Traffic State Estimation for Connected Vehicles using a Second Order Traffic Model

Suyash C. Vishnoi[a,*], Sebastian A. Nugroho[b], Ahmad F. Taha[c], Christian G. Claudel[a]

[a] *Department of Civil, Architectural, and Environmental Engineering, The University of Texas at Austin, 301 E. Dean Keeton St. Stop C1700, Austin, TX 78712.*
[b] *Department of Electrical Engineering and Computer Science, University of Michigan, 1301 Beal Ave., Ann Arbor, MI 48109.*
[c] *Department of Civil and Environmental Engineering, Vanderbilt University, 2201 West End Ave, Nashville, TN 37235.*

Abstract

This paper addresses the problem of traffic state estimation (TSE) in the presence of heterogeneous sensors which include both fixed and moving sensors. Traditional fixed sensors are expensive and cannot be installed throughout the highway. Moving sensors such as Connected Vehicles (CVs) offer a relatively cheap alternative to measure traffic states across the network. Moving forward it is thus important to develop such models that effectively use the data from CVs. One such model is the nonlinear second-order Aw-Rascle-Zhang (ARZ) model which is a realistic traffic model, reliable for TSE and control. A state-space formulation is presented for the ARZ model considering junctions in the formulation which is important to model real highways with ramps. Linear approximation of the state-space model is investigated with respect to two techniques, first-order Taylor series approximation and Carleman linearization. A Moving Horizon Estimation (MHE) implementation is presented for TSE using a linearized ARZ model. Various state-estimation techniques used for TSE in the literature along with the presented approach are compared with regard to accuracy, computational tractability and parameter tuning with the help of

*Corresponding author
Email addresses: scvishnoi@utexas.edu (Suyash C. Vishnoi), snugroho@umich.edu (Sebastian A. Nugroho), ahmad.taha@vanderbilt.edu (Ahmad F. Taha), christian.claudel@utexas.edu (Christian G. Claudel)

a case study using the VISSIM traffic simulation software. Several research questions are posed and addressed with thorough analysis of the results.
Keywords: Traffic state estimation, highway traffic networks, second-order models, Aw-Rascle-Zhang model, Moving Horizon Estimation, connected vehicles.

1. Motivation and Paper Contributions

With the large number of vehicles overloading the transportation systems across the world, problems like congestion, accidents, and pollution have become common. As a remedy to such circumstances, control techniques such as variable speed limits, ramp metering, route control and their combinations have become quite popular for instance see [1, 2, 3, 4].

These techniques require the knowledge of the system at all times to make them work effectively. A popular method for real-time monitoring of traffic systems is by means of traffic state estimation (TSE) using dynamic traffic models which provide a high-fidelity picture of the traffic spatio-temporally while utilizing data from sensors available throughout the highway. In general, more data results in better estimates of the system states. However, since fixed sensors like inductive loop detectors are quite expensive, they cannot be placed at short intervals throughout the highway. Connected vehicles (CVs) offer a potential solution to this problem by providing additional sources of data relatively free of cost [5]. Here, we assume that most of the communication between the CVs and the network operator will take place via existing cellular networks so there will be no additional costs of building connected highway infrastructure everywhere. As the proportion of CVs in the traffic rises, CVs will be able to provide useful data from across the system. Thus, moving forward, it is imperative to develop such models that can utilize well the data from both fixed sensors and CVs to perform state estimation and control.

Traditionally, TSE is performed using first-order traffic models such as the Lighthill-Whitham-Richards (LWR) model [6, 7]. First-order models are simple to implement as they only have a single equation that is the conservation of vehicles to describe the traffic dynamics. They also have very few calibration parameters, making them a popular choice for state estimation. However, they only consider equilibrium traffic conditions, that is, the traffic density (vehicles per unit distance) and traffic flux (vehicles per unit time) are assumed to follow a predefined relationship known as the *fundamental diagram*. This makes them unable to represent certain non-equilibrium traffic phenomena like capacity drop which are essential for the purpose of traffic control [8]. Thus, the use of these models in traffic control is considered less effective.

Second-order traffic models such as the Payne-Whitham (PW) model [9, 10] and the Aw-Rascle-Zhang (ARZ) model [11, 12], on the other hand, can represent non-equilibrium traffic phenomena with the help of an additional equation to describe the traffic dynamics. They are, therefore, considered more realistic than the first-order models. As a result, these models are not only good for state estimation but are also reliable for control. Additionally, second-order models provide a natural way to incorporate multiple sources of data as they consider both density and speed to be independent variables. In first-order models which only consider either the density or the speed as a variable at a time, any deviation of the speed from its equilibrium relationship must be considered a part of the modeling error. Thus, second-order models become a natural choice for state estimation using CV data. Note that while Lagrangian (vehicle-based) models of traffic exist [13] which are arguably more compatible with trajectory based data from CVs, here we are using road density and average vehicle speed information obtained using both CVs and fixed detectors and not just relying on trajectory based information. Therefore, we have chosen a Eulerian (location-based) second-order model over a Lagrangian model besides the above reasons.

In light of the aforementioned discussion, the objective of this work is to develop a state-space representation of a reliable second-order traffic model and further investigate various state estimation techniques for the purpose of TSE in the presence of heterogeneous traffic sensors including CVs. Given this objective, in the following discussion, we present a literature review of traffic models used for TSE followed by a brief discussion on the estimation techniques used.

The most popular model for TSE in the literature is the first-order LWR model. The simple form of the model with a minimal number of calibration parameters makes it an attractive option for large scale implementation. Some works that implement a first-order model to perform state estimation using heterogeneous sensors include [14, 15, 16]. Readers can also refer to [17] for a comprehensive review of TSE literature involving first-order models. Due to the known limitations of first-order models, several studies have also undertaken state estimation using second-order models such as in [18, 19, 20] and the references therein. Most of these studies use the second-order PW model implemented in the METANET [21, 22] framework. The PW model has well known limitations [8] such as physical inconsistency under certain heterogeneous traffic conditions which make it unreliable. A significantly better model is the ARZ model which retains the benefits of second-order models without sacrificing the physical consistency of the first-order models. Despite this, there are very few studies in the literature that use the ARZ model for state estimation. The work in [23] develops a state-space formulation for the nonlinear ARZ model and performs state estimation using Extended Kalman Filter (EKF) considering both fixed and moving sensors. In [24], the authors propose a boundary observer for state estimation using a linearized ARZ model. The study in [25] uses Particle Filter (PF) for estimation of traffic states using a modified ARZ model. However, it is worth noting that none of these papers considers junctions in the modeling. Modeling the traffic dynamics at junctions is essential to the modeling of traffic on real highways which consist of on-ramp and off-ramp connections. Therefore, unlike past studies, we formulate herein a state-space model for the nonlinear ARZ model considering junctions. Note that the aforementioned studies using second-order models as well as the present work are different from studies like [26] which while do consider the speed to be an independent variable like the second-order models but consider it to be known at all times using CV data. These have been categorized as *data-driven* methods by

[17]. In this work, unlike [26], we do not assume to have speed data from every part of the road and speed is still a variable to be estimated for the unmeasured segments.

A majority of the TSE literature either uses one of the Kalman Filter (KF) variants from among EKF, Unscented Kalman Filter (UKF), and Ensemble Kalman Filter (EnKF), or other techniques like PF, and observers to perform state estimation, for instance see [17, 27, 28]. While these methods are computationally attractive, they have certain limitations with respect to TSE. The primary limitation is that they do not have an inherent way to deal with state constraints. Thus, it is possible that the estimates generated from these methods contain nonphysical values of certain states which can further cause the process model to collapse. Besides, these methods do not naturally handle arbitrary constraints such as privacy constraints like those which forbid the state estimator to use data of certain CVs for estimation due to privacy reasons if the collection of such data is otherwise unavoidable.

An estimation technique which handles these limitations naturally, due to its optimization-based structure, is Moving Horizon Estimation (MHE). MHE has been explored extensively in the general state estimation literature, for instance in [29, 30, 31, 32], but not so much in the TSE literature. In [33] and [34], the authors propose an MHE formulation for estimation and control of large scale highway networks using the Macroscopic fundamental diagram (MFD). MFD is a network level traffic model and does not consider the variation in traffic density on individual stretches of the highway. Unlike [33] and [34], we investigate estimating the density throughout the highway stretch. The study in [35] presents an MHE formulation for traffic density estimation using the Asymmetric Cell Transmission model (ACTM). ACTM is based on the LWR model and therefore, has the drawbacks previously mentioned for first-order models. Moreover, the work in [35] does not consider moving sensors from CVs.

Given that, the main research gaps on this topic are: a) the absence of a state-space formulation for a reliable second-order traffic model with junctions, b) the lack of exploration of MHE in the context of TSE, and c) the absence of a comparative study between different state-estimation techniques for TSE using a second-order model in the presence of heterogeneous data. In what follows, we highlight the main contributions of this paper:

- We derive a nonlinear state-space formulation for the second-order ARZ model with junctions in the form of ramp-connections. In that, we present the detailed dynamic equations of the model. This is a development over [23] which does not consider junctions in the formulation. Addition of junctions adds additional complexity to the model in terms of the nonlinearity which now comprises of minimum and piecewise functions in the model. Second-order traffic models are more realistic than first-order models like the LWR model as they can capture certain phenomena like capacity drop which are essential to control applications. The obtained state-space formulation can thus be used for state estimation as well as control purposes.

- We consider heterogeneous sensors including both fixed and moving sensors. The former consists of sensors like inductive loop detectors while the latter includes CVs. The state-space description is appended to include the measurement model which is also nonlinear thus resulting in a nonlinear input-output mapping of the system dynamics.

- Since the model is indeed large scale due to modeling large highways with several ramps, there is a need for scalable state estimation methods. One way of achieving this is by linearizing the nonlinear model and applying linear state estimation techniques to it. We investigate the accuracy of some linear approximation techniques in approximating the nonlinear dynamics of traffic states in the ARZ model. In particular, we compare the performance of Taylor series and Carleman linearization [36] techniques of different orders. With the linearization of the ARZ model, it is also possible to use it in Model Predictive Control (MPC) [37] frameworks to perform real-time traffic control using for instance variable speed limit control or ramp metering based control [38] among others.

- We investigate the performance of various state estimation techniques in terms of accuracy, computational tractability and parameter tuning using the VISSIM traffic simulation software. As a departure from estimation based on KFs, PF, observers and so on, we investigate MHE for TSE. MHE, unlike the other techniques, allows us to include different constraints on the state variables making the problem more practical.

Paper's Notation: Let $\mathbb{N}$, $\mathbb{R}$, $\mathbb{R}^n$, and $\mathbb{R}^{p\times q}$ denote the set of natural numbers, real numbers, and real-valued row vectors with size of n, and p-by-q real matrices respectively. $\mathbb{S}^m_{++}$ denotes the set of positive definite matrices. For any vector $z \in \mathbb{R}^n$, $\|z\|_2$ denotes its Euclidean norm, i.e. $\|z\|_2 = \sqrt{z^\top z}$, where $z^\top$ is the transpose of z.

The symbol $\otimes$ denotes the Kronecker product. Tab. 1 provides the nomenclature utilized in this paper.

2. Nonlinear Discrete-Time Modeling of Traffic Networks with ramps

The objective of this section is to develop a state-space formulation for the nonlinear second-order ARZ model describing the evolution of traffic density on highways with ramps. The developed formulation is useful for several control theoretic purposes including state estimation and control of highway traffic.

2.1. The Aw-Rascle-Zhang model

In this section, we present the modeling of traffic dynamics for a stretched highway connected with ramps. To that end, we use the the second-order ARZ Model [11, 12] given by the following partial differential equations:

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho v}{\partial d} = 0, \tag{1a}$$

$$\frac{\partial \rho\left(v + p\left(\rho\right)\right)}{\partial t} + \frac{\partial \rho\left(v + p\left(\rho\right)\right) v}{\partial d} = -\frac{\rho\left(v - V_e\left(\rho\right)\right)}{\tau}, \tag{1b}$$

where t and d denote the time and distance; ρ is shorthand for $\rho(t, d)$ which denotes the traffic density (vehicles/distance), and v is shorthand for $v(t, d)$

Table 1: Paper nomenclature: parameter, variable, and set definitions.

Notation	**Description**
Ω	the set of highway segments on the stretched highway $\Omega = \{1, 2, \ldots, N\}$, $N := \lvert\Omega\rvert$
Ω_I	the set of highway segments with on-ramps $\Omega_I = \{1, 2, \ldots, N_I\}$, $N_I := \lvert\Omega_I\rvert$
Ω_O	the set of highway segments with off-ramps $\Omega_O = \{1, 2, \ldots, N_O\}$, $N_O := \lvert\Omega_O\rvert$
$\hat{\Omega}$	the set of on-ramps, $\hat{\Omega} = \{1, 2, \ldots, N_I\}$, $N_I = \lvert\hat{\Omega}\rvert$
$\check{\Omega}$	the set of off-ramps, $\check{\Omega} = \{1, 2, \ldots, N_O\}$, $N_O = \lvert\check{\Omega}\rvert$
T	duration of each time-step
l	length of each segment, on-ramp, and off-ramp
$\rho_i[k], \psi_i[k], w_i[k]$	traffic density, relative flow and driver characteristic for Segment $i \in \Omega$ at time kT, $k \in \mathbb{N}$
$q_i[k], \phi_i[k]$	traffic flow and relative flux from Segment $i \in \Omega$ into the next segment
$D_i[k], S_i[k]$	demand and supply functions for Segment $i \in \Omega$
$\hat{\rho}_i[k], \hat{\psi}_i[k], \hat{w}_i[k]$	traffic density, relative flow and driver characteristic for On-ramp $i \in \hat{\Omega}$ at time kT, $k \in \mathbb{N}$
$\hat{q}_i[k], \hat{\phi}_i[k]$	traffic flow and relative flux from On-ramp $i \in \hat{\Omega}$ into the attached highway segment
$\hat{D}_i[k], \hat{S}_i[k]$	demand and supply functions for On-ramp $i \in \hat{\Omega}$
$\check{\rho}_i[k], \check{\psi}_i[k], \check{w}_i[k]$	traffic density, relative flow and driver characteristic for Off-ramp $i \in \check{\Omega}$ at time kT, $k \in \mathbb{N}$
$\check{q}_i[k], \check{\phi}_i[k]$	traffic flow and relative flux from Off-ramp $i \in \check{\Omega}$
$\check{D}_i[k], \check{S}_i[k]$	demand and supply functions for Off-ramp $i \in \check{\Omega}$
$\bar{q}_i[k], \bar{\phi}_i[k]$	incoming traffic flow and relative flux for Segment $i \in \Omega$
$\bar{\check{q}}_i[k], \bar{\check{\phi}}_i[k]$	incoming traffic flow and traffic flux for Off-ramp $i \in \check{\Omega}$
$D_{in}[k], w_{in}[k]$	demand and driver characteristic of traffic wanting to enter Segment 1 of the highway
$\rho_{out}[k]$	traffic density downstream of Segment N of the highway
$\hat{D}_{in,i}[k], \hat{w}_{in,i}[k]$	demand and driver characteristic of traffic wanting to enter On-ramp $i \in \hat{\Omega}$
$\check{\rho}_{out,i}[k]$	traffic density downstream of Off-ramp $i \in \check{\Omega}$
$\beta_i[k]$	proportion of traffic entering from Segment $i \in \Omega$ into the next segment at an on-ramp junction, where $\beta_i[k] \in [0, 1]$
$\alpha_i[k]$	split ratio for the off-ramp attached to Segment $i \in \Omega$, where $\alpha_i[k] \in [0, 1]$
v_f	free-flow speed
ρ_m	maximum density
α	model parameter called relaxation time, where $\alpha \in \mathbb{R}_+$
γ	fundamental diagram parameter, where $\gamma \in \mathbb{R}_+$
$p(\rho)$	pressure function which takes traffic density ρ as input
$V_e(\rho)$	equilibrium traffic speed at traffic density ρ

which denotes the traffic speed (distance/time). Here, $p(\rho)$ is given by

$$p\left(\rho\right)=v_f\left(\frac{\rho}{\rho_m}\right)^{\gamma}, \tag{2}$$

and $V_e(\rho)$ is given by

$$V_e(\rho)=v_f\left(1-\left(\frac{\rho}{\rho_m}\right)^{\gamma}\right). \tag{3}$$

In traffic literature, relationships like (3) are commonly called the *fundamental diagram*. The first PDE in the ARZ model ensures the conservation of vehicles which is also present in the first-order traffic models. The second PDE which ensures conservation of traffic momentum is unique to second-order models and accounts for the deviation of traffic from an equilibrium position. This equation makes the second-order models more realistic than the first order models as it allows them to represent some non-equilibrium traffic phenomena such as capacity drop. As second-order models allow traffic flow to deviate from equilibrium, they also inherently allow traffic speed to deviate from the equilibrium speed which allows speed data to be incorporated independent of the density. With first-order models, any deviation of the speed from the equilibrium speed would have to be considered a part of modeling error. Therefore, second-order models are more naturally suited to perform estimation using both density and speed data provided by the fixed sensors and CVs. The quantity $v+p(\rho)$ is also called the *driver characteristic* and is denoted by the variable $w(t,d)$. The expression $\rho(v+p(\rho))$ is also called the *relative flow* denoted by $\psi(t,d)$ which is essentially the difference between the actual flow and the equilibrium flow at any ρ. Notice that in (1), ρv is the flux of traffic (vehicles/time) which will be denoted by $q(t,d)$, while $\rho(v+p(\rho))v$ is the flux of relative flow (vehicles/time^2), also called the relative flux, which will be denoted by $\phi(t,d)$. Using the relative flow and the two flux, the ARZ model can simply be rewritten as

$$\frac{\partial\rho(t,d)}{\partial t}+\frac{q(t,d)}{\partial d}=0, \tag{4a}$$

$$\frac{\partial\psi(t,d)}{\partial t}+\frac{\partial\phi(t,d)}{\partial d}=-\frac{\psi(t,d)}{\tau}+\frac{v_f\rho(t,d)}{\tau}, \tag{4b}$$

which can be converted to a state-space equation with ρ and y as the states.

To represent this model as a series of difference, state-space equations, we discretize the ARZ Model (4) with respect to both space and time, also referred to as the Godunov scheme [39]. This allows us to divide the highway of length L into segments of equal length l and the traffic networks model to be represented by discrete-time equations. These segments form both the highway and the attached ramps. Throughout the paper, the segments forming the highway are referred to as mainline segments. We assume the highway is split into N mainline segments.

To ensure computational stability, the Courant-Friedrichs-Lewy condition (CFL) [40] given as $v_f T l^{-1} \leq 1$ has to be satisfied. Since each segment is of the same length l, then we have $\rho(t, d) = \rho(kT, l)$, where $k \in \mathbb{N}$ represents the discrete-time index. For simplicity, we define $\rho(kT, l) := \rho[k]$. The other variables are also defined in the same way, namely $w[k], \psi[k], q[k]$ and $\phi[k]$. The expressions for the flux functions $q[k]$ and $\phi[k]$ for any segment depend on the arrangement of the segments before and after that segment. In what follows, we define the flux expressions for different types of segment junctions, but before that, we need to define two other functions called the demand and supply functions which are given below.

2.2. Demand and supply functions

The demand of a segment denotes the traffic flux that wants to leave that segment while the supply of a segment denotes the traffic flux that can enter that segment. Based on these definitions, the demand $D_i[k]$ for Segment i can be written, similar to [41, 42], as

$$D_i[k]=\begin{cases}\rho_i(w_i[k]-p(\rho_i[k])), & \text{if } \rho_i[k] \leq \sigma(w_i[k]),\\ \sigma(w_i[k])(w_i[k]-p(\sigma(w_i[k]))), & \text{if } \rho_i[k] > \sigma(w_i[k]),\end{cases}$$

Here, $\sigma(w_i[k])$ denotes the density that maximizes the demand function and is given as

$$\sigma_i(w[k]) = \rho_m \left(\frac{w_i[k]}{v_f(1+\gamma)}\right)^{\frac{1}{\gamma}}.$$

The supply function $S_i[k]$ on the other hand is given by

$$S_i[k]=\begin{cases}\sigma(w_i[k])(w[k]-p(\sigma(w[k]))), & \text{if } \rho_i[k]\leq\sigma(w[k]),\\ \rho_i(w[k]-p(\rho_i[k])), & \text{if } \rho_i[k]>\sigma(w[k]).\end{cases} \tag{5}$$

Notice that $w[k]$ used in (5) does not belong to Segment i. Instead, it is calculated from the $\rho[k]$ and $y[k]$ of the incoming traffic upstream of Segment i. The exact method of calculating this $w[k]$ is given in the following section. Next, we define the expressions for the flux functions $q[k]$ and $\phi[k]$.

2.3. Flux formulae at junctions

This section presents the expressions for the traffic flux $q[k]$ and the relative flux $\phi[k]$, which form the nonlinear part of the state-space model. Development of analytical equations for junction flows in the ARZ and other similar second-order models is an active field of research [43, 42, 41], with different papers providing different approaches to model the junction flows, some more complex than the others. However, since state estimation allows for some extent of modeling errors, it is possible to develop a simple state-space formulation for the ARZ model without going into intractable schemes. We consider three types of segment junctions, a one-to-one junction between two mainline segments, a merge junction between two mainline segments and an on-ramp, and a diverge junction between two mainline segments and an off-ramp. In the following discussion, we assume that the mainline segment before the junction has index $i \in \Omega$, the segment after the junction has index $i+1$ and the ramp has index $j \in \hat{\Omega}$ for on-ramp and $j \in \check{\Omega}$ for off-ramp.

2.3.1. One-to-one junction

The traffic flux leaving Segment i and entering Segment $i+1$ at a one-to-one junction is given as

$$q_i[k] = \min(D_i[k], S_{i+1}[k])),$$

while the relative flux is given as

$$\phi_i[k] = q_i[k]w_i[k] = q_i[k]\frac{\psi_i[k]}{\rho_i[k]}.$$

2.3.2. Merge junction (on-ramp connection)

At a merge junction, we have that $\bar{q}_{i+1}[k] = q_i[k] + \hat{q}_j[k]$. We assume that the flow entering Segment $i+1$ from each of the incoming segments is in the proportion of their demands, that is, if

$$\beta_i[k] = \frac{D_i[k]}{D_i[k] + \hat{D}_j[k]},$$

then

$$q_i[k] = \beta_i[k]\bar{q}_{i+1}[k], \tag{6a}$$

$$\hat{q}_j[k] = (1 - \beta_i[k])\bar{q}_{i+1}[k]. \tag{6b}$$

In case of a merge junction, the $w[k]$ used to calculate the supply for the outgoing segment using (5) is denoted as $\bar{w}[k]$ and is calculated as

$$\bar{w}[k] = \beta_i[k]w_i[k] + (1 - \beta_i[k])\hat{w}_j[k].$$

Then the traffic flux leaving Segment i is given by

$$q_i[k] = \min(\beta_i[k]S_{i+1}[k], D_i[k], \frac{\beta_i[k]}{1 - \beta_i[k]}\hat{D}_j[k]).$$

$\bar{q}_{i+1}[k]$ and $\hat{q}_j[k]$ can thereafter be calculated using (6). The relative flux entering Segment $i+1$ is given as

$$\bar{\phi}_{i+1}[k] = \bar{q}_{i+1}[k]\bar{w}[k],$$

and those exiting the incoming segments are given by

$$\phi_i[k] = q_i[k]w_i[k],$$
$$\hat{\phi}_j[k] = \hat{q}_j[k]\hat{w}_j[k].$$

2.3.3. Diverge junction (off-ramp connection)

At diverge junctions, we have that $q_i[k] = \bar{\hat{q}}_j[k] + \bar{q}_{i+1}[k]$. We assume that the proportion of the flow entering the Off-ramp j from Segment i is given by a predefined constant $\alpha_i[k]$, such that

$$\bar{\hat{q}}_j[k] = \alpha_i[k]q_i[k],$$

$$\bar{q}_{i+1}[k] = (1 - \alpha_i[k])q_i[k].$$

In case of a diverge junction, we use $w_i[k]$ to calculate the supply for both the mainline Segment $i+1$ and the Off-ramp j. The flow $q_i[k]$ can then be written as

$$q_i[k] = \min(D_i[k], \frac{\check{S}_j[k]}{\alpha_i[k]}, \frac{S_{i+1}[k]}{(1-\alpha_i[k])}),$$

while the relative flux leaving Segment i is given as

$$\phi_i[k] = q_i[k]w_i[k].$$

The relative flux entering the outgoing segments have the same relationship as the flows, that is

$$\bar{\phi}_j[k] = \alpha_i[k]\phi_i[k],$$
$$\bar{\phi}_{i+1}[k] = (1 - \alpha_i[k])\phi_i[k].$$

2.4. State-space equations

The discrete time traffic flow and relative flow conservation equations for any Segment $i \in \Omega$ can be written as

$$\rho_i[k+1] = \rho_i[k] + \frac{T}{l}(q_{i-1}[k] - q_i[k]), \tag{10a}$$
$$\psi_i[k+1] = \left(1 - \frac{1}{\tau}\right)\psi_i[k] + \frac{T}{l}(\phi_{i-1}[k] - \phi_i[k]) + \frac{v_f}{\tau}\rho_i[k]. \tag{10b}$$

Similar equations can be written for ramp segments as well. Here, $q_i[k]$ and $\phi_i[k]$ take the expressions from the previous section depending upon the arrangement of Segment i with respect to other segments. The state vector for this system can be defined as

$$\boldsymbol{x}[k] := [\rho_i[k]\ \psi_i[k]\ \ldots\ \hat{\rho}_j[k]\ \hat{\psi}_j[k]\ \ldots\ \check{\rho}_l[k]\ \check{\psi}_l[k]\ \ldots]^\top \in \mathbb{R}^{2(N+N_I+N_O)},$$

for which $i \in \Omega$, $j \in \hat{\Omega}$ and $k \in \check{\Omega}$. In this work, we assume that the demand and the driver characteristic upstream of the first mainline segment are known, that is, $D_0[k] = D_{in}[k]$ and $w_0[k] = w_{in}[k]$ and the density downstream of the last mainline segment is also assumed to be known, that is $\rho_{N+1}[k] = \rho_{out}[k]$. Similarly, the demand and driver characteristic upstream of the on-ramps and the density downstream of the off-ramps is also considered to be known. These values can be obtained using conventional detectors like the inductive loop detectors placed upstream of the input segments and downstream of the output segments of the highway. An approximate value of the demand can also be obtained using Origin-Destination flow matrices [44] if available for the given region. Then,

$$\boldsymbol{u}[k] := [D_{in}[k] \; w_{in}[k] \; \rho_{out}[k] \; \ldots \; \hat{D}_{in,j}[k] \; \hat{w}_{in,j}[k] \ldots \; \check{\rho}_{out,l}[k] \; \ldots]^\top \in \mathbb{R}^{3+2N_I+N_O},$$

where $j \in \hat{\Omega}$ and $l \in \check{\Omega}$.

The evolution of traffic density and relative flow described in (10) can be written in a compact state-space form as follows

$$\boxed{\boldsymbol{x}[k+1] = \boldsymbol{A}\boldsymbol{x}[k] + \boldsymbol{G}\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}),} \tag{11}$$

where $\boldsymbol{A} \in \mathbb{R}^{n_x \times n_x}$ for $n_x := 2(N + N_I + N_O)$ represents the linear dynamics of the system, $\boldsymbol{f} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_x}$ for $n_u = 3 + 2N_I + N_O$ is a vector valued function representing nonlinearities in the state-space equation, and $\boldsymbol{G} \in \mathbb{R}^{n_x \times n_x}$ is a matrix representing the distribution of nonlinearities.

The nonlinearities in $\boldsymbol{f}$ are in the form of a minimum of weighted nonlinear functions of the states and inputs. The structure of the above mentioned matrices and functions is provided in Appendix A. Next, we discuss the measurement model for the ARZ model which is also nonlinear in nature.

2.5. Sensor data and measurement model

We consider two types of sensors in this work, first are the fixed sensors like the inductive loop detectors and second are the CVs. This study assumes that it is possible to retrieve density and speed data from both the types of sensors. While it is not possible to obtain density data directly from a single loop detector, two loop detectors installed at opposite ends of a segment can provide us with this information. Here, one loop detector provides us with the vehicle counts entering the segment while the other detector provides the counts leaving that segment. Then, starting from an empty highway segment, the continuous vehicle counts can provide us with the total number of vehicles at any time on a road segment of known length which gives us the traffic density for that segment. This is similar to the approach used in [45] for density calculation. To determine when the segment is empty, one can look at both the loop detectors' readings over time and start to calculate the density from that point onwards. The loop detectors when installed in pairs are already known to provide speed data directly. CVs are known to provide the current position and speed data directly. The average speed of a segment can be assumed to be the average of the speed data provided by all the queried CVs in that segment similar to [46]. To obtain density data from CVs, we assume additional functionality including either spacing measurement equipment which is available as part of advanced driver assistance systems [47] or availability of vehicular ad-hoc networks (VANETs) which allow vehicles to communicate with each other in a neighbourhood around the queried CV [48]. When assuming the latter it is important to note the limitation imposed by the communication range of the vehicles on the maximum cell length for traffic modeling. In case of the former, while a cell length limitation may not be required, a sufficient penetration of CVs is necessary on the segments which are queried for data. The data from the CVs is sent via cellular network to a network operator who performs any prior computation if necessary to convert the received information like the spacing data or neighbourhood counts into density measurements before using them for state estimation. A measurement error can also be associated with the data at this point based on the available information on penetration rate or other factors.

Figure 1 presents a schematic of the sensors' placement on the highway.

Among the measurements, density $\rho_i[k]$ for any mainline segment $i \in \Omega$, and similarly for the ramps, is directly a state and is used as it is, while the velocity

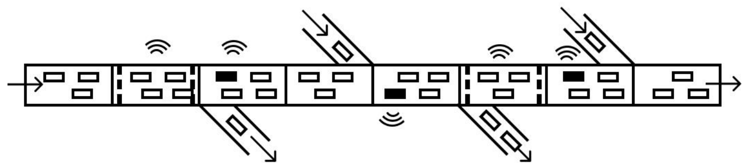

Figure 1: Heterogenous sensors on the highway: fixed sensors represented by dashed lines across the highway and CVs represented by the solid black rectangles.

$v_i[k]$ can be written in terms of the states as follows:

$$v_i[k] = \frac{\psi_i[k]}{\rho_i[k]} - p(\rho_i[k]).$$

We define a nonlinear measurement function $\boldsymbol{h}(\boldsymbol{x}[k])$ such that

$$\begin{aligned} h_{2i-1}(\boldsymbol{x}[k]) &= x_{2i-1}[k], \\ h_{2i}(\boldsymbol{x}[k]) &= \frac{x_{2i}[k]}{x_{2i-1}[k]} - p(x_{2i-1}[k]). \end{aligned}$$

Now, we can define the measurement vector $\boldsymbol{y}[k]$ as

$$\boxed{\boldsymbol{y}[k] = \boldsymbol{C}[k]\boldsymbol{h}(\boldsymbol{x}[k]) + \boldsymbol{\nu}[k],}$$

where $\boldsymbol{C}[k]$ is the observation matrix at time k describing the availability of measurements from sensors. Note, that the observation matrix here is variable in time because of the measurements from connected vehicles which keep changing their location with time. At any time k, $n_p[k]$ is the number of measurements. Here, $\boldsymbol{\nu}[k] \in \mathbb{R}^{n_\nu[k]}, n_\nu[k] = n_p[k]$ lumps all the measurement errors including the sensor noise into a single vector.

The above results are important as they allow us to perform state estimation for traffic systems using the second-order ARZ model. The state-space equation (11) can also be used for control purposes using control theoretic approaches from the literature. In the following section, we discuss some methods for linearization of nonlinear functions which allow us to apply some linear state estimation techniques to the otherwise nonlinear ARZ model.

3. Linear Model Approximation

The ARZ model specified in Section 2 is nonlinear due to the presence of the piecewise linear and nonlinear expressions in the traffic flux and relative flux terms. This prevents from directly using some of the well-known and efficient linear state estimation techniques from the literature. However, it is still possible to apply linear state estimation techniques to a linearized version of the ARZ model. Techniques such as Taylor series expansion [49] and Carleman Linearization [36] allow to obtain good linear approximations of nonlinear functions. In this section, we show that it is possible to obtain a close approximation of the nonlinear ARZ model about suitable operating points which can then be used for state estimation.

3.1. First-order Taylor series approximation

For the nonlinear function $\boldsymbol{f} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \rightarrow \mathbb{R}^{n_x}$, specified in (11), the first-order Taylor series expansion [49] about a point $(\boldsymbol{x}_0, \boldsymbol{u}_0)$ can be written as

$$\begin{aligned} \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}) \; &\approx \boldsymbol{f}(\boldsymbol{x}_0, \boldsymbol{u}_0) + \nabla \boldsymbol{f}_x(\boldsymbol{x}_0, \boldsymbol{u}_0)(\boldsymbol{x} - \boldsymbol{x}_0) \\ &\quad + \nabla \boldsymbol{f}_u(\boldsymbol{x}_0, \boldsymbol{u}_0)(\boldsymbol{u} - \boldsymbol{u}_0), \end{aligned} \tag{13}$$

where

$$\nabla \boldsymbol{f}_x(\boldsymbol{x}_0, \boldsymbol{u}_0) = \left[\frac{\partial \boldsymbol{f}}{\partial x_1}(\boldsymbol{x}_0, \boldsymbol{u}_0) \; \cdots \; \frac{\partial \boldsymbol{f}}{\partial x_n}(\boldsymbol{x}_0, \boldsymbol{u}_0) \right] \in \mathbb{R}^{n_x \times n_x},$$

and

$$\nabla \boldsymbol{f}_u(\boldsymbol{x}_0, \boldsymbol{u}_0) = \left[\frac{\partial \boldsymbol{f}}{\partial u_1}(\boldsymbol{x}_0, \boldsymbol{u}_0) \; \cdots \; \frac{\partial \boldsymbol{f}}{\partial u_m}(\boldsymbol{x}_0, \boldsymbol{u}_0) \right] \in \mathbb{R}^{n_x \times n_u}.$$

Here, the operating states $\boldsymbol{x}_0$ and operating inputs $\boldsymbol{u}_0$ are not fixed for all k, instead they are selected as close to the time step k as permitted by the availability of reliable input data and state estimates. The choice of operating point is discussed in Section 5.3. We add the coefficients of $\boldsymbol{x}$ from this linearization to the $\boldsymbol{A}$ matrix in (11) to get a new coefficient matrix for the approximate model. We obtain the following linear state-space equation

$$\boldsymbol{x}[k+1] \approx \tilde{\boldsymbol{A}}\boldsymbol{x}[k] + \boldsymbol{B}\boldsymbol{u}[k] + \boldsymbol{c}_1,$$

where $\tilde{\boldsymbol{A}} = \boldsymbol{A} + \boldsymbol{G}\nabla \boldsymbol{f}_x(\boldsymbol{x}_0, \boldsymbol{u}_0)$, $\boldsymbol{B} = \boldsymbol{G}\nabla \boldsymbol{f}_u(\boldsymbol{x}_0, \boldsymbol{u}_0)$, and $\boldsymbol{c}_1 = \boldsymbol{G}(\boldsymbol{f}(\boldsymbol{x}_0, \boldsymbol{u}_0) - \nabla \boldsymbol{f}_x(\boldsymbol{x}_0, \boldsymbol{u}_0)\boldsymbol{x}_0 - \nabla \boldsymbol{f}_u(\boldsymbol{x}_0, \boldsymbol{u}_0)\boldsymbol{u}_0)$. Similarly, we can also linearize the measurement model as follows:

$$\boldsymbol{y}[k] \approx \tilde{\boldsymbol{C}}[k]\boldsymbol{x}[k] + \boldsymbol{c}_2[k],$$

where $\tilde{\boldsymbol{C}}[k] = \boldsymbol{C}[k]\nabla \boldsymbol{h}_x(\boldsymbol{x}_0)$ and $\boldsymbol{c}_2[k] = \boldsymbol{C}[k]\boldsymbol{h}(\boldsymbol{x}_0) - \nabla \boldsymbol{h}_x(\boldsymbol{x}_0)\boldsymbol{x}_0$, where $\nabla \boldsymbol{h}_x(\boldsymbol{x}_0)$ is the gradient of the measurement function given in (12) at $\boldsymbol{x}_0$.

Since we know the input at every time step, we can always linearize using the current input value. In that case, we do not need the third term in the linearization equation (13) as it will always be equal to zero.

3.2. Carleman linearization

Another technique for linearization which is considered to give very good approximations for nonlinear functions is called Carleman linearization [36]. In this technique, a nonlinear model is first polynomialised using Taylor series approximation of a certain order (if not already in the form of a polynomial) followed by linearization of the polynomial by writing the unique monomials as new states of the system. For discrete-time systems, the methodology for Carleman linearization is specified in [50]. We discuss it briefly in Appendix C.

As also mentioned in Appendix C, a major drawback of the approach is that it requires truncation of the newly defined state vectors for implementation in a state-space formulation. Besides this, since the first step of this approach involves *polynomializing* the nonlinear model, the accuracy of the approach depends on the accuracy of the Taylor series approximation. A quantitative comparison between the given linearizaiton techniques is presented in Section 5.3.

4. State Estimation Techniques

In this section, we briefly discuss the different techniques implemented in this work for TSE using the ARZ model.

4.1. Moving Horizon Estimation

MHE is an optimization based state estimation technique which uses measurement data in batches from the most recent time horizon along with a process model to determine the states of the system. It involves solving an optimization problem at every time step of the process with the objective of minimizing the deviation of the estimated states from the modeled states as well as from the measurement data. Being an optimization problem, it is possible to include additional constraints in the problem such as bounds on the state variables. Depending upon whether the model is linear or nonlinear, MHE is divided into linear MHE and nonlinear MHE, both of which have been well explored in the literature. While linear MHE only requires solving a linear program or a quadratic program (QP) and is generally fast and easy to solve using available solvers, nonlinear MHE involves solving a nonlinear optimization problem which is both time consuming and difficult. Since TSE for the purpose of control is required to be done in real time, in practice it is not always possible to spend enough time in solving a nonlinear optimization problem. Therefore, in this paper we implement a linear MHE approach on a linearized version of the process model. Throughout this section, N is used to denote the size of the horizon for optimization. For time steps up to N, that is, near the start of the process, the horizon size is kept equal to the number of time steps from the initial time up to that time. The MHE implemented in this work has a similar objective function to [29]. In the following discussion, we first present our implementation of the MHE optimization problem and then discuss the limitation of another MHE implementation from the literature with respect to TSE and why the current implementation works better for TSE.

4.1.1. Decision variables

The primary decision variables for a single run of the MHE optimization problem at time step k are the state vectors from time step $k-N$ to k denoted by $\boldsymbol{x}_k[t]\ \forall\ t \in [k-N, k]$. These should not be confused with $\hat{\boldsymbol{x}}[k-N], \dots, \hat{\boldsymbol{x}}[k]$ which are the final state estimates. Out of the decision variables for the optimization at time step k, we set the value of the vector $\boldsymbol{x}_k[k]$ as the final estimate, that is, $\hat{\boldsymbol{x}}[k] = \boldsymbol{x}_k[k]$.

4.1.2. Objective function

The objective function for MHE at time step $k \in [N+1, \infty]$ is denoted by $J[k]$ and is given as

$$J[k] = \mu||\boldsymbol{x}_k[k-N] - \bar{\boldsymbol{x}}[k-N]||^2 + w_1 \sum_{i=k-N}^{k} ||\boldsymbol{y}[i] - (\tilde{\boldsymbol{C}}_i \boldsymbol{x}_k[i] + \boldsymbol{c}_{2i})||^2$$
$$+ w_2 \sum_{i=k-N}^{k-1} ||\boldsymbol{x}_k[i+1] - (\tilde{\boldsymbol{A}}_i \boldsymbol{x}_k[i] + \boldsymbol{B}_i \boldsymbol{u}[i] + \boldsymbol{c}_{1i})||^2, \tag{14}$$

Here, $\bar{\boldsymbol{x}}[k-N]$ is a prediction of $\boldsymbol{x}[k-N]$ based on a previously obtained state estimate and is expressed as

$$\bar{\boldsymbol{x}}[k-N] = \boldsymbol{A}\hat{\boldsymbol{x}}[k-N-1] + \boldsymbol{G}\boldsymbol{f}(\hat{\boldsymbol{x}}[k-N-1], \boldsymbol{u}[k-N-1]). \tag{15}$$

Some literature such as [51] suggest using another state estimation technique like UKF to obtain the predicted states from the previous estimate to better utilize the available measurement data. In this work, we simply use the process model as shown in (15). The notation $\boldsymbol{y}[i]$ is the measurement vector at time step $i \in [k-N, k]$, $\tilde{\boldsymbol{A}}_i, \boldsymbol{B}_i$ and $\boldsymbol{c}_{1i}$ are parameters of the linearized state-space equation $\forall\, i \in [k-N, k-1]$, and $\tilde{\boldsymbol{C}}_i$ and $\boldsymbol{c}_{2i}$ are parameters of the linearized measurement model $\forall\, i \in [k-N, k]$. Here, $\tilde{\boldsymbol{A}}_k, \boldsymbol{B}_k$ and $\boldsymbol{c}_{1k}$ are computed at $(\boldsymbol{x}_o, \boldsymbol{u}[k])$ where $\boldsymbol{x}_o = \sum_{i=k-1-N}^{k-1} \boldsymbol{x}_{k-1}[i]/(N+1)$, and $\tilde{\boldsymbol{C}}_k$ and $\boldsymbol{c}_{2k}$ are are computed at $\boldsymbol{x}_o$.

The first term in the the objective is known as the *arrival cost* which serves to connect the decision variables of the current optimization problem with the estimates up to the previous time step. This effectively allows us to consider the impact of data prior to the current horizon in the estimation process. The second and third terms in the objective are penalties on the deviation of the estimates from the measurement data and the modeled dynamics respectively. μ, w_1 and w_2 are the weights specifying our relative confidence on the past data and past estimates, the current measurement data, and the process model, and can be set by the modeler accordingly. The overall goal of the problem is to minimize $J[k]$ under the following constraints.

4.1.3. Constraints

The constraints for the MHE optimization problem consist of the lower and upper bounds on the states, that is, if the bound vectors are $\boldsymbol{x}_{\min}$ and $\boldsymbol{x}_{\max} \in \mathbb{R}^{n_x}$ respectively, then the constraints are defined as

$$\boldsymbol{x}_{\min} \leq \boldsymbol{x}_k[i] \leq \boldsymbol{x}_{\max}, \forall\, i \in [k-N, k]. \tag{16}$$

For the problem at hand, we have $\boldsymbol{x}_{\min} = \boldsymbol{0}$, and $\boldsymbol{x}_{\max} = [\rho_m \;\; \rho_m v_f \;\; \rho_m \;\; \rho_m v_f \;\; \cdots \;\; \rho_m \;\; \rho_m v_f]^T$.

4.1.4. Optimization problem

The above objective and constraints are used to write the following optimization problem

$$\begin{aligned} \underset{\boldsymbol{x}_k[k-N],\ldots,\boldsymbol{x}_k[k]}{\text{minimize}} \quad & J[k] \\ \text{subject to} \quad & (16). \end{aligned} \tag{17}$$

The objective function $J[k]$ can also be expressed as a sum of quadratic and linear terms of the state vectors as shown in Appendix B. Defining $\boldsymbol{z}_k$ by concatenating the decision variables from (17) such that $\boldsymbol{z}_k = [\boldsymbol{x}_k[k-N]^T \; \boldsymbol{x}_k[k-N+1]^T \;\; \cdots \;\; \boldsymbol{x}_k[k]]^T$, we can write the optimization problem (17) in the standard form of a QP defined as

$$\begin{aligned} \underset{\boldsymbol{z}_k}{\text{minimize}} \quad & \boldsymbol{z}_k^T \boldsymbol{H} \boldsymbol{z}_k + \boldsymbol{q}^T \boldsymbol{z}_k \\ \text{subject to} \quad & \boldsymbol{z}_{\min} \leq \boldsymbol{z}_k \leq \boldsymbol{z}_{\max}. \end{aligned} \tag{18}$$

where $\boldsymbol{H} \in \mathbb{R}^{(N+1)n_x \times (N+1)n_x}$ and $\boldsymbol{q} \in \mathbb{R}^{(N+1)n_x}$ consist of the coefficients of the quadratic and linear terms in the objective respectively. $\boldsymbol{z}_{\min}$ and $\boldsymbol{z}_{\max} \in \mathbb{R}^{(N+1)n_x}$ are the lower bound and upper bound vectors of $\boldsymbol{z}_k$ obtained by concatenating $\boldsymbol{x}_{\min}$ and $\boldsymbol{x}_{\max}$ respectively. From Appendix B, it can be seen that $\boldsymbol{H}$ is a positive definite matrix. This makes (18) a convex program which can

Algorithm 1: MHE Implementation for TSE

input: total time t_f, horizon length N, weights μ, w_1, and w_2, state-space matrices $\boldsymbol{A}, \boldsymbol{G}$ and function $\boldsymbol{f}$, measurements from sensors $\boldsymbol{y}[i] \,\forall\, i \in [1, t_f]$, inputs $\boldsymbol{u}[i] \,\forall\, i \in [1, t_f]$, assumed initial state $\hat{\boldsymbol{x}}[0]$, and state bounds $\boldsymbol{x}_{\min}$ and $\boldsymbol{x}_{\max}$

while $t \leq t_f$ **do**

- **set:** operating state $\boldsymbol{x}_o = \sum_{i=t-1-N}^{t-1} \boldsymbol{x}_{t-1}[i]/(N+1)$
- **compute:** predicted state $\bar{\boldsymbol{x}}[t-N]$ using (15), linearized state-space equation matrices $\tilde{\boldsymbol{A}}_t, \boldsymbol{B}_t$, and $\boldsymbol{c}_{1t}$ at $(\boldsymbol{x}_o, \boldsymbol{u}[t])$, linearized measurement equation matrices $\tilde{\boldsymbol{C}}_t$ and $\boldsymbol{c}_{2t}$ at $\boldsymbol{x}_o$
- **set:** coefficient matrices $\boldsymbol{H}$ and $\boldsymbol{q}$ using $\bar{\boldsymbol{x}}[t-N], \boldsymbol{y}[i] \,\forall\, i \in [t-N, t], \tilde{\boldsymbol{A}}_i, \boldsymbol{B}_i$, and $\boldsymbol{c}_{1i} \forall i \in [t-N, t-1]$, and $\tilde{\boldsymbol{C}}_i$ and $\boldsymbol{c}_{2i} \,\forall\, i \in [t-N, t]$, and bound vectors $\boldsymbol{z}_{\min}$ and $\boldsymbol{z}_{\max}$ using $\boldsymbol{x}_{\min}$ and $\boldsymbol{x}_{\max}$
- **solve:** optimization problem (18) for $\boldsymbol{z}_t$
- **set:** $\hat{\boldsymbol{x}}[t] = \boldsymbol{x}_t[t]$

output: $\hat{\boldsymbol{x}}[1], \ldots, \hat{\boldsymbol{x}}[t_f]$

be solved efficiently using readily available QP solvers like CPLEX or MATLAB's `quadprog` function. Algorithm 1 presents the steps involved in MHE as implemented in this study.

4.1.5. Limitation of other implementation

The MHE literature presents some other implementations of the optimization problem as well such as the one presented in [52]. The said approach only considers minimization of the arrival cost and the measurement errors but not the modeling errors that is, the third term in the objective function (14) is missing. This results in a problem that is faster to solve. However, since actual traffic states fluctuate more than what is captured by even a second-order traffic model like the ARZ model, there are always some modeling errors which need to be accounted for by considering modeling errors. Additionally, we have some errors due to the linearization of both the process and the measurement models. As a result, not considering modeling error or the third term in (14) results in a relatively bad performance of MHE for TSE.

Next, we present a brief discussion on the usage of KFs for TSE.

4.2. Kalman Filter variants and limitations

KFs are quite popular when it comes to TSE. Since the traffic process models are nonlinear we cannot use the ordinary KF, instead, most works use variants of KF designed for nonlinear systems namely the EKF, UKF, and EnKF. There is ample literature available on the design of these filters and their application in TSE, see [17] for references. A common limitation of the KF variants is that they do not inherently allow bounds on the state estimates. Since traffic states can only take values from a particular range, this makes it difficult to apply the KF variants directly. Instead some modifications are required such as manually restricting the states to within their bounds after the state estimate for any time step is obtained. Another limitation of the KF variants is that they assume all errors to be Gaussian. This assumption is not necessarily true in many cases including the traffic system which can result in potential errors in state estimation. MHE naturally overcomes both these limitations.

In the following section, we discuss the implementation and results obtained by applying the above mentioned estimation techniques with the help of a numerical example.

5. Case Study using VISSIM

In this section, we apply the state estimation techniques discussed above namely EKF, UKF, EnKF, and MHE, on a traffic simulation example generated in VISSIM micro-simulation software to highlight their advantages and limitations with respect to TSE.

All the simulations are carried out using MATLAB R2020a running on a 64-bit Windows 10 with 2.2GHz Intel^{R} Core^{TM} i7-8750H CPU and 16GB of RAM.

We use the `quadprog` function in MATLAB to solve the MHE optimization problem.

5.1. Case study objectives

The main idea of this case study is to test the performance of the state estimation techniques discussed in Section 4. In particular, we are interested in knowing the answers to the following questions:

- *Q1:* How does the number and placement of sensors impact the performance of the various estimation techniques?
- *Q2:* Is the state estimation performance robust to changing initial conditions?
- *Q3:* Is there an algorithm which requires least tuning to its parameters?
- *Q4:* Which estimation technique utilizes best the data from CVs?
- *Q5:* Which estimation technique is the most reliable overall?

Following is a description of the highway structure used for this study.

5.2. Highway setup and VISSIM simulation

In this study, we model the highway stretch as shown in Figure 2 consisting of one on-ramp and two off-ramps. An additional 100 m of highway stretch is modeled in VISSIM preceding the shown stretch. While we only perform state estimation on the latter 900 m and the attached ramps, this additional stretch of highway modeled in VISSIM provides us with the system inputs namely the demand upstream of Segment 1 as well as the upstream density and speed which are used to calculate the upstream driver characteristic. A similar 100 m stretch is modeled upstream of the on-ramp as well and serves the same purpose of providing the exact inputs. We set the following parameters for the Weidemann 99 model in VISSIM: CC0 1.50 m, CC1 0.9 s, CC2 4.00 m, CC3 8.00, CC4 0.50, CC5 0.60, CC6 6.00, CC7 0.25 m/s^2, CC8 1.00 m/s^2, and CC9 1.50 m/s^2. The ARZ model parameters are selected to keep the simulated state

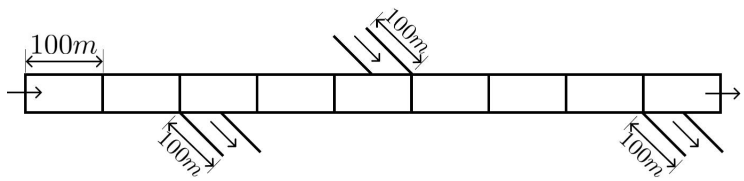

Figure 2: Schematic diagram of the highway considered in this study.

trajectories from the macroscopic model as close to the VISSIM simulation as possible. The selected values are: $v = 102$ km/hr, $\rho_m = 345$ veh/km, $\tau = 20$, and $\gamma = 1.75$. Under the Godunov scheme, the highway and ramps are divided into segments of length 100 m each with a time-step value of 1 s, which satisfies the CFL condition. Thus, there are a total of 24 states in this highway system. In this paper, we consider two types of sensors, first is the fixed sensors which provide density and speed data for every segment they are placed in. The second type of sensor is the CVs which provide the average speed and density data for the segment they are traveling in. In the next section, we look at the results obtained by applying the linearization techniques mentioned in Section 3 on the highway and model specified above. The study that follows will make use of the linearization technique selected based on the ensuing results.

5.3. Comparison of linearization techniques

Here, we present a quantitative comparison of the different linearization techniques discussed in Section 3. Since there is a large difference in scale between the density and relative flow, we use the *Normalized Root Mean Square Error* (NRMSE) for this comparison which is defined in this work, similar to [53] as

$$\text{NRMSE} = \sum_{i=1}^{n_x} \frac{1}{\sigma_i} \sqrt{\frac{1}{k_f} \sum_{k=1}^{k_f} (e_i[k])^2},$$

where σ_i is the standard deviation of the i^{th} state values, $k_f = 200$ sec is the total time of simulation, and $e_i[k]$ is the error for the i^{th} state between the nonlinear and linearized model at time k. Table 2 presents the NRMSE for first, second, and third-order Taylor series approximations for different choices of the gap

Table 2: NRMSE for different approximation techniques for different gap between state update of the operating point for linearization.

	Gap (time steps)			
Approximation Technique	**1**	**2**	**5**	**10**
1st Order Taylor	0.072	0.229	1.014	2.698
2nd Order Taylor	0.054	0.114	0.369	1.759
3rd Order Taylor	0.051	0.104	0.212	0.795
2st Order Carleman	1.379	1.377	1.386	1.670
3rd Order Carleman	0.396	0.431	0.630	2.349

between the states used for linearization. This is done to get a sense of where the first-order expansion stands and whether it is valuable to go for a higher order approximation instead and further linearize it with Carleman linearization. While the approximation is performed at every time step with the available input for that time step, the state in the operating point is only updated once every certain number of steps. This is done to emulate the condition when applying MHE where all the states within a horizon are decision variables and the approximation cannot use the knowledge of the current state at every time step but it can still use the inputs which are assumed to be known. As expected, the NRMSE between the linearized and nonlinear model trajectories increases with the increasing duration of this gap.

Table 2 also presents the NRMSE for the truncated Carleman Linearization for second and third-order polynomials of the ARZ model. Carleman linearization of both orders has a higher NRMSE than even the first-order Taylor approximation for smaller gap between state update of the operating point. This is probably because of the error induced by truncation which consistently deteriorates the quality of the higher-order terms in the state vector (C.2) over several iterations. Although, since our final aim is to perform state estimation, some amount of error in the model is acceptable and can simply be considered a part of the process noise. Based on these results, in what follows, we use the first-order Taylor series approximation for linearization of the ARZ model.

5.4. Observability of the system

To determine the required minimum number and the corresponding placement of sensors, we perform a test of observability for our system using the concept of *Observability Gramian* for discrete time systems [54]. The method is originally meant to determine the observability of linear systems. In this case, we use it to check the observability of the linearized ARZ model. The observability Gramian is defined as

$$\boldsymbol{W}_k = \sum_{m=0}^{\infty} (\tilde{\boldsymbol{A}}_k^T)^m \tilde{\boldsymbol{C}}[k]^T \tilde{\boldsymbol{C}}[k] \tilde{\boldsymbol{A}}_k^m,$$

where $\tilde{\boldsymbol{A}}_k$ is the coefficient matrix of the linearized state-space model and $\tilde{\boldsymbol{C}}[k]$ is the observation matrix of the linearized measurement model at time k around a suitable operating point. The system is considered observable if $\boldsymbol{W}_k$ is positive definite. In this case, since the model parameters change with time due to changing operating points of linearization, the Gramian changes with time as well. This can result in a change in the observability properties. To check if the system is observable for a given sensor placement, we calculate the Gramian for each time step over a duration of the simulation.

From this study, we find that to make the system observable, we need to at least sense the states on the last mainline segment and on all the off-ramps. Therefore, throughout the case study, we keep fixed sensors on these segments. Any additional sensors are placed after these segments are populated with sensors. This is similar to the observations in [26] with respect to observability of the model used in that paper. It appears to be a common property of traffic models that the states of the output segments of the network (last mainline segment and off-ramps) need to be measured to ensure full-observability of the system. This is not surprising as traffic models share similar state-update equations and therefore have similar structure of the state-space matrices which form the observability matrix. While the concept of observability can also be used to determine the optimal sensor placement for state estimation for any given number of sensors under certain conditions [55], here we only use it to determine a minimum number of sensors and its placement. In the following section, we discuss some nuances of implementing the aforementioned estimation techniques in the current study.

5.5. Implementation of estimation techniques

5.5.1. Parameter Tuning

In the implementation of KFs, we need to set three parameters beforehand namely the estimate error co-variance matrix denoted by $\boldsymbol{P}$, the process noise co-variance matrix denoted by $\boldsymbol{Q}$, and the measurement noise co-variance matrix denoted by $\boldsymbol{R}$. In practical applications, these matrices are not known in advance or are difficult to get. In this paper, for all the KF variants, we use a process noise co-variance matrix of the form $\boldsymbol{Q} = q\boldsymbol{I}_{n_x}$ where $q \in \mathbb{R}_+$ and $\boldsymbol{I}_{n_x}$ is a identity matrix of dimension n_x. Similarly, the measurement noise co-variance matrix is set as $\boldsymbol{R} = r\boldsymbol{I}_{n_p[k]}$ with $r \in \mathbb{R}_+$ and $n_p[k]$ is the number of measured states at time k. The initial guess for the estimate noise co-variance matrix is taken as $\boldsymbol{P} = 10^{-3}\boldsymbol{I}_{n_x}$. We manually tune q and r for each case whenever there is a different number or placement of sensors, in an attempt to get the best performance out of the Filters for a comparison between the techniques at their best. Similarly, we also tune the weights in the MHE objective function. In the case study, we note how robust these parameters are in terms of how often they need to be changed for each technique to achieve a good performance.

Besides these values, there are also some technique specific parameters such as in UKF, EnKF and MHE. We find that the fine tuning the values of these parameters does not influence the performance of the techniques considerably. For UKF, we set the following values: $\alpha = 0.1, \kappa = -4$ and $\beta = 2$. For EnKF, we set the number of ensemble points to 100, and for MHE, we set the value of N to 24, which is the number of states in the system. These values are found to be sufficient for the respective techniques. Interested readers can refer to [56] and [57] for interpretation of parameters and more detail on implementation of UKF and EnKF respectively.

5.5.2. Re-scaling to avoid numerical issues

The large difference in the order of magnitude of the two states, density and relative flow, results in numerical issues in both the KFs as well as in MHE. This is handled by re-scaling the objective and constraints of the optimization problem in case of MHE and by re-scaling the state vector in case of KFs.

5.5.3. Applying external bounds on states

The KFs sometimes run into the problem of producing non-physical states such as negative or extremely large densities and relative flows. This is an issue for the process model which includes terms like density raised to fractional power as in (2), which results in numerical issues and forces the estimation to stop. Therefore, it is important to bound the estimates from the KFs to only physical values of the states. In that, we project the obtained estimates in case of EKF to a range with lower bound of zero on all the states, and an upper bound of ρ_m on the traffic densities and $\rho_m v_f$ on the relative flows. In case of UKF, the sigma points are projected first followed by the obtained estimate. In case of EnKF, the ensemble points are projected to within specified bounds. This method of projecting vectors for EKF and UKF has been shown to fit in the KF theory mathematically and is among the popular methods mentioned in [58].

We present the results of the study in the following section.

5.6. Results and discussion

5.6.1. Impact of number of fixed sensors

As sensors are indeed costly, it is thus imperative to determine which state estimation techniques perform better with less number of sensors, and how the performance varies with a changing number of sensors. Herein, we test the effect of increasing the number of fixed sensors on the performance of the four estimation techniques. We do not consider any CVs in this case. As discussed in Section 5.4, we have a minimum of three sensors, one on the last mainline segment and one each on the off-ramps. The first additional sensor is placed

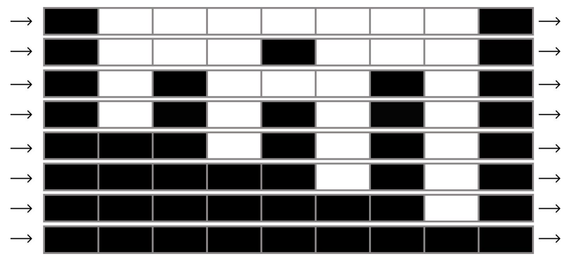

Figure 3: 8 sensor placement configurations for fixed sensors starting with 5 sensors (top) to 12 sensors (bottom). 3 sensors are placed on the 3 ramps and are not shown here. Black boxes depict segments with sensors and white boxes depict otherwise. Arrows indicate direction of traffic.

on the on-ramp. As we add more sensors we try to keep them well distributed across the highway. We have the following placement configurations up to eight sensors: $\{1, 9\}, \{1, 5, 9\}, \{1, 3, 7, 9\}$, and $\{1, 3, 5, 7, 9\}$, after which we add the sensors to the segments without sensors, starting from the second segment. Fig. 3 shows the configuration of sensors starting from 5 sensors up to 12 sensors. We do not add any additional process noise to the state values generated from VISSIM, but we do add a uniform random noise of the order 10^{-3} to the sensor measurements. Figure 4 presents the plots of NRMSE for each of the techniques with changing number of sensors. Figure 4 shows that all the techniques provide better estimation performance with more number of sensors. EKF and MHE perform better than UKF and EnKF in all the cases. Between EKF and MHE, EKF has a slight edge when there are less number of sensors while MHE performs better with more sensors. The difference is quite small though and can probably be attributed to tuning. Among all the techniques, UKF requires slight tweak in the values of the noise co-variance matrices with changing number of sensors to achieve its best performance while the other techniques work well with a single set of values for all the cases. UKF sometimes also runs into numerical issues if the best performance tuning of one case is used in another case. This is a major drawback of UKF making the technique unreliable. EnKF can also sometimes give much worse estimates than presented in Figure 4 as it depends on Monte Carlo sampling which is not always reliable. A noticeable advantage of MHE

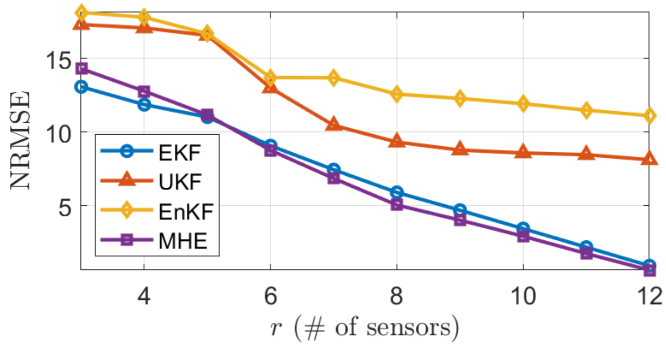

Figure 4: NRMSE values for the four state estimation techniques with different number of sensors with external bounds on the state estimates from KF.

Table 3: Computational time for state estimation per time step (1 sec) of simulation.

Technique	**Computation Time (sec)**
EKF	0.002
UKF	0.006
EnKF	0.016
MHE	0.075

is that it performs well with quite general values of weights for all settings like $\mu = 10^2, w_1 = 10^2, w_2 = 1$ without requiring any fine tuning whereas other techniques require fine tuning to achieve the same level of performance such as $q = 10^{-4}$ and $r = 4 \cdot 10^{-6}$ for EKF, $q = 10^{-4}$ and $r = 9 \cdot 10^{-4}$ for UKF, and $q = r = 10^{-8}$ for EnKF. The average run times per time step of simulation for the techniques are given in Table 3.

For all the KFs it includes the time spent on the prediction and update steps. Figure 6 presents a comparison of the real states and estimated states from MHE for a case with seven sensors. Figure 5 presents the actual and estimated trajectory of the density state from Segment 2 of the highway produced using MHE and EKF.

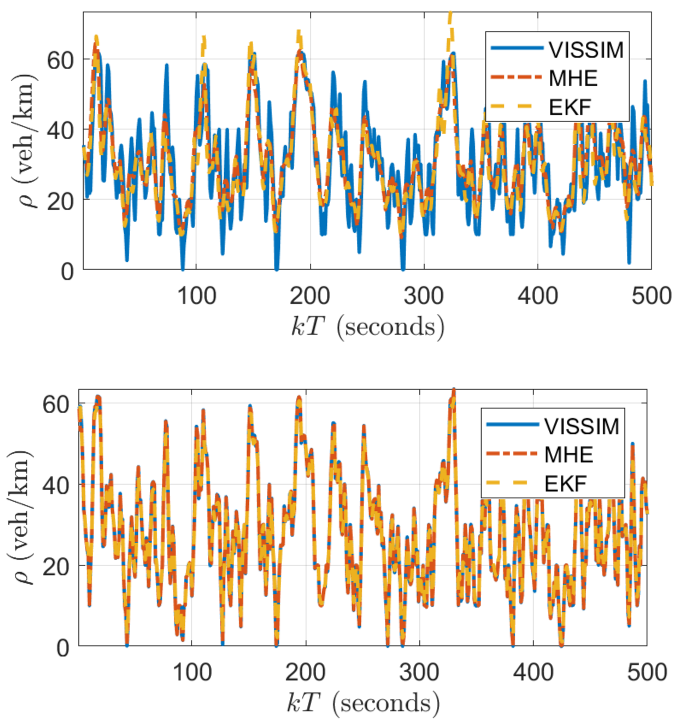

Figure 5: Plots of estimated trajectory for (a) unmeasured density state on Segment 2 [top] and (b) measured density state on Segment 3 [bottom] in the presence of 7 fixed sensors.

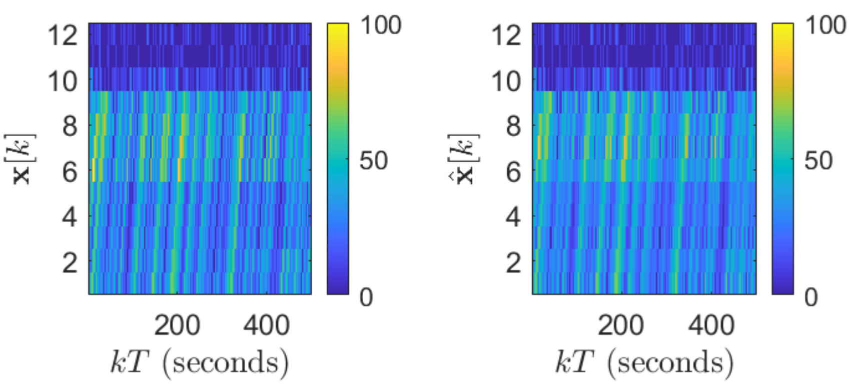

Figure 6: Comparison of real $\boldsymbol{x}[k]$ and estimated $\hat{\boldsymbol{x}}[k]$ traffic densities obtained from MHE for the mainline segments (1 to 9), on-ramp segment (10), and off-ramps segments (11 & 12).

Table 4: NRMSE values for the four state estimation techniques under different sensor placements. Here, in each case, we have sensors on all the ramps and the last mainline segment.

# of sensors	**5**			**7**		
Placement	**{1}**	**{5}**	**{8}**	**{1,2,3}**	**{4,5,6}**	**{6,7,8}**
EKF	11.02	10.01	10.21	7.89	7.24	7.39
UKF	16.58	15.37	15.81	13.71	11.84	12.57
EnKF	16.69	15.19	16.41	14.70	13.80	15.13
MHE	11.16	10.34	10.81	7.15	7.15	8.12

5.6.2. Impact of fixed sensor placement

Under a limited budget resulting in a limited number of sensors, it is important to determine where to place the available sensors to obtain the best state estimation performance. In this section, we test the impact of placing the same number of fixed sensors at different locations on the highway on the performance of the state estimation techniques. We again use NRMSE for a quantitative comparison between the four techniques. First, we test with a total of five sensors where three are needed to maintain observability and are placed on the last mainline segment and the two off-ramps. Here, we fix the fourth sensor on the on-ramp while the fifth sensor is placed at three different locations, once on the first mainline segment, once on the fifth mainline segment, and once on the eighth mainline segment. The idea is to see whether the performance varies if the sensor is placed towards the start, middle or end of the highway. We also test another case with seven sensors. Again, four sensors are placed on the last mainline segment and the three ramps. The remaining three sensors are placed in three placement configurations namely on mainline segments 1, 2 and 3, 4, 5 and 6, and 6, 7 and 8. Table 4 presents the NRMSE values for the different cases mentioned above.

While the error in all the cases with the same number of sensors is not very different for each technique, there appears to be a trend that is followed by all the techniques, that is, sensors at the middle produce the least error followed by sensors towards the end and the worst performance is with sensors at the start of the highway. A possible reason for this could be that since we already have a sensor at the end of the highway providing us data, and we have the inputs from the start of the highway, then the most value of additional sensors is derived by placing them in the middle of the highway.

5.6.3. Robustness to initial conditions

Since the initial value of the states is not known in advance, it is important to assess the robustness of the state estimation techniques to different assumed initial conditions. Here, we increase the gap between the actual initial condition and the assumed initial condition and compare the change in the NRMSE values for each technique. The gap here is measured in terms of the norm between the standardized initial state vectors. Fixed sensors are placed on all the ramps and on mainline segments 1, 3, 7 and 9. Figure 7 shows the trajectories of the NRMSE values for the four techniques with varying initial conditions. As expected, the error increases as we go further away from the actual initial conditions. A majority of this error is from the initial time-steps of simulation, that is the time taken for the estimate to converge to the actual state. The difference in error increment between different techniques therefore shows how quickly a method converges to the actual state. In case of the KF variants, the additional errors due to bad initial conditions disappear after convergence and if the NRMSE is calculated excluding the initial time-steps for instance the first minute, the results are comparable for all the tested initial conditions. In case of MHE however, even though the estimate converges to the actual state quite quickly, the estimates are deteriorated for future time steps as well. This somewhat explains why MHE performance worsens quickly compared to EKF despite being better than EKF when the errors in initial state are small.

5.6.4. Estimation with CVs

As CVs provide a cost effective alternative to the traditional fixed sensors, it is important to study the performance of the state estimation techniques using CVs as opposed to fixed sensors. The present work assumes that there are

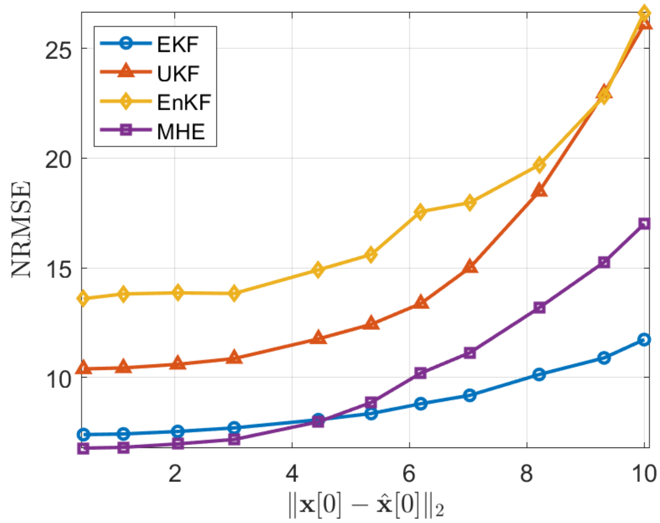

Figure 7: NRMSE values for the four state estimation techniques with changing L2-norm between actual and assumed initial states. Fixed sensors placed on all ramps and mainline segments 1, 3, 7 and 9.

sufficient number of CVs on the highway to obtain the density and speed values of decent quality from the segments of choice. Due to bandwidth constraints we can only query a limited number of CVs at a time, therefore we chose certain segments at the start of the process and follow the CVs on those segments for data. We assume that these connected vehicles are moving at a certain speed, here 0.25 segments/second, so the segments from which we are receiving the information keep changing with time. If these CVs slow down or change course, we stop querying them and select new CVs in the original segment progression. Note that a better method to select CVs for querying within the given bandwidth may be available but is not explored in this paper which only focuses on comparative performance of estimation techniques using the given sources of data. For this study, we fix the location of four fixed sensors, three on the ramps and one on the last mainline segment and further add CVs to the network. At first, the CVs are queried from the first mainline segment, second, from the first and second mainline segment and so on up to seven segments in total. We compare the performance of the estimation techniques using CV

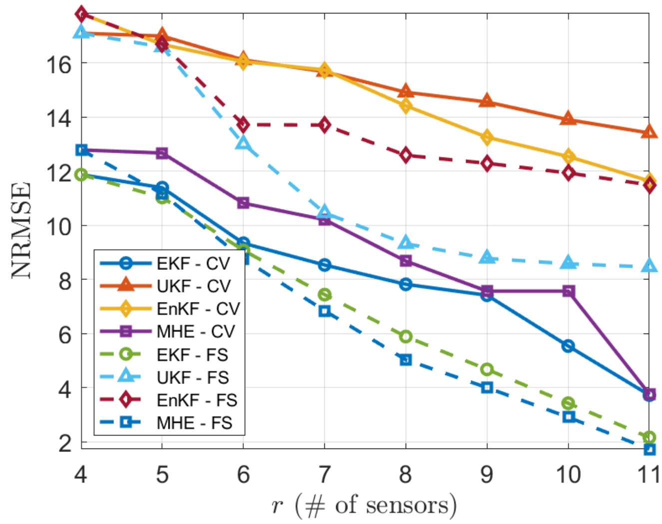

Figure 8: NRMSE values for the four state estimation techniques when using connected vehicles (CV) versus when using fixed sensors (FS).

data against their performance with an equal number of segments installed with fixed sensors. The results are presented in Figure 8. It is seen that among the four techniques, the difference between the performance with CVs and fixed sensors is highest for UKF. This is because, UKF heavily depends on the slight tweaks in the values of $\boldsymbol{Q}$ and $\boldsymbol{R}$ for achieving its best performance with fixed sensors. In this case, since we cannot tune UKF precisely, as the placement is changing with every few time steps, UKF does not achieve its best performance. While the performance of all the techniques is better with fixed sensors, it is promising to see that the techniques perform quite comparably even with just CV data which can be obtained for much less cost than installing fixed sensors. In general, more investigation related to selection of CVs at any time is needed to exploit the full potential of CV data availability for state estimation.

5.6.5. Impact of sensor noise

As sensors are prone to faults, it is possible that the sensor noise changes from its specified value for the sensor from time to time. In such scenarios, a technique more robust to sensor noise would be considered more reliable. In this section, we check the impact of changing the sensor noise on the estimation error of the four techniques. We assume a Gaussian error in this case with distribution $N(0, \sigma^2)$ with a changing standard deviation σ. Here, for the KFs, we set the $\boldsymbol{Q}$ same as the optimal tuning for the given sensor placement as was obtained for a uniform sensor error. The same is found to give pretty good estimates except in case of UKF which runs into instability for certain values of σ. The tuning of $\boldsymbol{R}$ is done in two ways, one is the standard method of tuning where $\boldsymbol{R} = \sigma^2 I_{n_p[k]}$. The other way is to fine tune $\boldsymbol{R}$ so as to reduce the estimation error and keep the same $\boldsymbol{R}$ for all values of σ. Note that this is not always possible in practice as we do not know the actual states. Here we do it to compare all the estimation techniques at a somewhat best performance for each. For MHE, the weights are kept the same as before. Figure 9 and Figure 10 present the variation of the NRMSE values with increasing σ. We subtract the error at $\sigma = 0.1$ from all the values for each estimation technique to only focus on the variation of the error. The error value at $\sigma = 0.1$ is instead reported in the legend next to each technique. It is observed that when the $\boldsymbol{R}$ is not tuned precisely, the errors from the KFs are much larger than MHE although the increase in the errors with σ is more for MHE. The poor performance of the KFs can be explained by the fact that tuning is an important component of the KF performance and without it the NRMSE is high. Why the NRMSE does not increase too much at larger values of measurement error can be explained by the fact that at larger corresponding $\boldsymbol{R}$ matrices, the impact of data on the estimation is already quite small so worsening the data does not influence the NRMSE further. On the other hand, when using a tuned $\boldsymbol{R}$, the performance of EKF is comparable to MHE, and the variation with σ is also almost the same. While the performance of UKF and EnKF are comparatively worse, the estimate is not affected much by measurement errors. This could indicate a trade-off between accuracy and robustness to noise. The sudden spikes in the plot of EnKF could be due to the associated randomness in its formulation which sometimes cause EnKF to produce larger NRMSE values for certain combinations of tuning and measurement errors.

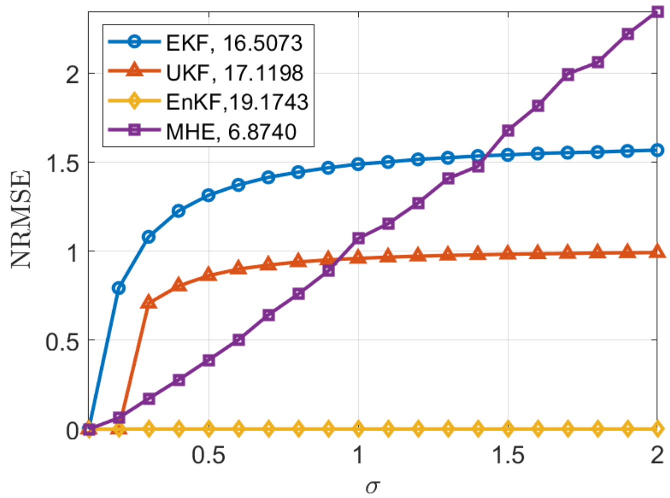

Figure 9: NRMSE values for the four state estimation techniques with changing σ with $\boldsymbol{R} = \sigma^2 I_{n_p[k]}$ for KFs. The NRMSE value at $\sigma = 0.1$ is subtracted from each trajectory and reported in the legend.

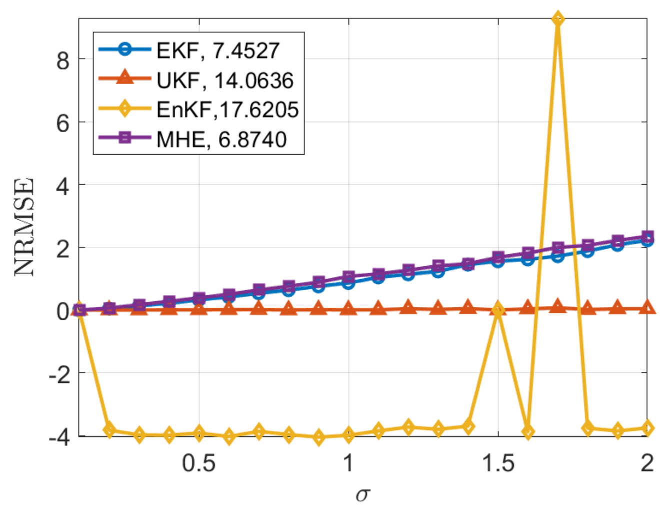

Figure 10: NRMSE values for the four state estimation techniques with changing σ with better tuned R for KFs. The NRMSE value at $\sigma = 0.1$ is subtracted from each trajectory and reported in the legend.

5.6.6. Estimation under congested scenario

All of the above results are obtained using simulations with free-flowing traffic conditions. In this section, we test whether the obtained results on TSE performance under free-flow conditions hold true when there is congestion somewhere on the highway stretch. We simulate a case with congestion forming internally on the highway. In VISSIM, this is achieved by adding reduced speed decisions on Segment 8 of the highway between 100 to 200 seconds from the start. First we test the impact of changing the number of fixed sensors and CVs on the state estimation performance of the different techniques. It is observed that while NRMSE based on standard deviation was a sufficient measure to compare TSE performance under free-flow conditions, it produces biased results when there is congestion. Since some segments of the highway undergo congestion for some part of the simulation and are at free-flow for the remainder, the state values for those segments vary significantly over the duration of the simulation leading to large values of standard deviation of states as compared to segments which are only under free-flow conditions or only under congested conditions during the entire simulation. Therefore, the total NRMSE values will be lower when the estimation error is lower for states with smaller standard deviation.. Therefore in this case we use a scaled version of the RMSE where a value of 0.01 is multiplied with all the relative flow states. The scaled-RMSE or SRMSE is reported in Figure 11.

Comparing with Fig. 8, the performance of the different techniques is more comparable now. MHE performs the best with both fixed sensors and CVs followed by EKF. The performance with fixed sensors and CVs in more comparable now especially for UKF which earlier showed much worse performance with CVs as the source of data. An interesting observation with respect to tuning is that while MHE and EnKF work well with the same tuning as the free-flow case, UKF and EKF do require the co-variance matrices to be tuned again for best results. While EKF works fine with the same new tuning applied to all cases of sensors, UKF still requires the tuning to be changed with different number

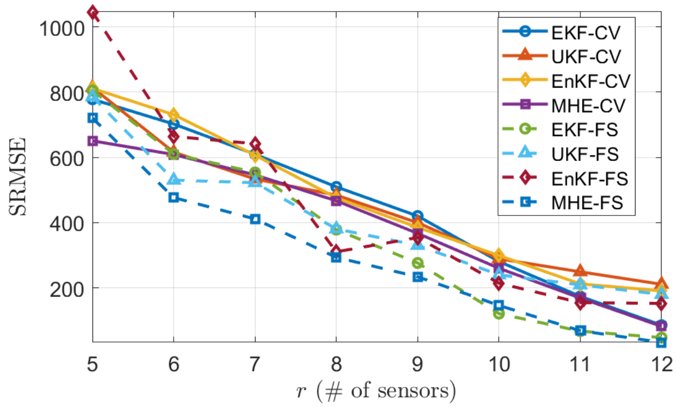

Figure 11: SRMSE values for the four state estimation techniques with changing number of fixed sensors (FS) and connected vehicles (CVs) with internal congestion on the highway.

of sensors or else risks running into numerical issues which increase the error considerably.

Next we check if the impact of sensor placement on estimation is increased when the highway is congested. Since an internal congestion on the highway is not directly captured by the state-space model or the inputs, performance of estimation in this case should depend more on the available data and therefore on the sensor placement. Table 5 presents the SRMSE values for the four techniques under three configurations of fixed sensor placement. The table shows that sensor placement does impact state estimation significantly when there is congestion occurring in some parts of the highway. Specifically, the middle of the highway where most of the congestion occurs is the best place to put the sensors, followed by the start of the highway and the end of the highway. This is similar to the results we saw before with the free-flow condition however the difference was very small in that case. It appears that middle of the highway is indeed better for sensor placement especially when there is congestion. However, a theoretical study may be needed to say anything for sure. Figure 12 presents the actual and estimated density on the highway stretch with congestion with 8 fixed sensors.

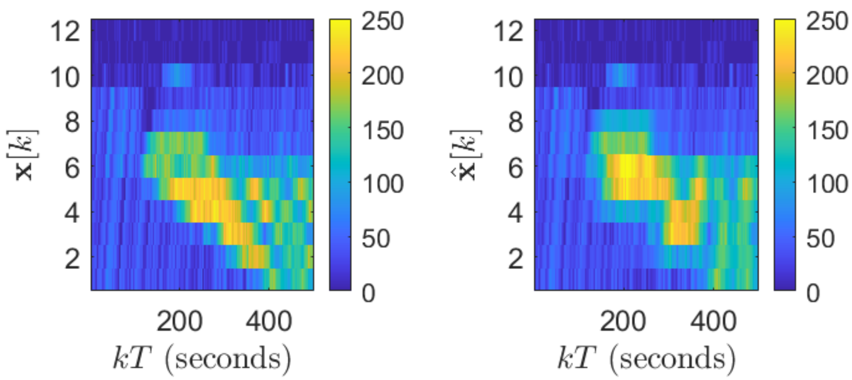

Figure 12: Comparison of real $\boldsymbol{x}[k]$ and estimated $\hat{\boldsymbol{x}}[k]$ traffic densities obtained from MHE for the mainline segments (1 to 9), on-ramp segment (10), and off-ramps segments (11 & 12) for congested scenario.

Table 5: SRMSE values for the four state estimation techniques under different sensor placements with internal congestion on the highway. Here, in each case, we have sensors on all the ramps and the last mainline segment.

# of sensors	**7**		
Placement	**{1,2,3}**	**{4,5,6}**	**{6,7,8}**
EKF	583.0	436.0	636.4
UKF	594.2	428.9	583.9
EnKF	618.1	549.6	868.6
MHE	477.4	316.3	453.1

6. Conclusions and Future Work

From the previous analysis, we have some preliminary suggestions regarding the questions posed in Section 5.1 which are as follows:

- *A1:* As expected, the performance of the state estimation techniques is improved upon increasing the number of sensors in the system. The placement of sensors, on the other hand, does not considerably affect the performance under free-flow conditions but plays a significant role in congested conditions. There is incremental benefit to putting any additional sensors in the middle of the highway as opposed to the start or the end.

- *A2:* The error in state estimation increases as the assumed initial state is taken further away from the actual state. Out of the considered techniques, EKF is most robust to the changing initial conditions as its error changes the least as we go away from the actual initial condition.

- *A3:* UKF requires tweaks in the values of the error co-variance matrices to achieve its best performance and sometimes also runs into numerical issues if the best tuning from one case is used in another case. Other techniques work consistently with a single set of tuning parameters and do not run into numerical issues. MHE performs well with a quite general setting for the weights as opposed to other techniques where the values need to be selected carefully, something not always possible in practice.

- *A4:* EKF, MHE and EnKF work well with CV data under free-flow conditions while all the methods perform closely under congested condition. EKF and MHE perform better overall. Under the free-flow scenario, UKF does not utilize the data well because of its limitation with tuning and numerical issues.

- *A5:* Out of the four techniques studied in this paper, EKF and MHE perform similarly in different scenarios with the exception that MHE is more computationally intensive than EKF while MHE can handle arbitrary constraints and is easier to tune. UKF and EnKF are less reliable than the previous two techniques.

To summarize, we present a state-space formulation for the nonlinear ARZ model while considering junctions in the form of ramp connections. Since the ARZ model is nonlinear, it is not possible to directly apply linear state estimation techniques on it which are considered to be more efficient than nonlinear techniques. We show that it is possible to linearize the nonlinear state-space equation with reasonable accuracy and use the linearized model to implement linear state estimation such as through linear MHE. Besides, the gradient calculated in the process can also be used to implement EKF. We present the formulation for linear MHE which has not previously been used for TSE and show that it is a good choice for TSE. At the end, we compare the linear MHE and EKF with nonlinear state estimation techniques namely UKF and EnKF and show that MHE and EKF perform better than the latter. Future work will consider the optimal placement of sensors considering CVs for TSE. Besides, while the performance of ARZ model against first-order LWR model has been studied in prior research [23] which claims the superiority of the former, some of the newer works [59] have suggested the possibility of the order of the model being less significant for TSE in the presence of sufficient data. Therefore, it will be interesting to carry out a detailed comparative study between the performance of the ARZ model and a first-order model under different scenarios specifically those depicting non-equilibrium conditions under different sensor placements.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

This work is partially supported by the Valero Energy Corporation and National Science Foundation (NSF) under Grants 1636154, 1728629, 2152928, and 1917056, and 2152450, and USDOT CAMMSE.

References

[1] G. Gomes, R. Horowitz, Optimal freeway ramp metering using the asymmetric cell transmission model, Transportation Research Part C: Emerging Technologies 14 (4) (2006) 244–262.

[2] R. C. Carlson, I. Papamichail, M. Papageorgiou, A. Messmer, Optimal motorway traffic flow control involving variable speed limits and ramp metering, Transportation Science 44 (2) (2010) 238–253.

[3] Y. Han, A. Hegyi, Y. Yuan, S. Hoogendoorn, M. Papageorgiou, C. Roncoli, Resolving freeway jam waves by discrete first-order model-based predictive control of variable speed limits, Transportation Research Part C: Emerging Technologies 77 (2017) 405–420.

[4] G. S. van de Weg, A. Hegyi, S. P. Hoogendoorn, B. De Schutter, Efficient freeway MPC by parameterization of ALINEA and a speed-limited area, IEEE Transactions on Intelligent Transportation Systems 20 (1) (2018) 16–29.

[5] N. Lu, N. Cheng, N. Zhang, X. Shen, J. W. Mark, Connected vehicles: Solutions and challenges, IEEE Internet of Things Journal 1 (4) (2014) 289–299.

[6] M. J. Lighthill, G. B. Whitham, On kinematic waves II. A theory of traffic flow on long crowded roads, Proc. R. Soc. Lond. A 229 (1178) (1955) 317–345.

[7] P. I. Richards, Shock waves on the highway, Operations Research 4 (1) (1956) 42–51.

[8] C. F. Daganzo, Requiem for second-order fluid approximations of traffic flow, Transportation Research Part B: Methodological 29 (4) (1995) 277–286.

[9] H. J. Payne, Model of freeway traffic and control, Mathematical Model of Public System (1971) 51–61.

[10] G. B. Whitham, Linear and nonlinear waves, Vol. 42, John Wiley & Sons, 2011.

[11] A. Aw, M. Rascle, Resurrection of "second order" models of traffic flow, SIAM Journal on Applied Mathematics 60 (3) (2000) 916–938.

[12] H. M. Zhang, A non-equilibrium traffic model devoid of gas-like behavior, Transportation Research Part B: Methodological 36 (3) (2002) 275–290.

[13] L. Leclercq, J. A. Laval, E. Chevallier, The lagrangian coordinates and what it means for first order traffic flow models, in: Transportation and Traffic Theory 2007. Papers Selected for Presentation at ISTTT17Engineering and Physical Sciences Research Council (Great Britain) Rees Jeffreys Road FundTransport Research FoundationTMS ConsultancyOve Arup and Partners, Hong KongTransportation Planning (International) PTV AG, 2007.

[14] D. B. Work, O.-P. Tossavainen, S. Blandin, A. M. Bayen, T. Iwuchukwu, K. Tracton, An ensemble Kalman filtering approach to highway traffic estimation using GPS enabled mobile devices, in: 2008 47th IEEE Conference on Decision and Control, IEEE, 2008, pp. 5062–5068.

[15] M. Wright, R. Horowitz, Fusing loop and GPS probe measurements to estimate freeway density, IEEE Transactions on Intelligent Transportation Systems 17 (12) (2016) 3577–3590.

[16] A. Nantes, D. Ngoduy, A. Bhaskar, M. Miska, E. Chung, Real-time traffic state estimation in urban corridors from heterogeneous data, Transportation Research Part C: Emerging Technologies 66 (2016) 99–118.

[17] T. Seo, A. M. Bayen, T. Kusakabe, Y. Asakura, Traffic state estimation on highway: A comprehensive survey, Annual Reviews in Control 43 (2017) 128–151.

[18] Y. Wang, M. Papageorgiou, Real-time freeway traffic state estimation based on extended Kalman filter: a general approach, Transportation Research Part B: Methodological 39 (2) (2005) 141–167.

[19] M. Zhao, X. Yu, Y. Hu, J. Cao, S. Hu, L. Zhang, J. Guo, Y. Wang, Real-time freeway traffic state estimation with fixed and mobile sensing data, in: 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC), IEEE, 2020, pp. 1–7.

[20] I. Agalliadis, M. Makridis, A. Kouvelas, Traffic estimation by fusing static and moving observations in highway networks, in: 20th Swiss Transport Research Conference (STRC 2020)(virtual), STRC, 2020.

[21] A. Kotsialos, M. Papageorgiou, C. Diakaki, Y. Pavlis, F. Middelham, Traffic flow modeling of large-scale motorway networks using the macroscopic modeling tool METANET, IEEE Transactions on Intelligent Transportation Systems 3 (4) (2002) 282–292.

[22] A. Messner, M. Papageorgiou, METANET: A macroscopic simulation program for motorway networks, Traffic Engineering & Control 31 (8-9) (1990) 466–470.

[23] T. Seo, A. M. Bayen, Traffic state estimation method with efficient data fusion based on the Aw-Rascle-Zhang model, in: 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), 2017, pp. 1–6. doi:10.1109/ITSC.2017.8317597.

[24] H. Yu, A. M. Bayen, M. Krstic, Boundary observer for congested freeway traffic state estimation via Aw-Rascle-Zhang model, IFAC-PapersOnLine 52 (2) (2019) 183–188.

[25] R. Wang, Y. Li, D. B. Work, Comparing traffic state estimators for mixed human and automated traffic flows, Transportation Research Part C: Emerging Technologies 78 (2017) 95–110.

[26] N. Bekiaris-Liberis, C. Roncoli, M. Papageorgiou, Highway traffic state estimation with mixed connected and conventional vehicles, IEEE Transactions on Intelligent Transportation Systems 17 (12) (2016) 3484–3497.

[27] S. A. Nugroho, A. F. Taha, C. G. Claudel, A control-theoretic approach for scalable and robust traffic density estimation using convex optimization, IEEE Transactions on Intelligent Transportation Systems 22 (1) (2019) 64–78.

[28] S. C. Vishnoi, S. A. Nugroho, A. F. Taha, C. Claudel, T. Banerjee, Asymmetric cell transmission model-based, ramp-connected robust traffic density estimation under bounded disturbances, in: 2020 American Control Conference (ACC), IEEE, 2020, pp. 1197–1202.

[29] C. V. Rao, J. B. Rawlings, J. H. Lee, Constrained linear state estimation—a moving horizon approach, Automatica 37 (10) (2001) 1619–1628.

[30] C. V. Rao, J. B. Rawlings, D. Q. Mayne, Constrained state estimation for nonlinear discrete-time systems: Stability and moving horizon approximations, IEEE Transactions on Automatic Control 48 (2) (2003) 246–258.

[31] Y. Wang, J. Yuan, S. Yu, Y. Hu, H. Chen, Resilient moving horizon estimation for cyber-physical systems under sensor attacks, in: 2017 11th Asian Control Conference (ASCC), IEEE, 2017, pp. 2274–2279.

[32] A. Alessandri, M. Gaggero, Fast moving horizon state estimation for discrete-time systems using single and multi iteration descent methods, IEEE Transactions on Automatic Control 62 (9) (2017) 4499–4511.

[33] I. I. Sirmatel, N. Geroliminis, Nonlinear moving horizon estimation for large-scale urban road networks, IEEE Transactions on Intelligent Transportation Systems 21 (12) (2019) 4983–4994.

[34] I. Sirmatel, N. Geroliminis, Model-based identification, estimation, and control for large-scale urban road networks, in: 2020 European Control Conference (ECC), IEEE, 2020, pp. 408–413.

[35] S. Timotheou, C. G. Panayiotou, M. M. Polycarpou, Moving horizon fault-tolerant traffic state estimation for the cell transmission model, in: 2015 54th IEEE Conference on Decision and Control (CDC), IEEE, 2015, pp. 3451–3456.

[36] A. Rauh, J. Minisini, H. Aschemann, Carleman linearization for control and for state and disturbance estimation of nonlinear dynamical processes, IFAC Proceedings Volumes 42 (13) (2009) 455–460.

[37] E. F. Camacho, C. B. Alba, Model predictive control, Springer Science & Business Media, 2013.

[38] S. C. Vishnoi, C. G. Claudel, Variable speed limit and ramp metering control of highway networks using lax-hopf method: A mixed integer linear programming approach, IEEE Transactions on Intelligent Transportation Systems.

[39] S. K. Godunov, A difference method for numerical calculation of discontinuous solutions of the equations of hydrodynamics, Matematicheskii Sbornik 89 (3) (1959) 271–306.

[40] R. Courant, K. Friedrichs, H. Lewy, On the partial difference equations of mathematical physics, IBM journal of Research and Development 11 (2) (1967) 215–234.

[41] S. Gottlich, M. Herty, S. Moutari, J. Weißen, Second-order traffic flow models on networks, SIAM Journal on Applied Mathematics 81 (1) (2021) 258–281.

[42] O. Kolb, G. Costeseque, P. Goatin, S. Gottlich, Pareto-optimal coupling conditions for the Aw–Rascle–Zhang traffic flow model at junctions, SIAM Journal on Applied Mathematics 78 (4) (2018) 1981–2002.

[43] A. Khelifi, H. Haj-Salem, J.-P. Lebacque, L. Nabli, Generic macroscopic traffic flow models on junctions, in: 2017 International Conference on Control, Automation and Diagnosis (ICCAD), IEEE, 2017, pp. 445–450.

[44] H. Yang, H. Rakha, A novel approach for estimation of dynamic from static origin–destination matrices, Transportation Letters 11 (4) (2019) 219–228.

[45] C. Lee, J.-B. Lee, M. Kim, Density measurement algorithm for freeway segment using two point detectors, Journal of Advanced Transportation 45 (3) (2011) 207–218.

[46] N. Bekiaris-Liberis, C. Roncoli, M. Papageorgiou, Highway traffic state estimation per lane in the presence of connected vehicles, Transportation Research Part B: Methodological 106 (2017) 1–28.

[47] T. Seo, T. Kusakabe, Y. Asakura, Estimation of flow and density using probe vehicles with spacing measurement equipment, Transportation Research Part C: Emerging Technologies 53 (2015) 134–150.

[48] S. Panichpapiboon, W. Pattara-atikom, Evaluation of a neighbor-based vehicle density estimation scheme, in: 2008 8th International Conference on ITS Telecommunications, IEEE, 2008, pp. 294–298.

[49] P. J. Moreno, B. Raj, R. M. Stern, A vector Taylor series approach for environment-independent speech recognition, in: 1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings, Vol. 2, IEEE, 1996, pp. 733–736.

[50] S. Pruekprasert, T. Takisaka, C. Eberhart, A. Cetinkaya, J. Dubut, Moment propagation of discrete-time stochastic polynomial systems using truncated carleman linearization, IFAC-PapersOnLine 53 (2) (2020) 14462–14469.

[51] C. C. Qu, J. Hahn, Computation of arrival cost for moving horizon estimation via unscented kalman filtering, Journal of Process Control 19 (2) (2009) 358–363.

[52] A. Alessandri, M. Baglietto, G. Battistelli, Receding-horizon estimation for discrete-time linear systems, IEEE Transactions on Automatic Control 48 (3) (2003) 473–478.

[53] A. Lapedes, Nonlinear signal processing using neural networks, Technical Report No. LA-UR-87-2662.

[54] C.-T. Chen, Linear system theory and design.

[55] S. A. Nugroho, S. C. Vishnoi, A. F. Taha, C. G. Claudel, T. Banerjee, Where should traffic sensors be placed on highways?, IEEE Transactions on Intelligent Transportation Systems (2021) 1–14doi:10.1109/TITS.2021.3119211.

[56] E. A. Wan, R. Van Der Merwe, The unscented Kalman filter for nonlinear estimation, in: Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No. 00EX373), Ieee, 2000, pp. 153–158.

[57] G. Evensen, The ensemble Kalman filter: Theoretical formulation and practical implementation, Ocean dynamics 53 (4) (2003) 343–367.

[58] D. Simon, Kalman filtering with state constraints: a survey of linear and nonlinear algorithms, IET Control Theory & Applications 4 (8) (2010) 1303–1318.

[59] Y. Wang, M. Zhao, X. Yu, Y. Hu, P. Zheng, W. Hua, L. Zhang, S. Hu, J. Guo, Real-time joint traffic state and model parameter estimation on freeways with fixed sensors and connected vehicles: State-of-the-art overview, methods, and case studies, Transportation Research Part C: Emerging Technologies 134 (2022) 103444.

Appendix A. State-Space Equation Parameters

In this section, we present the parameters of the state-space equation (11). The said parameters are given as follows:

$$\boldsymbol{A} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & \dots \\ \frac{v_f}{\tau} & 1-\frac{1}{\tau} & 0 & 0 & 0 & \dots \\ 0 & 0 & 1 & 0 & 0 & \dots \\ 0 & 0 & \frac{v_f}{\tau} & 1-\frac{1}{\tau} & 0 & \dots \\ 0 & 0 & 0 & 0 & 1 & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \tag{A.1}$$

$$\boldsymbol{G} = \begin{bmatrix} \frac{T}{l} & 0 & 0 & 0 & \dots \\ 0 & \frac{T}{l} & 0 & 0 & \dots \\ 0 & 0 & \frac{T}{l} & 0 & \dots \\ 0 & 0 & 0 & \frac{T}{l} & \dots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \tag{A.2}$$

$$\boldsymbol{f}(\boldsymbol{x}[k], \boldsymbol{u}[k]) = \begin{bmatrix} q_0[k] - q_1[k] \\ \phi_0[k] - \phi_1[k] \\ q_1[k] - q_2[k] \\ \phi_1[k] - \phi_2[k] \\ q_2[k] - q_3[k] \\ \phi_2[k] - \phi_3[k] \\ \vdots \end{bmatrix} \tag{A.3}$$

where $q_i[k]$ and $\phi_i[k]$ are the traffic flux and relative flux terms for traffic leaving some segment i. As an example of the nonlinear difference terms in (A.3), we present the expressions for $q_{i-1}[k] - q_i[k]$ and $\phi_{i-1}[k] - \phi_i[k]$ where segments $i, i-1, i+1 \in \Omega$. There is an On-ramp $j \in \hat{\Omega}$ between Segment $i-1$ and i and no ramp between Segment i and $i+1$. The time parameter is omitted from the notations for all the discrete time variables for compactness of the expressions. Also, the state variables are written in terms of the traffic variables that they represent.

$$
\begin{aligned}
q_{i-1} - q_i &= \min(\beta_{i-1} S_i, D_{i-1}, \frac{\beta_{i-1}}{1-\beta_{i-1}} \hat{D}_j) - \min(D_i, S_{i+1}), \\
\phi_{i-1} - \phi_i &= q_{i-1} \frac{\psi_{i-1}}{\rho_{i-1}} - q_i \frac{\psi_i}{\rho_i},
\end{aligned}
$$

where the expressions for the various terms are given in Section 2.3. Similarly, if there is an Off-ramp $j \in \check{\Omega}$ between Segment i and $i+1$, and no ramp between Segment $i-1$ and i, then the expressions are given as

$$
\begin{aligned}
q_{i-1} - q_i &= \min(D_{i-1}, S_i) - \min(D_i, \frac{\check{S}_j}{\alpha_i}, \frac{S_{i+1}}{(1-\alpha_i)}), \\
\phi_{i-1} - \phi_i &= q_{i-1} \frac{\psi_{i-1}}{\rho_{i-1}} - q_i \frac{\psi_i}{\rho_i}.
\end{aligned}
$$

Other expressions can also be written in the same manner.

Appendix B. QP formulation for MHE

The MHE objective function is given in (14) as

$$
\begin{aligned}
J[k] = \mu ||\boldsymbol{x}_k[k-N] - \bar{\boldsymbol{x}}[k-N]||^2 + w_1 \sum_{i=k-N}^{k} ||\boldsymbol{y}[i] - (\tilde{\boldsymbol{C}}_i \boldsymbol{x}_k[i] + \boldsymbol{c}_{2i})||^2 \\
+ w_2 \sum_{i=k-N}^{k-1} ||\boldsymbol{x}_k[i+1] - (\tilde{\boldsymbol{A}}_i \boldsymbol{x}_k[i] + \boldsymbol{B}_i \boldsymbol{u}[i] + \boldsymbol{c}_{1i})||^2.
\end{aligned}
$$

The square of the Euclidean norm can be expressed as a product of vectors which can be simplified into quadratic and linear terms in the associated decision variables. For instance, the *arrival cost* term can be expanded as

$$
\begin{aligned}
&\mu ||\boldsymbol{x}_k[k-N] - \bar{\boldsymbol{x}}[k-N]||^2 \\
&= (\boldsymbol{x}_k[k-N] - \bar{\boldsymbol{x}}[k-N])^T \mu \boldsymbol{I}_{n_x} (\boldsymbol{x}_k[k-N] - \bar{\boldsymbol{x}}[k-N]) \\
&= \boldsymbol{x}_k[k-N]^T \mu \boldsymbol{I}_{n_x} \boldsymbol{x}_k[k-N] - \bar{\boldsymbol{x}}[k-N]^T 2\mu \boldsymbol{I}_{n_x} \boldsymbol{x}_k[k-N] \\
&+ \bar{\boldsymbol{x}}[k-N]^T \mu \boldsymbol{I}_{n_x} \bar{\boldsymbol{x}}[k-N]
\end{aligned}
\tag{B.1}
$$

Similarly, the second term can be expressed as

$$
w_1 \sum_{i=k-N}^{k} ||\boldsymbol{y}[i] - (\tilde{\boldsymbol{C}}_i \boldsymbol{x}_k[i] + \boldsymbol{c}_{2i})||^2
$$

$$= \sum_{i=k-N}^{k} (\tilde{\boldsymbol{C}}_i \boldsymbol{x}_k[i]^T w_1 \boldsymbol{I}_{n_p[k]} \tilde{\boldsymbol{C}}_i \boldsymbol{x}_k[i] - (\boldsymbol{y}[i] - \boldsymbol{c}_{2i})^T 2 w_1 \boldsymbol{I}_{n_p[k]} \tilde{\boldsymbol{C}}_i \boldsymbol{x}_k[i]$$

$$+ (\boldsymbol{y}[i] - \boldsymbol{c}_{2i})^T w_1 \boldsymbol{I}_{n_p[k]} (\boldsymbol{y}[i] - \boldsymbol{c}_{2i})) \tag{B.2}$$

and the third term as

$$\begin{aligned} & w_2 \sum_{i=k-N}^{k-1} ||\boldsymbol{x}_k[i+1] - (\tilde{\boldsymbol{A}}_i \boldsymbol{x}_k[i] + \boldsymbol{B}_i \boldsymbol{u}[i] + \boldsymbol{c}_{1i})||^2 \\ & = \sum_{i=k-N}^{k-1} ((\boldsymbol{x}_k[i+1] - \tilde{\boldsymbol{A}}_i \boldsymbol{x}_k[i])^T w_2 \boldsymbol{I}_{n_x} (\boldsymbol{x}_k[i+1] - \tilde{\boldsymbol{A}}_i \boldsymbol{x}_k[i]) \\ & - (\boldsymbol{B}_i \boldsymbol{u}[i] + \boldsymbol{c}_{1i})^T 2 w_2 \boldsymbol{I}_{n_x} (\boldsymbol{x}_k[i+1] - \tilde{\boldsymbol{A}}_i \boldsymbol{x}_k[i]) \\ & + (\boldsymbol{B}_i \boldsymbol{u}[i] + \boldsymbol{c}_{1i})^T w_2 \boldsymbol{I}_{n_x} (\boldsymbol{B}_i \boldsymbol{u}[i] + \boldsymbol{c}_{1i})) \end{aligned} \tag{B.3}$$

Here, the last term in each expansion is a constant and can be removed from the objective function. The sum of the remaining terms can be expressed in terms of the vector $\boldsymbol{z}_k = [\boldsymbol{x}_k[k-N]^T \ \boldsymbol{x}_k[k-N+1]^T \ \cdots \ \boldsymbol{x}_k[k]]^T$ as

$$= \boldsymbol{z}_k^T (\boldsymbol{H}_1 + \boldsymbol{H}_2 + \boldsymbol{H}_3) \boldsymbol{z}_k + (\boldsymbol{q}_1 + \boldsymbol{q}_2 + \boldsymbol{q}_3)^T \boldsymbol{z}_k \tag{B.4}$$

where the various matrices and vectors are defined as follows:

$$\boldsymbol{H}_1 = \begin{bmatrix} \mu \boldsymbol{I}_{n_x} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix}, \tag{B.5}$$

$$\boldsymbol{H}_2 = \boldsymbol{H}_C^T w_1 \boldsymbol{I}_{N_p} \boldsymbol{H}_C \tag{B.6}$$

where

$$\boldsymbol{H}_C = \begin{bmatrix} \tilde{\boldsymbol{C}}_{k-N} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \ddots & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \tilde{\boldsymbol{C}}_k \end{bmatrix},$$

and $N_p = \sum_{i=k-N}^{k} n_p[i]$, and

$$\boldsymbol{H}_3 = \boldsymbol{H}_A^T w_2 \boldsymbol{I}_{N n_x} \boldsymbol{H}_A \tag{B.7}$$

where

$$
\boldsymbol{H}_A = \begin{bmatrix} -\tilde{\boldsymbol{A}}_{k-N} & \boldsymbol{I}_{n_x} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & -\tilde{\boldsymbol{A}}_{k-N+1} & \boldsymbol{I}_{n_x} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \ddots & \ddots & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & -\tilde{\boldsymbol{A}}_{k-1} & \boldsymbol{I}_{n_x} \end{bmatrix}.
$$

$$
\boldsymbol{q}_1 = -2 \left(\begin{bmatrix} \bar{\boldsymbol{x}}[k-N] \\ \boldsymbol{0} \end{bmatrix}^T \boldsymbol{H}_1 \right)^T, \tag{B.8}
$$

$$
\boldsymbol{q}_2 = -2 \left(\begin{bmatrix} \boldsymbol{y}[k-N] - \boldsymbol{c}_{2k-N} \\ \vdots \\ \boldsymbol{y}[k] - \boldsymbol{c}_{2k} \end{bmatrix}^T w_1 \boldsymbol{I}_{N_p} \boldsymbol{H}_C \right)^T, \tag{B.9}
$$

and

$$
\boldsymbol{q}_3 = -2 \left(\begin{bmatrix} \boldsymbol{B}_{k-N} \boldsymbol{u}[k-N] + \boldsymbol{c}_{1k-N} \\ \vdots \\ \boldsymbol{B}_{k-1} \boldsymbol{u}[k-1] + \boldsymbol{c}_{1k-1} \end{bmatrix}^T w_2 \boldsymbol{I}_{N n_x} \boldsymbol{H}_A \right)^T. \tag{B.10}
$$

Replacing $\boldsymbol{H}_1 + \boldsymbol{H}_2 + \boldsymbol{H}_3$ with $\boldsymbol{H}$ and $\boldsymbol{q}_1 + \boldsymbol{q}_2 + \boldsymbol{q}_3$ with $\boldsymbol{q}$ we get the objective function in (18).

Appendix C. Carleman Linearization

Let the polynomial expansion for the update of the state vector $\boldsymbol{x}$ be defined as:

$$
\boldsymbol{x}[k+1] = \sum_{i=0}^{d} \boldsymbol{F}_i[k] \boldsymbol{x}^{[i]}[k], k \in \mathbb{N}, \tag{C.1}
$$

where d is the maximum degree of the polynomial and $\boldsymbol{x}^{[i]}[k]$ represents the i^{th} Kronecker product of the state vector at time k. Since we assume to know the input at each time step, we consider the input to be a part of the constant term in (C.1). Let all the coefficients in this expansion be lumped into a single vector $\boldsymbol{F}[k]$ as follows:

$$\boldsymbol{F}[k] \triangleq [\boldsymbol{F}_0[k]\ \boldsymbol{F}_1[k]\ \boldsymbol{F}_2[k]\ \cdots\ \boldsymbol{F}_d[k]] \in \mathbb{R}^{n \times n^{\leq d}}.$$

We define a new state vector $\boldsymbol{\Lambda}(j,k)$ consisting of the unique mononomials in $\boldsymbol{x}[k]$ up to an arbitrary degree $d_\Lambda \in \mathbb{N}$,that is,

$$\boldsymbol{\Lambda}(d_\Lambda, k) \triangleq [1\ \boldsymbol{x}[k]'\ \boldsymbol{x}^{[2]}[k]'\ \cdots\ \boldsymbol{x}^{[d_\Lambda]}[k]']' \in \mathbb{R}^{n^{\leq d_\Lambda}}. \tag{C.2}$$

Then we can write

$$\begin{aligned}
\boldsymbol{x}[k+1] &= \boldsymbol{F}[k]\boldsymbol{\Lambda}(d,k) \\
\boldsymbol{x}^{[j]}[k+1] &= (\boldsymbol{F}[k]\boldsymbol{\Lambda}(d,k))^{[j]} \\
&= \boldsymbol{F}^{[j]}[k]\boldsymbol{\Lambda}^{[j]}(d,k) \\
&= \sum_{h=0}^{jd}\left(\sum_{(i_l)_{l\leq j}\in H_{j,h}} \boldsymbol{F}_{i_1}[k]\otimes\cdots\otimes\boldsymbol{F}_{i_j}[k]\right)\boldsymbol{x}^{[h]}[k],
\end{aligned} \tag{C.3}$$

where

$$H_{j,h} \triangleq \{(i_l)_{l\leq j} | \sum_{l=1}^{j} i_l = k \text{ and } i_l \leq d\}.$$

The above can be reduced to the following evolution equation for the new state vector for any d_Λ:

$$\begin{aligned}
\boldsymbol{\Lambda}(d_\Lambda, k+1) &= [1\ \boldsymbol{x}[k+1]'\ \boldsymbol{x}^{[2]}[k+1]'\ \cdots\ \boldsymbol{x}^{[d_\Lambda]}[k+1']] \\
&= \boldsymbol{A}(k; d_\Lambda, d_\Lambda d)[1\ \boldsymbol{x}[k]'\ \boldsymbol{x}^{[2]}[k]'\ \cdots\ \boldsymbol{x}^{[d_\Lambda d]}[k']] \\
&= \boldsymbol{A}(k; d_\Lambda, d_\Lambda d)\boldsymbol{\Lambda}(d_\Lambda d, k)
\end{aligned} \tag{C.4}$$

where $\boldsymbol{A}(k; d_\Lambda, d_\Lambda d)$ is a matrix of the coefficients of the monomial terms as obtained from (C.3). Interested readers are referred to [50] for the exact structure of $\boldsymbol{A}(k; d_\Lambda, d_\Lambda d)$ matrix.

Note that $\boldsymbol{\Lambda}(d_\Lambda d, k)$ on the right-hand side in (C.4) has a different length from $\boldsymbol{\Lambda}(d_\Lambda, k)$ on the left-hand side due to different number of Kronecker product terms and $\boldsymbol{A}(k; d_\Lambda, d_\Lambda d)$ is also not square, so this equation cannot directly be used as a state-space equation for state estimation. To deal with this, we need to truncate the $\boldsymbol{A}(k; d_\Lambda, d_\Lambda d)$ matrix and $\boldsymbol{\Lambda}(d_\Lambda d, k)$ vector which results in some loss of accuracy especially at small values of d_Λ.