)$ (i.e., a concave flux function).

The classic numerical schemes for computing the traffic evolution on a single link are based on discretization of the governing PDE. These schemes include the Godunov scheme [17] and the discrete velocities kinetic scheme [3] for the LWR PDE (1), or the dynamic programming approach [11] for the HJ PDE (2). Recently, Claudel and Bayen [5] and Mazaré et al. [33] proposed a semi-explicit HJ PDE solver based on the *Lax–Hopf* formula for a single link, which has been demonstrated to be more efficient than discretization-based schemes [6]. Built upon the semi-explicit HJ PDE solver on single link, this article develops a numerical scheme for extending the HJ PDE solver to a network.

To extend the link traffic flow model to networks, a junction model is required to describe how the traffic sent from links $s \in \mathcal{S}_v \subset \mathcal{L}$ entering the junction v is received by links $r \in \mathcal{R}_v \subset \mathcal{L}$ exiting the junction at any point in time. It is well known that conservation of vehicles across the junction, i.e., $\sum_{s \in \mathcal{S}_v} q_s(t, b_s) = \sum_{r \in \mathcal{R}_v} q_r(t, a_r)$, is insufficient to uniquely define the flows at the junction. To address this issue, a variety of junction models [10, 15, 18, 19, 20, 22, 23, 24, 27] have been proposed to define a unique internal boundary flow solution using additional rules governing the distribution or priority of the flows. Compared to the merge junction, for which relatively few models have been proposed, the diverge junctions have led to a number

of modeling efforts. The diverge models can be classified as *first-in-first-out* (FIFO) and non-FIFO models. The FIFO model is directly applicable to single-lane roadways while the non-FIFO model can be applied to multilane scenarios. In this article, we adopt a merge junction model [15, 18] and propose a new (FIFO) diverge junction model which allows rerouting behaviors while maintaining consistency (or equivalently invariance [26]). Moreover, both junction models allow the junction flows to be computed pointwise in time as a convex optimization program, which is an important feature used in the proposed numerical scheme. It should be noted that the computed boundary flow values can be posed as the strong boundary condition to the PDEs which is commonly discussed using the concept of Riemann solver [15].

The difficulty of solving the LWR PDE or HJ PDE on the network comes from the coupling nature of the PDEs at the junction. Given the initial condition on each link and the network boundary conditions, solving the HJ PDE (2) on each link requires the internal boundary condition for the entire time domain, which is unknown unless the junction model is solved. On the other hand, solving the junction model at any point in time requires the knowledge of the current local traffic condition on connecting links which comes from the solution of the governing HJ PDEs. We refer the reader to [15, 21] on the integration of junction models in the network PDEs models, as well as the discussion on the well-posedness of the LWR PDE and HJ PDE on a network.

To approximate solutions to the models on networks, several of the numerical schemes for solving the LWR PDE or HJ PDE on a single link have also been extended to the network with an additional treatment at the junction. Like the link schemes, the network schemes discretize each link into cells and time into steps. At the junction, the internal boundary flows are computed using the traffic density for the LWR PDE [10] or the cumulative number of vehicles for the HJ PDE [8] at the previous time step in the boundary cells of connecting links. After the boundary flow at the next step is obtained, the single step evolution of the traffic state on each link is computed using the link update scheme, such as the Godunov scheme [17].

Similarly, the single link semi-explicit HJ PDE solver [5, 6] can also be extended to the network using a sequential update scheme. Suppose a time grid (not necessarily uniform) with j_{max} intervals is provided at a junction v. At each time interval j, a junction solver can be formulated as a *convex program* (CP) $\mathcal{CP}_j$ as shown in section 3.2, which computes the optimal internal boundary flow with respect to the distribution or priority parameters. However, the construction of $\mathcal{CP}_j$ relies on the optimal internal boundary flow solutions up to interval $j-1$. Consequently, computing the internal boundary flows over the entire time horizon requires solving the sequence of convex programs $\mathcal{CP}_j \ \forall j \in \mathcal{J}$ consecutively.

Alternatively, this article develops a numerical scheme which reformulates the sequence of convex programs $\mathcal{CP}_j \ \forall j \in \{1, 2, \ldots, j_{max}\}$ as an equivalent single convex program. We show the dependency of $\mathcal{CP}_j$ on the optimal solutions up to interval $j - 1$ can be relaxed if the objective function of the equivalent single convex program is properly constructed. The constraint set of the single convex program over the entire time horizon is simply the union of the constraint sets of the sequence of convex programs with the optimal internal boundary flow solutions at each interval substituted by the corresponding decision variables. Then, the objective function is designed to guarantee the equivalence of the single convex program to the sequence of CPs $\mathcal{CP}_j \ \forall j \in \{1, 2, \ldots, j_{max}\}$. Finally, the obtained internal boundary flows are used as the boundary conditions to solve the corresponding HJ PDE on each link using the semi-explicit single link HJ PDE solver [5, 6].

The main contribution of this article is the development of the single convex

program scheme for computing the internal boundary flows at a merge or diverge junction. Compared to the discretization based methods [8, 10, 17], the proposed convex optimization scheme does not require discretization of the time-space domain except at the initial time, and at the link spatial boundaries. Moreover, it provides a natural framework for optimal traffic control applications as demonstrated in an example.

Note that there are other related approaches that also do not require discretization of the time-space domain, including our earlier result on optimal traffic control on networks [29] and a recent continuous-time solver of traffic dynamics on the network [18]. Our earlier work [29] investigates control of the HJ PDE on a network and assumes all junctions are fully signalized by traffic actuators. Therefore, it does not require a model of the traffic dynamics at the junction and, consequently, it cannot be used to solve the HJ PDE on a network when the junction dynamics are prescribed. The continuous-time numerical solver [18] for computing the evolution of traffic dynamics on a network uses a link-based kinematic wave model for the link and a mixed integer optimization program for solving the junction problem. In contrast, this article formulates a single convex program to solve the HJ PDE on the network over the entire time horizon.

The remainder of the article is organized as follows. Section 2 reviews the semi-explicit HJ PDE solver on a single link, which enables explicit formulation of the upper bound of the internal boundary flows that can be sent or received on each link at any time. These upper bounds define the feasible set of the convex program for computing the internal boundary flows. In section 3, we first describe the behavioral models of the merge and the diverge junction used in this article. Then we show at any time interval, the selected junction models can be posed as a convex program. Section 3.3 presents our main contribution, where we formulate a single convex program for solving the junction model over the entire time horizon. Finally in section 4, an on-ramp metering controller which improves the safety at a work zone by alleviating congestion is proposed to demonstrate the potential of the framework.

2. Sending and receiving boundary flows on a single link. This section first reviews the semi-explicit HJ PDE solver on a single link [5, 6]. Given the initial condition, the upstream and downstream boundary conditions, the HJ PDE modeling the traffic dynamics on a single link can be semi-explicitly solved. Based on this semi-explicit HJ PDE framework, we then show that if the upstream or downstream boundary condition is unknown, then the upper bound for the boundary flow can be obtained, which denotes the maximum traffic flow that can be sent or received on a link. The obtained bounds are equivalent to the maximum supply and demand [25], but it can be computed without discretizing the time and space domain as required in the cell transmission model [9]. The upper bounds are later used to compute the internal boundary flow solution to the junction models. This section discusses the formulation of the upper bounds for a single link l. The link ID subscript l is included in the notation in this section indicating that the same formulation will later be applied to each link in a network.

2.1. Semi-explicit HJ PDE solver on a single link. This subsection reviews the semi-explicit HJ PDE solver [5, 6, 33]. In the remainder of this article, we further assume the Hamiltonian $\psi_l(\cdot)$ on the link is defined as a piecewise affine function [9,

11]:

$$(3) \qquad \psi_l(\rho) = \begin{cases} v_l^f \rho & \text{if } \rho \in [0, \rho_l^\kappa], \\ w_l(\rho - \rho_l^m) & \text{if } \rho \in [\rho_l^\kappa, \rho_l^m]. \end{cases}$$

The parameters ρ_l^κ, ρ_l^m, v_l^f, and w_l represent the critical density, the maximal density, the free flow speed, and the maximum negative congestion wave speed on link l. The capacity is then computed as $q_l^{max} = v_l^f \rho_l^\kappa$. These parameters are assumed to be known, and can be obtained either from the Highway Capacity Manual [1] or calibrated from measurement data [14]. Besides the triangular fundamental diagram, the proposed convex scheme can be applied to all concave piecewise linear fundamental diagrams, which allows the formulation of piecewise linear constraints. Other concave fundamental diagrams (e.g., the quadratic Greenshields diagram) can be approximated by concave piecewise linear functions for the convex scheme to be applicable.

In the semi-explicit HJ PDE solver, the initial and boundary conditions of the HJ PDE (2) on the link l are given by piecewise affine functions defined on an arbitrarily discretized grid. Note that the discretized grid is only required at the boundary of the time-space domain $\{\{0\} \times [a_l, b_l]\} \cup \{[0, t_{max}] \times \{a_l, b_l\}\}$, which is fundamentally different from the discretization of the entire time-space domain $[0, t_{max}] \times [a_l, b_l]$ into cells and steps in other schemes [9, 17]. Specifically, the initial condition at $t = 0$ is defined over an arbitrary space grid $\{x_0, x_i \; \forall i \in \mathcal{I} := \{1, 2, \ldots, i_{max}\} \mid x_0 = a_l, x_i = a_l + \sum_{\eta=1}^{i} \Delta x_\eta\}$, where Δx_η is the length of spatial interval η. Similarly, the upstream $x = a_l$ and downstream $x = b_l$ boundary conditions are defined over a time grid $\{t_0, t_j \; \forall j \in \mathcal{J} := \{1, 2, \ldots, j_{max}\} \mid t_0 = 0, t_j = \sum_{\eta=1}^{j} \Delta t_\eta\}$, where Δt_η is the duration of the temporal interval η. For conciseness of notation, we omit the subscript l for the grids, which may be link specific in the general case.

On the time space grid $\{t_0, t_1, \ldots, t_{j_{max}}\} \times \{x_0, x_1, \ldots, x_{i_{max}}\}$, the initial and boundary conditions for the HJ PDE (2) are defined as piecewise affine functions, which are piecewise linear in closed intervals:

$$(4) \qquad \begin{aligned} \mathbf{M}_{l,0}(x) &= \left\{ c_{l,0}^i(x) \text{ if } x \in [x_{i-1}, x_i] \mid i \in \mathcal{I} \right\}, \\ \mathbf{M}_{a_l}(t) &= \left\{ c_{a_l}^j(t) \text{ if } t \in [t_{j-1}, t_j] \mid j \in \mathcal{J} \right\}, \\ \mathbf{M}_{b_l}(t) &= \left\{ c_{b_l}^j(t) \text{ if } t \in [t_{j-1}, t_j] \mid j \in \mathcal{J} \right\}. \end{aligned}$$

The terms $c_{l,0}^i(x), c_{a_l}^j(t), c_{b_l}^j(t)$ respectively represent the affine initial or boundary condition defined in the ith space or jth time interval. For compactness, we denote the set of affine initial and boundary conditions for all intervals by $\mathcal{C}_l$:

$$\mathcal{C}_l := \{c_{l,0}^i(x), \; c_{a_l}^j(t), \; c_{b_l}^j(t) \mid \forall i \in \mathcal{I}, j \in \mathcal{J}\}.$$

In general, the Moskowitz solution $\mathbf{M}_l(t, x)$ to the HJ PDE (2) cannot be computed explicitly for arbitrary piecewise affine initial and boundary conditions (4). However, each affine initial and boundary condition defined in their respective interval, e.g., $c_{l,0}^i, c_{a_l}^j, c_{b_l}^j$, can be used to compute an explicit *partial solution* in the time-space domain $[0, t_{max}] \times [a_l, b_l]$ by the *Lax–Hopf* formula. At each point (t, x), a partial solution for each affine initial or boundary condition in $\mathcal{C}_l$ can be obtained. The main result of the single link HJ PDE solver [5, 6, 33] shows that the Moskowitz solution is the minimum of all partial solutions at (t, x), which is known as the *inf-morphism* property. Using the *Lax–Hopf* formula and the *inf-morphism* property,

the Moskowitz solution $\mathbf{M}_l(t, x)$ in the domain $[0, t_{max}] \times [a_l, b_l]$ can be computed semi-explicitly as shown next.

PROPOSITION 1 (explicit partial solution [2, 5, 6]).   *The partial solution, i.e.,* $\mathbf{M}_l^c(t, x)$ *in the domain* $[0, t_{max}] \times [a_l, b_l]$ *associated with each affine initial or boundary condition* $c \in \mathcal{C}_l$ (4)*, can be explicitly expressed as a linear function of the initial and boundary conditions using the Lax–Hopf formula.*

*   The partial solution* $\mathbf{M}_l^{c_{a_l}^j}(t, x)$ *associated with the upstream boundary condition* $c_{a_l}^j(t)$ $\forall j \in \mathcal{J}$ *is written as follows:*

(5)

$$
\mathbf{M}_l^{c_{a_l}^j}(t, x) = \begin{cases} c_{a_l}^j(t_{j-1}) + \left(\frac{c_{a_l}^j(t_j) - c_{a_l}^j(t_{j-1})}{\Delta t_j} \right) \left(t - \frac{x - a_l}{v_l^f} - t_{j-1} \right) & \text{if } t_{j-1} + \frac{x - a_l}{v_l^f} \leq t \\ & \text{and } t < t_j + \frac{x - a_l}{v_l^f}, \\ c_{a_l}^j(t_j) + \rho_l^\kappa v_l^f \left(t - t_j - \frac{x - a_l}{v_l^f} \right) & \text{if } t \geq t_j + \frac{x - a_l}{v_l^f}, \\ +\infty & \text{otherwise.} \end{cases}
$$

The explicit solutions $\mathbf{M}_l^{c_{t,0}^i}(t, x)$ *and* $\mathbf{M}_l^{c_{b_l}^j}(t, x)$ *associated with the initial and downstream boundary conditions are defined similarly; see* [33] *for a complete description.*

The partial solution domain (i.e., where $\mathbf{M}_l^{c_{a_l}^j}(t, x)$ is finite) for each affine initial or boundary condition $c \in \mathcal{C}_l$ consists of two parts, namely the characteristic domain and the fan domain. In the partial solution associated with the upstream boundary conditions (5), the characteristic domain is $\{(t, x) \mid t_{j-1} + (x - a_l)/v_l^f \leq t < t_j + (x - a_l)/v_l^f\}$ in the first line of (5) and the fan domain is $\{(t, x) \mid t \geq t_j + (x - a_l)/v_l^f\}$ in the second line. The vehicle speed and density are constant in the characteristic domain, while the fan domain represents a rarefaction wave connecting the characteristic domain of adjacent affine initial or boundary conditions. We refer the reader to [33] for a detailed interpretation of the partial solutions. Physically, the partial solution gives the largest possible vehicle ID in the solution domain by only considering the information in each affine initial or boundary condition.

By the explicit formula (5), a partial solution $\mathbf{M}_l^c(t, x)$ can be computed for each $c \in \mathcal{C}_l$ at each point (t, x). For the Moskowitz solution at (t, x) to be compatible with all affine initial and boundary conditions, it must be less than or equal to the smallest vehicle ID computed by all partial solutions. The following proposition constructs the solution to HJ PDE (2) from the set of partial solutions.

PROPOSITION 2 (inf-morphism property [2, 5, 6]).   *The Moskowitz solution* $\mathbf{M}_l(t, x)$ *to the HJ PDE* (2) *with piecewise affine initial and boundary conditions* (4) *can be computed as the minimum of all partial solutions defined in Proposition* 1 *associated with each affine initial and boundary condition:*

$$
\mathbf{M}_l(t, x) = \min_{c \in \mathcal{C}_l} \mathbf{M}_l^c(t, x) \ \ \forall (t, x) \in [0, t_{\max}] \times [a_l, b_l].
$$

In summary, the traffic density on a single link can be computed as follows: (i) compute the partial solutions $\mathbf{M}_l^c(t, x) \ \forall c \in \mathcal{C}$; (ii) compute the minimum among the set of partial solutions at (t, x) to obtain the Moskowitz solution $\mathbf{M}_l(t, x)$; (iii) take the derivative of $\mathbf{M}_l(t, x)$ with respect to x to recover the traffic density $\rho_l(t, x)$.

2.2. Linear constraints on the boundary flows. The semi-explicit single link HJ PDE solver assumes the initial and boundary conditions of the link are given.

In the cases when the downstream or the upstream boundary condition is unknown, a feasible set can be computed denoting the maximum flow that can be sent or received on the link based on the initial condition and the boundary condition at the other end of the link.

By Proposition 3, any boundary flow value in the feasible set can be prescribed as a strong boundary condition for the link while guaranteeing the existence of a unique weak solution to the HJ PDE.

PROPOSITION 3 (compatibility conditions [7]). *Suppose the initial and the downstream boundary conditions are given in a piecewise affine form for the HJ PDE for a link l; the upstream boundary flow data $\tilde{q}_{a_l}(t)$ in the continuous time domain prescribes the boundary condition to the HJ PDE* (2) *in the strong sense, a.e. $t \in (0, t_{max}], \mathbf{M}_l(t, a_l) = \mathbf{M}_{a_l}(t) = \int_{\tau=0}^{t} \tilde{q}_{a_l}(\tau)\mathrm{d}\tau$ if and only if*

$$(6) \qquad \mathbf{M}_{a_l}(t) \leq \min_{\forall c \in \mathcal{C}_l} \mathbf{M}_l^c(t, a_l) \qquad \forall t \in [0, t_{max}].$$

Similarly, given piecewise affine initial and upstream boundary conditions, the downstream boundary flow data $\tilde{q}_{b_l}(t)$ in the continuous time domain prescribes the boundary condition to the HJ PDE (2) *in the strong sense, a.e. $t \in (0, t_{max}], \mathbf{M}_l(t, b_l) = \mathbf{M}_{b_l}(t) = \int_{\tau=0}^{t} \tilde{q}_{b_l}(\tau)\mathrm{d}\tau + c_{l,0}^{i_{max}}(b_l)$ if and only if*

$$(7) \qquad \mathbf{M}_{b_l}(t) \leq \min_{\forall c \in \mathcal{C}_l} \mathbf{M}_l^c(t, b_l) \qquad \forall t \in [0, t_{max}].$$

The magnitude of the term $c_{l,0}^{i_{max}}(b_l)$ gives the number of vehicles initially on the link.

The *compatibility conditions* give the upper bound of the boundary flows in a continuous functional space in which it is difficult to be used for analyzing the junction dynamics. In addition, as shown in [5, 6], if the given initial and boundary conditions are piecewise affine and the fundamental diagram is triangular, then the unknown boundary flow belongs to a piecewise constant functional space. Therefore, in the numerical implementation, we assume an arbitrary boundary grid with interval length $\Delta t_j, j \in \mathcal{J}$, is provided. At each interval j, the continuous boundary flow data $\tilde{q}_{a_l}(t)$ is approximated by the average flow, i.e., $q_{a_l}(j) = \frac{1}{\Delta t_j} \int_{t_{j-1}}^{t_j} \tilde{q}_{a_l}(\tau)\mathrm{d}\tau, t \in [t_{j-1}, t_j]$, at the upstream boundary. Similarly, $\tilde{q}_{b_l}(t)$ is approximated by $q_{b_l}(j) = \frac{1}{\Delta t_j} \int_{t_{j-1}}^{t_j} \tilde{q}_{b_l}(\tau)\mathrm{d}\tau, t \in [t_{j-1}, t_j]$ at the downstream boundary. This approximation allows the construction of an explicit form of the constraints for the boundary flows which is essential for solving the junction models.

By applying the *compatibility conditions* at the boundary grid points $(t, x) \in \{t_0, t_1, \ldots, t_{j_{max}}\} \times \{a_l, b_l\}$, the explicit feasible set of boundary flows that can be sent $\mathcal{F}_{l,s}$ or received $\mathcal{F}_{l,r}$ on the link can be obtained. Recall the relationship between the the Moskowitz downstream boundary condition and the boundary flow, i.e., $\mathbf{M}_{b_l}(t_j) = \sum_{\eta=1}^{j} q_{b_l}(\eta)\Delta t_\eta + c_{l,0}^{i_{max}}(b_l)$. Given the initial condition $c_{l,0}^i \ \forall i \in \mathcal{I}$ and the upstream boundary condition $c_{a_l}^j \ \forall j \in \mathcal{J}$, the feasible set of downstream boundary flows that can be sent on the link at each interval is defined as

$$(8) \quad \mathcal{F}_{l,s} := \left\{ q(j) \ \forall j \in \mathcal{J} \mid \sum_{\eta=1}^{j} q(\eta)\Delta t_\eta + c_{l,0}^{i_{max}}(b_l) \leq \mathbf{M}_l^c(t_j, b_l) \ \forall j \in \mathcal{J}, \forall c \in \mathcal{C}_l \right\},$$

where the subscript s denotes the sending flow and l is the link label. Similarly, given the initial condition $c_{l,0}^i \ \forall i \in \mathcal{I}$ and the downstream boundary condition $c_{b_l}^j \ \forall j \in \mathcal{J}$,

the explicit feasible set $\mathcal{F}_{l,r}$ of the upstream boundary flows that can be received on the link is formulated as follows:

$$(9) \qquad \mathcal{F}_{l,r} := \left\{ q(j) \ \forall j \in \mathcal{J} \mid \sum_{\eta=1}^{j} q(\eta)\Delta t_\eta \leq \mathbf{M}_l^c(t_j, a_l) \ \forall j \in \mathcal{J}, \forall c \in \mathcal{C}_l \right\},$$

where the subscripts l, r denote the receiving flow on link l. It is easy to verify that the inequality constraints are linear in the unknown boundary flows $q(j)$ by using the explicit forms of $\mathbf{M}_l^c(t,x)$ in (5), and realizing the relationship $q(j)\Delta t_j = c_{a_l}^j(t_j) - c_{a_l}^j(t_{j-1})$ for the downstream boundary flow (8), and $q(j)\Delta t_j = c_{b_l}^j(t_j) - c_{b_l}^j(t_{j-1})$ for the upstream boundary flow (9). Moreover, the capacity constraints, i.e., $q(j) \leq q_l^{max}$, are built into the feasible set by the constraints $\mathbf{M}_{a_l}(t_j) \leq \mathbf{M}_l^{c_{a_l}^{j-1}}(t_j, a_l)$, $\mathbf{M}_{b_l}(t_j) \leq \mathbf{M}_l^{c_{b_l}^{j-1}}(t_j, b_l)$.

In the extension of the single link HJ PDE solver to a network, the feasible sets of the sending and receiving flows of links are used to compute the internal boundary flows based on the junction models which is discussed in detail in the next section.

3. Convex formulation of traffic on networks. This section focuses on junction models and the development of a junction solver which computes the internal boundary flows on the network. Since the emphasis of this article is the formulation of a convex program for solving the selected junction models, we focus on a network consisting of three links connected by a merge or diverge junction as shown in Figure 1 in the remainder of this article. In addition to the merge and diverge, a simpler junction is the connection where one upstream link is connected to a downstream link. The connection junction is useful in modeling the road network when the physical property of the road changes (e.g., reduction of lanes).
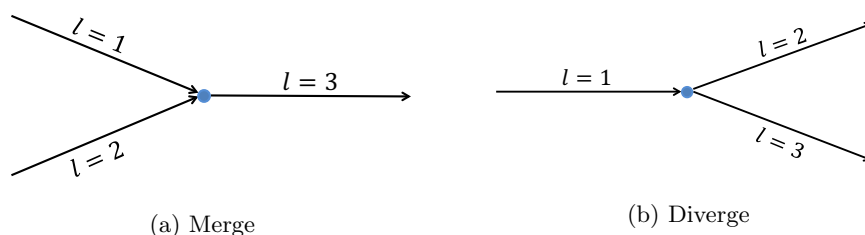

(a) Merge   (b) Diverge

FIG. 1. *A transportation network containing a merge or diverge junction with links indexed by l.*

3.1. Junction models on a network. A junction model describes how the internal boundary flows are distributed across the junction. Specifically, a junction model defines a unique internal boundary flow solution which reflects realistic physical behavior of traffic, such as flow maximization and routing preferences.

Merge model. This article adopts an existing model [15, 18] for the merge junction in Figure 1a. The merge model can be summarized into the following three rules:
(A1) The mass across the junction is conserved.
(A2) The throughput flow is maximized subject to the maximum flow that can be sent or received on each connecting link.
(A3) The distribution of the internal boundary flows, i.e., $q_{a_3} \mapsto (q_{b_1}, q_{b_2})$, satisfies a priority equation $q_{b_2} = P q_{b_1}$, where P is a prescribed parameter that models

the priority of upstream flows. When (A3) conflicts with (A2), that is, the internal boundary flow solution that satisfies the priority equation does not maximize the throughput, then (A3) is relaxed, i.e., the solution satisfies (A2) and minimizes the deviation from the prescribed priority parameter, e.g., $\|q_{b_2}/q_{b_1} - P\|_1$.

Diverge model. At a diverge junction in Figure 1b, this article proposes a model defined by the following rules:

(A1') The mass across the junction is conserved.

(A2') The throughput flow is maximized subject to the maximum flow that can be sent or received on each connecting link.

(A3') The distribution of the internal boundary flows, i.e., $q_{b_1} \mapsto (q_{a_2}, q_{a_3})$, satisfies $q_{a_3} = D q_{a_2}$, where D is a prescribed parameter that models the routing preference to the downstream links. When (A3') conflicts with (A2'), that is, the internal boundary flow solution that satisfies the distribution equation does not maximize the throughput, then (A3') is relaxed, such that the solution satisfies (A2') and minimizes the deviation from the prescribed distribution parameter, e.g., $\|q_{a_3}/q_{a_2} - D\|_1$.

The proposed diverge model is a FIFO model with a varying distribution parameter. The classic FIFO diverge model maximizes the throughput subject to the distribution rule $q_{a_3}(j) = D q_{a_2(j)}$ with a constant distribution parameter D. The classic FIFO model circumvents the difficulty of resolving the conflicts between the throughput maximization and flow distribution, but it produces unrealistic solutions in some applications. For example, using the classic FIFO model, a blocked offramp will completely stop the traffic on all lanes of a multilane highway, which is unlikely. To resolve this issue, several diverge junction models were proposed previously, such as a multilane junction model (non-FIFO) [19], a dynamic distribution parameter [24], and a junction model with internal dynamics [27]. In the same spirit of these models, this article proposes a diverge junction model that produces similar traffic condition dependent solutions without introducing additional complexity of non-FIFO models on the traffic dynamics. The main assumption of the proposed diverge model is that drivers will reroute to the other link if the initially desired link becomes congested [20]. The rerouting assumption makes the composition of queuing vehicles on the upstream link time-invariant which is critical for the unique solution to be consistent [26].

The connection junction model is significantly simpler compared to the merge and diverge models since there is no distribution or priority parameters involved. Therefore, the connection model simply maximizes the throughput.

The structure of the merge and diverge models used in this article are similar, i.e., both maximize the throughput and then minimize the deviation from the prescribed priority or distribution parameters. Therefore, the remainder of this section will focus on the formulation of a junction solver for the merge model and note the same analysis can be easily transferred to the diverge. The connection junction solver will be briefly discussed considering its simplicity.

3.2. Junction solver over a single interval. This subsection proposes a junction solver in the form of a convex program that computes the internal boundary flow solution at a single time interval for the merge model.

To compute the unique internal boundary flows $q_{b_1}(j), q_{b_2}(j)$ at time interval j, we assume that the unique internal boundary flow solutions up to interval $j-1$ are given and denoted by $q_{a_l}^*(\eta), q_{b_l}^*(\eta), \eta \in \{1, 2, \ldots, j-1\}$. It should be noted that this assumption requires the merge junction problem to be solved sequentially in time.

Accordingly, the convex set (8) of the internal boundary flows that can be sent on link l for all time interval up to j is reduced to a convex set of internal boundary flows that can be sent at interval j:

$$(10) \qquad \mathcal{F}_{l,s}^{j} := \left\{ q(j) \mid q(j)\Delta t_j \le \mathbf{M}_l^c(t_j, b_l) - \sum_{\eta=1}^{j-1} q_{b_l}^*(\eta)\Delta t_\eta - c_{l,0}^{i_{max}}(b_l) \;\; \forall c \in \mathcal{C}_l \right\}.$$

Similarly, the internal boundary flows that can be received on link l at time interval j are subject to a reduced feasible set of (9):

$$(11) \qquad \mathcal{F}_{l,r}^{j} := \left\{ q(j) \mid q(j)\Delta t_j \le \mathbf{M}_l^c(t_j, a_l) - \sum_{\eta=1}^{j-1} q_{a_l}^*(\eta)\Delta t_\eta \;\; \forall c \in \mathcal{C}_l \right\}.$$

The terms $\mathbf{M}_l^c(t_j, b_l) - c_{l,0}^{i_{max}}(b_l)$ and $\mathbf{M}_l^c(t_j, a_l)$ denote the maximum number of vehicles that can be sent or received during time $(0, t_j)$. The summations of the given boundary flows $\sum_{\eta=1}^{j-1} q_{b_l}^*(\eta)\Delta t_\eta$ and $\sum_{\eta=1}^{j-1} q_{a_l}^*(\eta)\Delta t_\eta$ represent the number of vehicles that have been sent or received during $(0, t_{j-1})$. Hence, the right-hand side terms in (10) and (11) are constants representing the maximum number of vehicles that can be sent or received during interval j, i.e., (t_{j-1}, t_j).

As shown next, the merge junction solver is posed as a convex program with a carefully constructed objective function to accommodate the throughput maximization (A2) and the flow priority (A3) objectives. The equations (10) and (11) combined define the constraint set of the convex program.

DEFINITION 4 (merge junction solver over a single interval). *The junction solver for computing the internal boundary flow solution $(q_{b_1}(j), q_{b_2}(j))$ at a merge during interval j is formulated in the form of a convex program as follows:*

$$(12) \qquad \begin{array}{ll} \underset{q_1(j), q_2(j)}{Maximize} & f\left(q_1(j), q_2(j)\right) \\ s.t. & q_1(j) \in \mathcal{F}_{1,s}^{j} \quad (10), \\ & q_2(j) \in \mathcal{F}_{2,s}^{j} \quad (10), \\ & q_3(j) \in \mathcal{F}_{3,r}^{j} \quad (11), \\ & q_3(j) = q_1(j) + q_2(j), \end{array}$$

where $f(q_1(j), q_2(j))$ is a convex function of $q_1(j), q_2(j)$ and satisfies

$$(13a) \qquad \frac{\partial f}{\partial q_l(j)} > 0 \qquad\qquad \forall l \in \{1, 2\},$$

$$(13b) \qquad \frac{\partial f}{\partial q_1(j)} > \frac{\partial f}{\partial q_2(j)} \qquad\qquad when \;\; q_2(j) \ge Pq_1(j),$$

$$(13c) \qquad \frac{\partial f}{\partial q_1(j)} < \frac{\partial f}{\partial q_2(j)} \qquad\qquad when \;\; q_2(j) < Pq_1(j).$$

The junction solver CP (12) computes the unique internal boundary flow solution defined by the merge junction model (A1), (A2), and (A3), as stated in the following proposition.

PROPOSITION 5. *The merge junction solver CP (12) computes the unique internal boundary flow solution $q^*(j) = \left(q_{b_1}^*(j), q_{b_2}^*(j), q_{a_3}^*(j)\right)$ at interval j, where $q^*(j)$ satisfies (A1), (A2), (A3):*

(i) *The internal boundary flows satisfy mass conservation rule* (A1), $q_{a_3}^*(j) = q_{b_1}^*(j) + q_{b_2}^*(j)$.

(ii) *The throughput flow at the junction is maximized subject to the feasible sets on connecting links* (A2), *i.e.,*

$$q^*(j) \in \mathcal{Q}_j := \underset{q(j) \in \mathcal{F}_{1,s}^j \times \mathcal{F}_{2,s}^j \times \mathcal{F}_{3,r}^j}{\operatorname{argmax}} q_{a_3}(j).$$

(iii) *The deviation from the priority equation is minimized* (A3), *e.g.,*

$$q^*(j) = \underset{q(j) \in \mathcal{Q}_j}{\operatorname{argmin}} \|q_{b_2}(j) - P q_{b_1}(j)\|_1.$$

Proof. A sketch of the proof is provided here, and a detailed proof appears in the arXiv version of this article [30]. By construction of the junction solver, the constraint set of the convex program is nonempty, which guarantees the existence of a solution. The property (i) is satisfied since it is explicitly included in the constraint set in CP (12). The property (ii) is guaranteed by the condition (13a) since the gradient with respect to each internal boundary flow is strictly positive. The intuition of conditions (13b), (13c) is that the gradient of the objective function f points towards the line $q_{b_2}(j) = P q_{b_1}(j)$ from both sides in the feasible set. Consequently, the points closer to the distribution line have a smaller deviation $\|q_{b_2}(j)/q_{b_1}(j) - P\|_1$ as defined in (iii). The uniqueness of the solution is proved in the detailed proof using conditions in (13). ☐

The consistency (equivalently invariance) of the merge junction solver can be verified by direct application of the definition of invariance [26].

The unique solution during a single time interval j computed by the junction solver (12) for the merge is illustrated in Figure 2. There are in total three scenarios depending on the feasible sets on links (10), (11), namely, (i) when the maximum receiving flow on link 3 exceeds the total sending flow from links 1 and 2 in Figure 2a; (ii) when the maximum receiving flow is smaller than the total sending flow from links 1 and 2 but the prescribed priority ratio cannot be followed exactly in Figure 2b; (iii) when the maximum receiving flow is smaller than the total sending flow on links 1 and 2 and the prescribed priority ratio can be followed exactly in Figure 2c. The feasible set of the convex program is denoted by the shaded area, where the upper bounds of the feasible internal boundary flows for three links are, respectively, $\bar{N}_1 = \max \{q_1(j) \mid q_1(j) \in \mathcal{F}_{1,s}^j\}$, $\bar{N}_2 = \max \{q_2(j) \mid q_2(j) \in \mathcal{F}_{2,s}^j\}$, and $\bar{N}_3 = \max \{q_3(j) \mid q_3(j) \in \mathcal{F}_{3,r}^j\}$. The solid lines denote the maximum sending flows from links 1 and 2. The dashed line denotes the maximum receiving flow on link 3. The dotted line denotes the prescribed priority of the boundary flows, i.e., $q_1(j) = \frac{q_3(j)}{1+P}$, $q_2(j) = \frac{P q_3(j)}{1+P}$. The unique solution computed by the diverge junction solver CP (12) is marked at point Q.

In scenario (i), see Figure 2a, the maximum receiving flow on link 3 exceeds the the total flow that can be sent by links 1 and 2 combined. Hence, the single point that maximizes the throughput admits the maximum sending flow from links 1 and 2, and is the optimal solution to CP (12).

In scenario (ii), see Figure 2b, the total sending flow is higher than the maximum receiving flow on the downstream link, and the sending flows cannot be distributed exactly following the priority rule. In this case, link 3 first admits all flows from the higher priority link (i.e., link 1 in Figure 2b), and then admits as much flow as

possible for the lower priority link to maximize the throughput. Consequently, the optimal solution to CP (12) is the solution Q that is closest to the dotted line among the solutions on the dashed line within the feasible set.

In the last scenario (iii), see Figure 2c, the total sending flow from upstream links combined is higher than the maximum receiving flow on link 3, and the sending flow can be distributed exactly following the priority ratio. In this case, there is no conflict between maximizing the throughput (A2) and following the flow priority (A3). Therefore, the solution at Q that satisfies both (A2) and (A3) is the optimal solution to the merge junction solver CP (12).

Similar to the merge model, the diverge model also has three scenarios for determining the unique solution. The three scenarios are depicted and discussed in the arXiv version of this article [30].
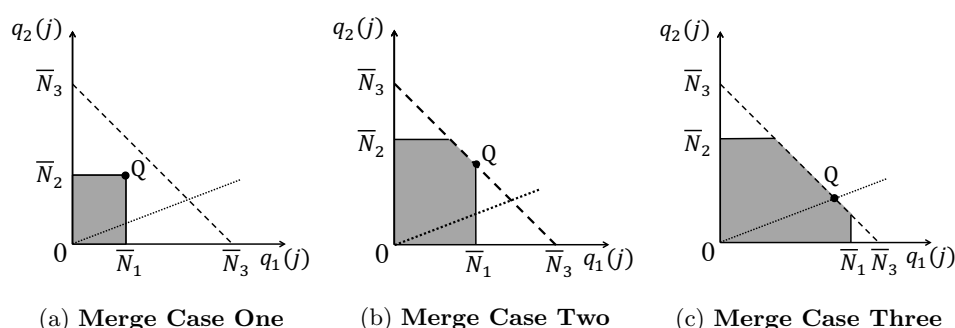

(a) **Merge Case One**   (b) **Merge Case Two**   (c) **Merge Case Three**

FIG. 2. *Three scenarios at a merge. Links 1 and 2 merge to link 3. The two solid lines represent the maximum sending flow on links 1 and 2. The dashed line denotes the maximum receiving flow on link 3. The dotted line denotes the prescribed priority of sending flows. The shaded area is the feasible set of boundary flows (10), (11). The unique solution computed by CP (12) is depicted by Q.*

The junction solver CP (12) applies to an arbitrary convex objective function $f(q_1(j), q_2(j))$ that satisfies (13), and subderivatives can be used if $f(q_1(j), q_2(j))$ is not differentiable. Though no explicit form of the objective function is used to define the solver, selecting a suitable objective function is straightforward, as shown in the following example.

Example. At a merge, a convex objective function to maximize the throughput (A2) and then minimize the deviation from prescribed flow priority ratio (A3) for any flow maximizing solution can be defined as follows:

$$(14) \qquad f(q_1(j), q_2(j)) = \alpha \left(q_1(j) + q_2(j) \right) - \beta \left(q_2(j) - Pq_1(j) \right)^2,$$

where α and β are weights defined by

$$\alpha = 1 - \beta, \quad \beta = \min \left(\frac{1}{1 + 2P^2 q_1^{max} + \epsilon}, \frac{1}{1 + 2q_2^{max} + \epsilon} \right), \quad \text{and } \epsilon = 0.01.$$

The derivation of the coefficients is presented as follows. Define the objective function as (14) and assume $\alpha > 0$ and $\beta > 0$. The conditions (13b) and (13c) in Definition 4 are trivially satisfied given $\beta > 0$. Then by condition (13a),

$$\frac{\partial f}{\partial q_1(j)} = \alpha - 2\beta(q_2(j) - Pq_1(j))(-P) > 0 \quad \text{and} \quad \frac{\partial f}{\partial q_2(j)} = \alpha - 2\beta(q_2(j) - Pq_1(j)) > 0.$$

Let $\alpha = 1 - \beta$ and $\beta \in (0, 1)$, and rearrange the above inequalities as

$$1 > \beta \left(1 - 2Pq_2(j) + 2P^2 q_1(j) \right) \qquad \text{and} \quad 1 > \beta \left(1 + 2q_2(j) - 2Pq_1(j) \right).$$

To guarantee the above inequalities hold for all possible values of $q_1(j)$ and $q_2(j)$, we use the following chain rule:

$$\beta\left(1 + 2P^2 q_1^{max}\right) \geq \beta\left(1 + 2P^2 q_1(j)\right) \geq \beta\left(1 - 2P q_2(j) + 2P^2 q_1(j)\right),$$
$$\beta\left(1 + 2q_2^{max}\right) \geq \beta\left(1 + 2q_2(j)\right) \geq \beta\left(1 + 2q_2(j) - 2P q_1(j)\right).$$

Hence, it suffices to require

$$1 > \beta\left(1 + 2P^2 q_1^{max}\right) \quad \text{and} \quad 1 > \beta\left(1 + 2q_2^{max}\right).$$

Then the value of β can be selected as

$$\beta = \min\left(\frac{1}{1 + 2P^2 q_1^{max} + \epsilon}, \frac{1}{1 + 2q_2^{max} + \epsilon}\right) \in [0, 1],$$

where ϵ is a small positive constant (e.g., 0.01) to guarantee the strict inequality. Accordingly, α is computed then by $\alpha = 1 - \beta$.

The connection junction model only requires the mass conservation and maximization of the throughput. Therefore, the connection junction solver can be formulated by maximizing the throughput (i.e., downstream flow of upstream link) subject to the maximum sending and receiving boundary flows defined in (10) and (11).

3.3. Junction solver over the entire horizon. As shown in the previous subsection, the unique solution over a single time interval can be computed by a junction solver in the form of a convex optimization program. When constructing the constraint set of CP (12) at interval j, the unique solutions up to interval $j-1$ were assumed to be known (10), (11). As a result, computing the internal boundary flow solution over the entire time horizon requires consecutively constructing and solving a convex program at each time interval. This subsection presents a merge junction solver in the form of a single convex program for computing the interval boundary flows over the entire time horizon.

The intuition of the junction solver as a single convex program is that the union of the constraint set for each interval $\mathcal{F}_{l,r}^j$ or $\mathcal{F}_{l,s}^j$ is a subset of the constraint set $\mathcal{F}_{l,r}$ or $\mathcal{F}_{l,s}$ over the entire time horizon. Given the structure of the constraint set, it is possible to design the objective function for a single convex program such that the solution over the entire time horizon is equivalent to the solution computed by sequentially solving a convex program at each time interval. The following definition articulates the conditions required for the objective function.

DEFINITION 6 (merge junction solver over the entire time horizon). *The junction solver for computing the internal boundary flows at a merge* (q_{b_1}, q_{b_2}) *where* $q_{b_1} = \{q_{b_1}(j) \ \forall j \in \mathcal{J}\}$ *and* $q_{b_2} = \{q_{b_2}(j) \ \forall j \in \mathcal{J}\}$ *for the entire time horizon is defined as the following convex program:*

(15)
$$\begin{aligned}
\underset{q_1, q_2}{Maximize} \quad & f(q_1, q_2) \\
s.t. \quad & q_1 \in \mathcal{F}_{1,s} \quad (8), \\
& q_2 \in \mathcal{F}_{2,s} \quad (9), \\
& q_3 \in \mathcal{F}_{3,r} \quad (9), \\
& q_3(j) = q_1(j) + q_2(j) \ \forall j \in \mathcal{J},
\end{aligned}$$

where $q_1 = \{q_1(j) \ \forall j \in \mathcal{J}\}$, $q_2 = \{q_2(j) \ \forall j \in \mathcal{J}\}$, $q_3 = \{q_3(j) \ \forall j \in \mathcal{J}\}$. *The function*

$f(q_1, q_2)$ *is convex in* $q_1(j), q_2(j)$ $\forall j \in \mathcal{J}$ *and satisfies*

(16a) $$\frac{\partial f}{\partial q_1(1)\Delta t_1} > \frac{\partial f}{\partial q_1(2)\Delta t_2} > \cdots > \frac{\partial f}{\partial q_1(j_{max})\Delta t_{j_{max}}} > 0,$$

(16b) $$\frac{\partial f}{\partial q_2(1)\Delta t_1} > \frac{\partial f}{\partial q_2(2)\Delta t_2} > \cdots > \frac{\partial f}{\partial q_2(j_{max})\Delta t_{j_{max}}} > 0,$$

and $\forall j \in \{1, 2, \ldots, j_{max} - 1\}$,

(17a) $$\frac{\partial f}{\partial q_1(j)\Delta t_j} - \frac{\partial f}{\partial q_2(j)\Delta t_j} - \frac{\partial f}{\partial q_1(j+1)\Delta t_{j+1}} > 0 \quad \text{if} \quad q_2(j) \geq Pq_1(j),$$

(17b) $$\frac{\partial f}{\partial q_2(j)\Delta t_j} - \frac{\partial f}{\partial q_1(j)\Delta t_j} - \frac{\partial f}{\partial q_2(j+1)\Delta t_{j+1}} > 0 \quad \text{if} \quad q_2(j) < Pq_1(j),$$

and when $j = j_{max}$,

(18a) $$\frac{\partial f}{\partial q_1(j_{max})} - \frac{\partial f}{\partial q_2(j_{max})} > 0 \quad \text{if} \quad q_2(j_{max}) \geq Pq_1(j_{max}),$$

(18b) $$\frac{\partial f}{\partial q_1(j_{max})} - \frac{\partial f}{\partial q_2(j_{max})} < 0 \quad \text{if} \quad q_2(j_{max}) < Pq_1(j_{max}).$$

Before stating the formal proposition on the equivalence of the junction solver CP (15) to the single interval junction solver CP (12) solved sequentially for all intervals, we briefly interpret the conditions (16), (17), (18) on the objective function.

The conditions (16) assign higher weights to the internal boundary flows at earlier time intervals, such that the throughput at earlier intervals is first maximized. Consequently, the convex program will produce a solution that satisfies the throughput maximization rule (A2) for all intervals. The unique solution also requires minimum deviation of the solution from the prescribed parameter (A3), i.e., $q_{b_2}(j) = Pq_{b_1}(j)$ for all intervals $j \in \mathcal{J}$. The conditions (17), (18) define the direction of the gradient of the objective function as pointing towards the line $q_{b_2}(j) = Pq_{b_1}(j)$ from both sides. If (17), (18) are satisfied for CP (15), then conditions (13b), (13c) are satisfied for all CP_j at all intervals $\forall j \in \mathcal{J}$.

PROPOSITION 7. *The junction solver* (15) *gives the same unique solution* $\{q_{b1}(j), q_{b_2}(j), q_{a_3}(j) \mid \forall j \in \mathcal{J}\}$ *obtained by sequentially solving CP* (12) *at each time interval.*

Proof. A detailed proof is presented in the arXiv version of this article [30] and the intuition is briefly described as follows. The proof of Proposition 7 relies on the equivalence of the *Karush–Kuhn–Tucker* (KKT) conditions associated with the sequence of CPs (12) and CP (15). The main idea is to show that the set of KKT multipliers associated with the optimal solution to CP (15) also satisfies the KKT conditions associated with the same solution to CP (12) for each time interval. Since the constraints in CP (12) are linear, the KKT conditions are also sufficient conditions. Therefore, the solution to CP (15) is also the optimal solution to CP (12) for each time interval. □

The junction solver CP (15) does not provide an explicit form of the objective function. Similarly, as in Proposition 5, one can first define a weighted objective function with undetermined weights and then select values for the weights to satisfy the proposed conditions on the objective function. This process is illustrated in the following example.

Example. For simplicity, suppose the number of time steps $j_{max} = 2$. Define the following objective function for a diverge with weights $\alpha > 0, \beta > 0, \omega_1 > 0, \omega_2 > 0$:

$$f(q_1, q_2) = \sum_{j=1}^{2} w_j \left(\alpha \cdot (q_1(j) + q_2(j)) - \beta \cdot \|q_2(j) - Pq_1(j)\|_1 \right).$$

The term $q_1(j) + q_2(j)$ maximizes the throughput at time interval j and the term $\|q_2(j) - Pq_1(j)\|_1$ penalizes the deviation from the prescribed priority ratio. In addition, weights w_j are selected for the two intervals to assign higher weights to the internal boundary flows at earlier intervals, so the vehicles are not stopped and sent at later time intervals, which is unrealistic at a junction without actuators. By applying the conditions (16), (17), (18) on the objective function $f(q_2, q_3)$, the parameters and weight coefficients can be set as follows:

$$\alpha = 1 + P + \max(1, P), \quad \beta = 1, \quad \omega_1 = \frac{\alpha + \max(1, P)}{1 + P} + \epsilon, \quad \omega_2 = 1,$$

where $\epsilon > 0$ is a small positive constant to guarantee the strict inequality.

At the connection junction, the throughput must be maximized at all time intervals. The following lemma shows the unique solution over the entire time horizon for a connection can be computed in a single convex program.

LEMMA 8. *Consider a junction in which one upstream link (link 1) connects to one downstream link (link 2). The unique boundary flow solution $q_{b_1}(j)$ for all time intervals $j \in \mathcal{J}$ can be solved by*

(19)
$$\begin{aligned} &\underset{q_{b_1}}{Maximize} \quad f(q_{b_1}) \\ &s.t. \quad \{q_{b_1}(j) \mid \forall j \in \mathcal{J}\} \in \mathcal{F}_{1,s} \quad (8), \\ &\qquad \{q_{a_2}(j) \mid \forall j \in \mathcal{J}\} \in \mathcal{F}_{2,r} \quad (9), \\ &\qquad q_{b_1}(j) = q_{a_2}(j) \ \forall j \in \mathcal{J} \end{aligned}$$

if $f(q_{b_1})$ satisfies $\forall j \in \mathcal{J}$,

(20)
$$\frac{\partial f}{\partial q_{b_1}(1)\Delta t_1} > \frac{\partial f}{\partial q_{b_1}(2)\Delta t_2} > \cdots > \frac{\partial f}{\partial q_{b_1}(j_{max})\Delta t_{j_{max}}} > 0.$$

The condition (20) assigns a higher weight to the internal boundary flow at an earlier interval, hence satisfying the throughput maximization rule for all intervals. The proof of Lemma 8 can be derived using the same technique in the proof of Proposition 7 and is not detailed here.

Compared to a sequential scheme, such as the sequential convex program scheme or the cell transmission model (CTM) [10], the single convex program framework allows a natural extension to optimal traffic control which is demonstrated in section 4.

4. Application: On-ramp metering control for work zones. This section demonstrates how the convex optimization scheme for computing the internal boundary flows can be reformulated as a control framework. An example for on-ramp metering control in work zones is provided. The proposed on-ramp metering controller uses historical data and the real-time measurement data at the entrance and exit of each road section to predict the traffic states in the work zone, and then avoids congestion upstream of the work zone by directly penalizing the congested traffic states.

4.1. Optimal on-ramp metering control framework. The convex optimization framework in the previous section computes the internal boundary flows on a network using the HJ PDE link model and proposed junction solver. Particularly, the junction models are encoded in the objective function in the convex program in Definition 6. Without the conditions on the objective function in Definition 6, the junction dynamics are no longer modeled and the internal boundary flow solution can be any value that is optimal for a given arbitrary objective function. In this sense, the convex program is an optimal controller assuming all directions of traffic are signalized at the junction. We refer the reader to our earlier work [29] for a detailed discussion.

The optimal controller for an on-ramp meter, however, is more complicated since both uncontrolled freeway flows and controlled on-ramp flows appear in the convex program. The CP (15) is used for solving the joint PDEs in the unsignalized merge and hence needed to be modified such that one upstream link (i.e., the on-ramp) allows control input. Meanwhile, the two freeway links preserve the unique internal boundary flow solution, which is similar to the problem of solving the joint PDEs at a connection junction in Lemma 8. Therefore, in the formulation of the optimal controller, the convex program for the connection (19) is first used to guarantee the unique solution to the uncontrolled boundary flows. The remainder of this section shows how to include additional constraints and objectives into the convex program such that the on-ramp flows are controlled optimally to minimize the congestion.

4.2. Penalty on the congested states. In the convex optimization framework, the congested states on links can be directly penalized via sampling the traffic condition at a set of points of interest $\mathcal{P}$ which we refer to as *congestion sampling points*. For example, the set of congestion points is defined at discrete time points at a fixed location x_q, i.e., $\mathcal{P} := \{(t_k, x_q) \mid k \in \{1, 2, \ldots\}\}$. If the congestion does not extend past the fixed location x_q, the traffic state is considered to be lightly congested and no penalty will be computed. Otherwise, the objective function penalizes the congested states depending on its severity. This subsection shows how this penalty mechanism is incorporated in the convex program by including additional constraints and variables. We briefly summarize the main ideas on the formulation of the constraints relying on the property of the partial solutions, and we refer the reader to [33] for detailed interpretation of the property of partial solutions.

By Proposition 2, the Moskowitz solution at each congestion sampling point $(t_k, x_q) \in \mathcal{P}$ is computed as the minimum of the partial solutions computed from the affine initial conditions, the upstream boundary condition, or the downstream boundary condition. The affine initial conditions can be further grouped into two categories by whether the initial condition interval is in free flow or congested states, i.e., $\rho_{l,0}(i) \leq \rho_l^\kappa$ or $\rho_{l,0}(i) > \rho_l^\kappa$, noting the density is related to the initial condition (4) by $\rho_{l,0}(i) = \frac{c_{l,0}^i(x_i) - c_{l,0}^i(x_{i-1})}{\Delta x_i}$. Denote the sets of affine initial conditions that are in free flow and congested states, respectively, as $\mathcal{C}_{ff} = \{c_{l,0}^i(x) \mid \rho_{l,0}(i) \leq \rho_l^\kappa\}$ and $\mathcal{C}_{cs} = \{c_{l,0}^i(x) \mid \rho_{l,0}(i) > \rho_l^\kappa\}$ and the sets of affine upstream and downstream boundary conditions as $\mathcal{C}_{us} = \{c_{a_l}^j(t) \; \forall j \in \mathcal{J}\}$ and $\mathcal{C}_{ds} = \{c_{b_l}^j(t) \; \forall j \in \mathcal{J}\}$.

The partial solutions associated with $\mathcal{C}_{ff}$ and $\mathcal{C}_{us}$ imply free flow traffic conditions, and the partial solutions of $\mathcal{C}_{cs}$ and $\mathcal{C}_{ds}$ indicate congested traffic conditions in their respective characteristic solution domain. A congestion sampling point $(t_k, x_q) \in \mathcal{P}$ is in a free flow condition if and only if

$$(21) \qquad \mathbf{M}_l(t_k, x_q) = \mathbf{M}_{ff}(t_k, x_q) := \min\left(\mathbf{M}_l^c(t_k, x_q) \; \forall c \in \mathcal{C}_{ff} \cup \mathcal{C}_{us}\right).$$

In other words, the solution at (t_k, x_q) is in free flow if it is defined by the upstream condition or the free flow initial condition. Similarly, the congestion sampling point is in a congested state if and only if

$$(22) \qquad \mathbf{M}_l(t_k, x_q) = \mathbf{M}_{cs}(t_k, x_q) := \min\left(\mathbf{M}_l^c(t_k, x_q) \ \forall c \in \mathcal{C}_{cs} \cup \mathcal{C}_{ds}\right).$$

In addition, the partial solutions have the following properties: (i) if a point (t, x) is located in the characteristic domain of an affine initial or boundary condition, then the partial solution associated with the affine initial or boundary condition is equal to the solution $\mathbf{M}_l(t, x)$ [33]; (ii) the characteristic solution domains for $c \in \mathcal{C}_{ff} \cup \mathcal{C}_{us}$ do not overlap; see (5). Utilizing the above properties, the nonlinear operator min in (21) can be removed by precomputing to which characteristic domain the point (t_k, x_q) belongs. Similarly, the min operator in (22) can be removed by applying the same technique.

For each congestion sampling point (t_k, x_q), the following constraints are added to the convex program (15):

$$(23) \qquad \begin{array}{l} \forall (t_k, x_q) \in \mathcal{P}, \\ \left\{ \begin{array}{l} \mathbf{M}_{ff}(t_k, x_q) \leq \mathbf{M}_{cs}(t_k, x_q) + p_k, \\ p_k \geq 0, \end{array} \right. \end{array}$$

where p_k is a variable that denotes the penalty associated with the congested states at each point (t_k, x_q). These constraints guarantee $p_k > 0$ when the point (t_k, x_q) is congested, i.e., $\mathbf{M}_{ff}(t_k, x_q) > \mathbf{M}_{cs}(t_k, x_q)$ and $p_k = 0$ otherwise. The penalty variables p_k for all congestion sampling points in $\mathcal{P}$ are regarded as decision variables and minimized in the objective function.

4.3. Simulation configuration. This subsection briefly describes the experiment setup and the next subsection constructs an optimal on-ramp metering controller using the convex program (19) and the additional constraints (23).

The experiment setup for validating the optimal on-ramp metering controller is shown in Figure 3. A microscopic traffic simulation software, AIMSUN, is used to simulate a traffic environment and collect aggregated traffic data. The microscopic traffic simulator simulates the behavior of individual vehicles as a proxy for a real freeway network composed of human drivers, and is commonly used to validate traffic controllers based on macroscopic models [39]. The simulated data is then streamed to the optimal on-ramp metering controller implemented in MATLAB, which computes the optimal control signals based on the traffic dynamics modeled by the HJ PDE. Finally, the optimal on-ramp meter control is applied in the AIMSUN environment to simulate the evolution of traffic.

A 6 km stretch of freeway and an on-ramp is modeled in AIMSUN as in Figure 3: a two-lane freeway (link 1) merges with a single-lane on-ramp (link 2) and connects to a two-lane freeway (link 3). A downstream work zone creates a single-lane bottleneck which induces congestion if the traffic is not controlled. Loop detectors are assumed to be installed at the entrance and exit of each link with a detection cycle set at 30 seconds. An on-ramp meter is installed at the exit of the on-ramp. Due to the limited work zone capacity, severe congestion will be generated upstream of the work zone if the on-ramp is not controlled. In comparison, the proposed controller can limit the level of congestion by regulating on-ramp inflows.

The total simulated time horizon in this example is one hour. The controller is embedded in a *model predictive control* (MPC) scheme [4, 16, 32] which updates the
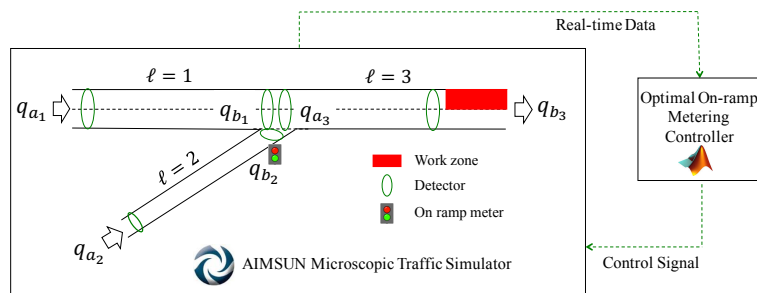
FIG. 3. *On-ramp metering control for a work zone: AIMSUN simulates a microscopic traffic environment and collects and feeds data to the optimal controller. The optimal controller implemented in MATLAB computes the optimal on-ramp signals and applies to AIMSUN.*

control signals based on real-time measurements. The MPC scheme is illustrated in Figure 4: (i) the optimal controller predicts the traffic states over the next 10-minute time horizon and computes the optimal ramp meter signal; (ii) AIMSUN applies only the first minute of the control signal to the on-ramp meter, simulates the evolution of the traffic, and feeds back the aggregated traffic data to the controller; (iii) the controller reoptimizes the ramp meter signal over the next 10-minute time horizon using the new traffic measurement data from AIMSUN; (iv) repeat steps (ii) and (iii) to adjust the optimal control signal to the realtime traffic measurement data.

In this example, the historical data is used for computing the optimal traffic signals over each 10-minute time horizon. The error of the measurement data is not modeled in this example and we refer the reader to our earlier work [28] on robust optimal control with incorporates measurement uncertainty.
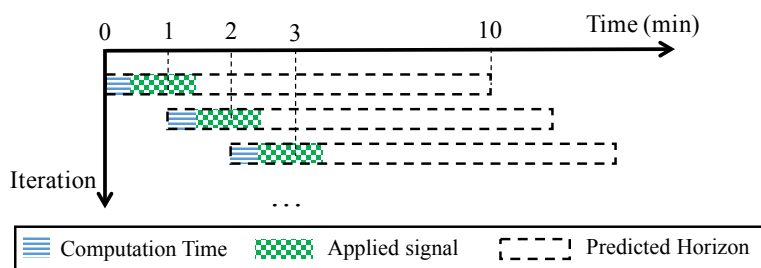

FIG. 4. *MPC scheme for the on-ramp metering control: Once new data is received, the optimal controller initializes a 10-minute predicted horizon for computing the optimal meter signals. New data is received each minute, and hence only the first minute of the control signal during the 10-min horizon is applied to the on-ramp meter.*

4.4. Formulation of the optimal controller. This subsection summarizes the constraints, designs the objective function, and formulates an optimal on-ramp metering controller.

4.4.1. Decision variables. During each 10-minute time horizon, the on-ramp boundary flows $q_{b_2}(j) \ \forall j \in \mathcal{J}$ are controlled and the freeway downstream flows $q_{b_1}(j), q_{b_3}(j) \ \forall j \in \mathcal{J}$ are computed using the junction model. Therefore, they are regarded as the decision variables in the convex program.

The inflow to the downstream freeway link $l = 3$ satisfies $q_{a_3}(j) = q_{b_1}(j) + q_{b_2}(j) \ \forall j \in \mathcal{J}$ and does not need to be explicitly included as a decision variable. The inflows to the upstream freeway and the on-ramp $q_{a_1}(j)$ and $q_{a_2}(j) \ \forall j \in \mathcal{J}$ are assumed to be known from the historical data.

In addition, the penalty p_k associated with each congestion sampling point (t_k, x_q) $\in \mathcal{P}$ is also used as the decision variable to penalize congestion in the workzone.

In summary, the decision variable of the convex program is defined as follows:

$$\mathcal{X} = \{q_{b_1}(j), q_{b_2}(j), q_{b_3}(j), p_k \mid \forall j \in \mathcal{J}, (t_k, x_q) \in \mathcal{P}\}.$$

4.4.2. Linear constraints. The constraints in the convex program for the optimal controller consist of the following linear inequalities and equalities:
- The boundary flows on each link l subject to the feasible constraints $\mathcal{F}_{l,s}$ and $\mathcal{F}_{l,r}$ defined in (8), (9).
- The internal boundary flows at the on-ramp junction satisfy mass conservation for all time steps.
- The congestion sampling points are selected as $\mathcal{P} = \{(t_k, x_q) \mid t_k = 30k \, s \ \forall k \in \{0, 1, 2, \ldots\}, \ x_q = 50 \, m\}$. The penalty variable associated with each congestion sampling point is subject to the constraints defined in (23).

4.4.3. Objective function. One of the main safety concerns in work zones is the high-speed rear-end crashes which can be caused by the congestion upstream of the work zone. Therefore, the primary objective used in this example is to improve the safety for traveling through the work zone by alleviating the congestion on the freeway at the upstream of the work zone. Meanwhile, a secondary objective is to minimize the total travel time by sending on-ramp flow to the freeway as much as possible without causing congestion. The objective function f is defined as a linear combination of several objective components.
- The first component of the objective is to alleviate the congestion upstream of the work zone by directly penalizing the congested states at the congestion sampling points:

$$\underset{p_k}{\text{Maximize}} \qquad - w_0 \sum_{(t_k, x_q) \in \mathcal{P}} p_k,$$

where w_0 is a weight parameter which can be adjusted.
- The second component of the objective is to maximize the on-ramp flow which is metered by the controller:

$$\underset{q_{b_2}(j)}{\text{Maximize}} \qquad \sum_{j=1}^{j_{max}} w_1(j) q_{b_2}(j).$$

Note that if the weights $w_1(j)$ are the same for all time intervals $j \in \mathcal{J}$, the optimal controller may hold on-ramp flows to later time intervals, which increases the waiting time of vehicles on the on-ramp. Therefore, we assign higher weights to on-ramp flows at earlier time intervals. Specifically, the weights $w_1(j)$ satisfy

(24) $$w_1(j) > \frac{\Delta t_j}{\Delta t_{j+1}} w_1(j+1) \ \forall j \in \mathcal{J} \setminus \{j_{max}\}.$$

- In this example, the downstream boundary flows on the two freeway sections are not controlled. To obtain the unique solution at those two boundaries,

the objective function f must satisfy the conditions (19):

$$(25) \qquad \begin{cases} \frac{\partial f}{\partial q_{b_1}(j)} > \frac{\Delta t_j}{\Delta t_{j+1}} \frac{\partial f}{\partial q_{b_1}(j+1)}, \\ \frac{\partial f}{\partial q_{b_3}(j)} > \frac{\Delta t_j}{\Delta t_{j+1}} \frac{\partial f}{\partial q_{b_3}(j+1)}. \end{cases}$$

The following objective component is added to the objective function to guarantee the unique solution:

$$\underset{q_{b_1}(j), q_{b_3}(j)}{\text{Maximize}} \qquad \sum_{j=1}^{j_{max}} w_2(j) q_{b_1}(j) + \sum_{j=1}^{j_{max}} w_3(j) q_{b_3}(j).$$

It should be noted that p_k relates to $q_{b_1}(j)$ in (23) and minimizing p_k implicitly minimizes $q_{b_1}(j)$. The selection of $w_2(j)$ using conditions (25) should subtract the implicit weight on $q_{b_1}(j)$ induced by penalizing the congested states. In addition, to prevent the on-ramp flows from blocking the upstream freeway flows, the objective function should satisfy

$$(26) \qquad \frac{\partial f}{\partial q_{b_1}(j)} > \frac{\partial f}{\partial q_{b_2}(j)}.$$

In summary, the objective function f is to maximize

$$f = -w_0 \sum_{(t_k, x_q) \in \mathcal{P}} p_k + \sum_{j=1}^{j_{max}} w_1(j) q_{b_2}(j) + \sum_{j=1}^{j_{max}} w_2(j) q_{b_1}(j) + \sum_{j=1}^{j_{max}} w_3(j) q_{b_3}(j).$$

Specifically in this simulation, we select $w_0 = 1$, $w_1(j_{max}) = \Delta t_{j_{max}}$, and other weights according to conditions (24), (25), (26). The weights w_0 and w_1 can be adjusted to balance the congestion on the highway and the queue on the on-ramp.

4.5. Simulation results. A one-hour time horizon was simulated in AIMSUN for the optimal on-ramp metering control. A scenario with uncontrolled on-ramp meter was also simulated for comparison. The simulation result is shown in Figure 5, where Figure 5a plots the traffic states for the uncontrolled scenario and Figure 5b plots for the optimal on-ramp metering control scenario.

As shown in Figure 5a, the on-ramp flow is not controlled and all on-ramp traffic merges to the downstream freeway. On the downstream freeway, the work zone reduced the road capacity and caused severe congestion which could cause safety issues. In comparison in Figure 5b, the optimal on-ramp metering controller regulated the on-ramp traffic to the downstream freeway such that no severe congestion formed. The additional delay time of the vehicles waiting on the on-ramp is compensated by the shorter travel time on the uncongested downstream link.

In summary, this section demonstrated the feasibility of reformulating the convex optimization scheme to optimal traffic control applications. The general idea is to relax the junction models encoded by the conditions on the objective function in Definition 6. A variety of objectives, such as maximizing the boundary flow, or penalizing the congested states can be directly formulated in the objective function.

5. Conclusion. This article proposed a numerical scheme which can compute the traffic evolution modeled by HJ PDEs on a network using a convex optimization program, which could also be applied for optimal control.
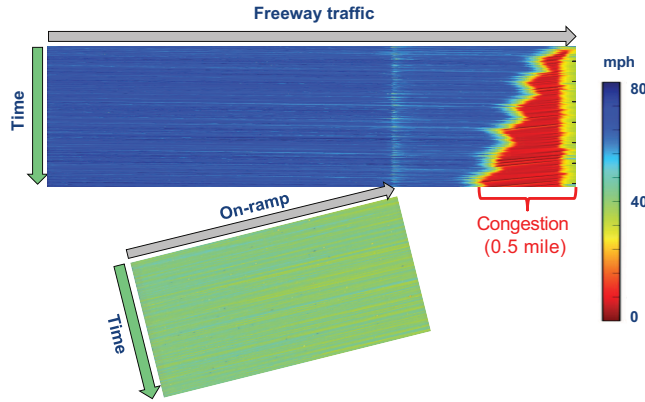
(a) Uncontrolled on-ramp. On-ramp flows caused severe congestion on the downstream freeway link before the work zone bottleneck.
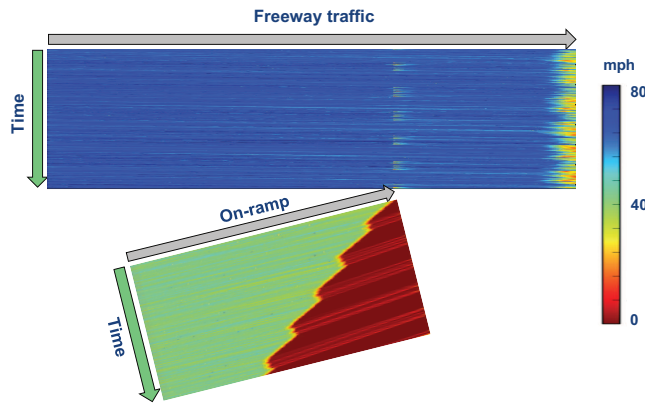
(b) Optimal on-ramp control. On-ramp flows were restricted to prevent the formation of congestion on the downstream freeway link. Meanwhile, on-ramp flows were admitted to the freeway when there was space on the downstream freeway link. Overall, no severe congestion was generated on the downstream freeway link except for slightly slower traffic due to the reduced speed limit and merging activities upstream of the work zone.

FIG. 5. *True speed states on the freeway and the on-ramp. Grey arrows denote the freeway and on-ramp links. Green arrow denote the time (1 hr). (a) Uncontrolled on-ramp. (b) Optimally controlled on-ramp.*

The proposed framework relies on a semi-explicit single link HJ PDE solver, and does not require discretization of the time-space domain. In addition, it computes the internal boundary flows at a merge, or diverge, or connection over the entire time horizon using a single convex program. The convex optimization scheme provides a natural framework for optimal traffic control applications which is demonstrated in a work zone on-ramp metering control example.

REFERENCES

[1] *Highway Capacity Manual* 2010, Transportation Research Board, National Research Council, Washington, DC, 2010, https://doi.org/10.5555/9780309160773.

[2] J. P. AUBIN, A. M. BAYEN, AND P. SAINT-PIERRE, *Dirichlet problems for some Hamilton–Jacobi equations with inequality constraints*, SIAM J. Control Optim., 47 (2008), pp. 2348–2380, https://doi.org/10.1137/060659569.

[3] G. BRETTI, R. NATALINI, AND B. PICCOLI, *Numerical approximations of a traffic flow model on networks*, Netw. Heterog. Media, 1 (2006), pp. 57–84, https://doi.org/10.3934/nhm.2006.1.57.

[4] E. F. CAMACHO AND C. BORDONS, *Model Predictive Control in the Process Industry*, Springer Science & Business Media, Secaucus, NJ, 2012, https://doi.org/10.1007/978-1-4471-3008-6.

[5] C. G. CLAUDEL AND A. M. BAYEN, *Lax–Hopf based incorporation of internal boundary conditions into Hamilton-Jacobi equation. Part* I: *Theory*, IEEE Trans. Automat. Control, 55 (2010), pp. 1142–1157, https://doi.org/10.1109/TAC.2010.2041976.

[6] C. G. CLAUDEL AND A. M. BAYEN, *Lax–Hopf based incorporation of internal boundary conditions into Hamilton-Jacobi equation. Part* II: *Computational methods*, IEEE Trans. Automat. Control, 55 (2010), pp. 1158–1174, https://doi.org/10.1109/TAC.2010.2045439.

[7] C. G. CLAUDEL AND A. M. BAYEN, *Convex formulations of data assimilation problems for a class of Hamilton-Jacobi equations*, SIAM J. Control Optim., 49 (2011), pp. 383–402, https://doi.org/10.1137/090778754.

[8] G. COSTESEQUE AND J. P. LEBACQUE, *Discussion about traffic junction modelling: Conservation laws vs Hamilton–Jacobi equations*, Discrete Contin. Dyn. Syst. Ser. S, 7 (2014), pp. 411–433, https://doi.org/10.3934/dcdss.2014.7.411.

[9] C. F. DAGANZO, *The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory*, Transport. Res. B: Methodological, 28 (1994), pp. 269–287, https://doi.org/10.1016/0191-2615(94)90002-7.

[10] C. F. DAGANZO, *The cell transmission model, part* II: *Network traffic*, Transport. Res. B: Methodological, 29 (1995), pp. 79–93, https://doi.org/10.1016/0191-2615(94)00022-R.

[11] C. F. DAGANZO, *A variational formulation of kinematic waves: Basic theory and complex boundary conditions*, Transport. Res. B: Methodological, 39 (2005), pp. 187–196, https://doi.org/10.1016/j.trb.2004.04.003.

[12] C. F. DAGANZO, *A variational formulation of kinematic waves: Solution methods*, Transport. Res. B: Methodological, 39 (2005), pp. 934–950, https://doi.org/10.1016/j.trb.2004.05.003.

[13] C. F. DAGANZO, *On the variational theory of traffic flow: Well-posedness, duality and applications*, Netw. Heterog. Media, 1 (2006), pp. 601–619, https://doi.org/10.3934/nhm.2006.1.601.

[14] G. DERVISOGLU, G. GOMES, J. KWON, R. HOROWITZ, AND P. VARAIYA, *Automatic calibration of the fundamental diagram and empirical observations on capacity*, in Proceedings of the Transportation Research Board 88th Annual Meeting, no. 09-3159, 2009.

[15] M. GARAVELLO AND B. PICCOLI, *Traffic Flow on Networks*, American Institute of Mathematical Sciences, Springfield, MO, 2006.

[16] C. E. GARCIA, D. M. PRETT, AND M. MORARI, *Model predictive control: Theory and practice – a survey*, Automatica, 25 (1989), pp. 335–348, https://doi.org/10.1016/0005-1098(89)90002-2.

[17] S. K. GODUNOV, *A difference method for numerical calculation of discontinuous solutions of the equations of hydrodynamics*, Mat. Sb. (N.S.), 89 (1959), pp. 271–306.

[18] K. HAN, B. PICCOLI, AND W. Y. SZETO, *Continuous-time link-based kinematic wave model: Formulation, solution existence, and well-posedness*, Transportmetrica B: Transport Dynamics, 4 (2016), pp. 187–222, https://doi.org/10.1080/21680566.2015.1064793.

[19] M. HERTY AND A. KLAR, *Modeling, simulation, and optimization of traffic flow networks*, SIAM J. Sci. Comput., 25 (2003), pp. 1066–1087, https://doi.org/10.1137/S106482750241459X.

[20] H. HOLDEN AND N. H. RISEBRO, *A mathematical model of traffic flow on a network of unidirectional roads*, SIAM J. Math. Anal., 26 (1995), pp. 999–1017, https://doi.org/10.1137/S0036141093243289.

[21] C. IMBERT, R. MONNEAU, AND H. ZIDANI, *A Hamilton–Jacobi approach to junction problems and application to traffic flows*, ESAIM Control Optim. Calc. Var., 19 (2013), pp. 129–166.

[22] W. JIN, *Continuous kinematic wave models of merging traffic flow*, Transport. Res. B: Methodological, 44 (2010), pp. 1084–1103, https://doi.org/10.1016/j.trb.2010.02.011.

[23] W. JIN, *A Riemann solver for a system of hyperbolic conservation laws at a general road junction*, Transport. Res. B: Methodological, 98 (2017), pp. 21–41, https://doi.org/10.1016/j.trb.2016.12.007.

[24] W. JIN AND H. M. ZHANG, *On the distribution schemes for determining flows through a merge*, Transport. Res. B: Methodological, 37 (2003), pp. 521–540, https://doi.org/10.1016/S0191-2615(02)00026-7.

[25] J. P. LEBACQUE, *The Godunov scheme and what it means for first order traffic flow models*, in Proceedings of the International Symposium on Transportation and Traffic Theory, 1996, pp. 647–677.

[26] J. P. LEBACQUE, *First-order macroscopic traffic flow models: Intersection modeling, network modeling*, in Proceedings of the 16th International Symposium on Transportation and Traffic Theory, 2005, pp. 365–386.

[27] J. P. LEBACQUE, *Intersection modeling, application to macroscopic network traffic flow models and traffic management*, in Traffic and Granular Flow, Springer-Verlag, Berlin, Heidelberg, 2005, pp. 261–278, https://doi.org/10.1007/3-540-28091-X_26.

[28] Y. LI, E. CANEPA, AND C. CLAUDEL, *Efficient robust control of first order scalar conservation laws using semi-analytical solutions*, Discrete Contin. Dyn. Syst. Ser. S, 7 (2014), pp. 525–542, https://doi.org/10.3934/dcdss.2014.7.525.

[29] Y. LI, E. CANEPA, AND C. G. CLAUDEL, *Optimal control of scalar conservation laws using linear/quadratic programming: Application to transportation networks*, IEEE Trans. Control Netw. Syst., 1 (2014), pp. 28–39, https://doi.org/10.1109/TCNS.2014.2304152.

[30] Y. LI, C. G. CLAUDEL, B. PICCOLI, AND D. B. WORK, *A Convex Formulation of Traffic Dynamics on Transportation Networks*, preprint, https://arxiv.org/abs/1702.03908, 2017.

[31] M. LIGHTHILL AND G. WHITHAM, *On kinematic waves. II. A theory of traffic flow on long crowded roads*, Proc. Roy. Soc. London. Ser. A, 229 (1999), pp. 317–345, https://doi.org/10.1098/rspa.1955.0089.

[32] J. M. MACIEJOWSKI, *Predictive Control: With Constraints*, Pearson Education, Prentice-Hall, London, 2002.

[33] P. MAZARÉ, A. DEHWAH, C. G. CLAUDEL, AND A. M. BAYEN, *Analytical and grid-free solutions to the Lighthill-Whitham-Richards traffic flow model*, Transport. Res. B: Methodological, 45 (2011), pp. 1727–1748, https://doi.org/10.1016/j.trb.2011.07.004.

[34] K. MOSKOWITZ, *Discussion of freeway level of service as influenced by volume and capacity characteristics*, Highway Research Record, 99 (1965), pp. 43–44.

[35] G. F. NEWELL, *A simplified theory of kinematic waves in highway traffic, part* I: *General theory*, Transport. Res. B: Methodological, 27 (1993), pp. 281–287, https://doi.org/10.1016/0191-2615(93)90038-C.

[36] G. F. NEWELL, *A simplified theory of kinematic waves in highway traffic, part* II: *Queueing at freeway bottlenecks*, Transport. Res. Part B: Methodological, 27 (1993), pp. 289–303, https://doi.org/10.1016/0191-2615(93)90039-D.

[37] G. F. NEWELL, *A simplified theory of kinematic waves in highway traffic, part* III: *Multi-destination flows*, Transport. Res. Part B: Methodological, 27 (1993), pp. 305–313, https://doi.org/10.1016/0191-2615(93)90040-H.

[38] P. RICHARDS, *Shock waves on the highway*, Operations Res., 4 (1956), pp. 42–51, https://doi.org/10.1287/opre.4.1.42.

[39] X. SUN AND R. HOROWITZ, *Localized switching ramp-metering control with queue length estimation and regulation and microscopic simulation results*, IFAC Proceedings Volumes, 38 (2005), pp. 156–161, https://doi.org/10.3182/20050703-6-CZ-1902.02049.