Lagrangian data assimilation in traffic-flow models

Chao Xia* Courtney Cochran[†] Joseph DeGuire[‡] Gaoyang Fan[§] Emma Holmes[¶]
Melissa McGuirl^{||} Patrick Murphy[§] Jenna Palmer^{**} Paul Carter^{††} Laura Slivinski^{‡‡}
Björn Sandstede^{||}

March 4, 2016

Abstract

Data assimilation of traffic flow remains a challenging problem. One difficulty is that data come from different sources ranging from stationary sensor and camera data to GPS and cell phone data from moving cars. Sensors and cameras give information about traffic density, while GPS data provide information about the positions and velocities of individual cars. Previous methods for assimilating Lagrangian data collected from individual cars relied on specific properties of the underlying computational model or its reformulation in Lagrangian coordinates: these approaches make it hard to assimilate both Eulerian density and Lagrangian positional data simultaneously. In this paper, we propose an alternative approach that allows us to assimilate both Eulerian and Lagrangian data. We show that the proposed algorithm is accurate and works well in different traffic scenarios and regardless of whether ensemble Kalman or particle filters are used. We also show that the algorithm is capable of estimating parameters and assimilating real traffic data and data obtained from microscopic models.

1 Introduction

Estimating the traffic state on a network of highways is a challenging problem. With the availability of stationary sensor data, traffic cameras, and GPS cell phone data, more and more data are now available to predict car density, velocity, and travel time in real time. Predicting traffic involves (i) a model that predicts the traffic at later times based on given initial conditions and (ii) techniques to incorporate and assimilate actual traffic-state observations into the initial conditions to improve the model forecast. The general process of data assimilation is illustrated in Figure 1. To formulate a framework for estimating traffic flow, we therefore need a mathematical model of traffic flow, a sense of what types of observations are available, and a scheme for assimilating these observations into the model. We now discuss these three ingredients in turn.

Mathematical models for traffic flow come in many different flavors. Common models range from cellular automata or microscopic car-following models for individual cars to macroscopic discrete or partial-differential equation (PDE) models for car densities, velocities, and possibly other quantities [26, 31, 37]. Since knowledge of the positions and velocities of all cars on a road or road network is required to successfully employ a microscopic model, it is typically not feasible to use such models for traffic flow prediction since most of this information

*Corresponding author: Chao Xia

E-mail address: chao.xia@brown.edu Tel.: +1 401 863 2335 Fax: +1 401 863-1355

Division of Applied Mathematics, Brown University, Providence, RI 02912, USA

[†]Department of Mathematics, Davidson College, Davidson, NC 28035, USA

[‡]Department of Economics and Mathematics, University of Wisconsin, Madison, WI 53706, USA

[§]Department of Mathematics, University of Utah, Salt Lake City, UT 84112, USA

[¶]Department of Mathematics, Colorado College, Colorado Springs, CO 80903, USA

^{||}Division of Applied Mathematics, Brown University, Providence, RI 02912, USA

^{**}Department of Earth, Environmental and Planetary Science, Brown University, Providence, RI 02912, USA

^{††}Department of Mathematics, Brown University, Providence, RI 02912, USA

^{‡‡}Cooperative Institute for Research in Environmental Sciences, University of Colorado Boulder, Boulder, CO 80309, USA

Figure 1: *Illustration of the process of assimilating observations into models.*

is not known. Thus, macroscopic models are used more frequently for traffic estimation. A commonly used minimal macroscopic PDE model is the Lighthill–Whitham equation [31]: denoting by $\rho(x, t)$ the density of cars and by $v(x, t)$ their velocity at time t at position x , the density ρ satisfies the Lighthill–Whitham equation

$$\rho_t + (\rho v)_x = 0, \quad (1)$$

where ρv is the flux, that is, the number of cars per unit time moving past the location x . To close this equation, we need to assume a constitutive law $v = v(\rho)$ that provides a relation between density and velocity, or add equations for the velocity itself. In this paper, we will use variants of (1) and assume that the velocity and density are related linearly via the Greenshields function [23]

$$v(\rho) = v_{\max} \left(1 - \frac{\rho}{\rho_{\max}} \right), \quad (2)$$

which assumes that there is a maximal velocity $v_{\max}$ at zero density, and a maximal density $\rho_{\max}$ at which traffic will stop.

Traffic-flow observations come from a variety of sources. Stationary sensors (such as induction loops or cameras) provide the flux, average velocity, and local density of cars that move past the fixed sensor location as a function of time. GPS data from cell phones or navigation devices provide information about the positions and velocities of individual cars that communicate this information as a function of time. We refer to data from fixed sensors as Eulerian observations and to GPS data of cars that move with the traffic as Lagrangian observations.

There are a number of different data-assimilation techniques available that help incorporate traffic observations at discrete times into an underlying model: most of these are based on a Bayesian statistics analysis that treats the forecast from the model as the prior distribution and then calculates a posterior distribution based on the available observations. Most traffic estimators employ ensemble Kalman filters for this calculation. In this paper, we will use both ensemble Kalman filters and particle filters: these filters will be discussed in more detail in §3 below. We remark that data assimilation has also been used to estimate parameters (such as the constants $v_{\max}$ and $\rho_{\max}$ in (2)) that appear in the underlying mathematical traffic-flow model as these may not be known a priori and can also change over time.

The initial motivation for this paper was the realization that sensor and GPS data require very different assimilation approaches. For instance, if we use the Lighthill–Whitham model (1) for the car density $\rho(x, t)$ as our underlying model, then sensor data produce estimates for the car density at the sensor location which can be compared directly to the prediction made by the model. GPS data, on the other hand, are Lagrangian in nature: they can provide the location and velocity of an individual car but cannot produce an estimate of the car density at that location; in particular, such data cannot be assimilated directly into a density-based model such as (1). Two approaches have been considered to accomplish the assimilation of Lagrangian data: First, the underlying model can be written in terms of Lagrangian coordinates [47, 48], which makes it possible to assimilate Lagrangian observations. Second, one can convert equation (1) for the density to an equation for the velocity by inverting the density-velocity function $v(\rho)$ and then assimilate velocity observations [44, 45].

The purpose of this paper is to show that Lagrangian data assimilation, originally developed in the context of fluid flows [33, 38], provides an efficient alternative approach to the assimilation of GPS data that keeps the ability to simultaneously assimilate stationary sensor data. The idea behind Lagrangian data assimilation is to

append the differential equations for the positions of the, say, n cars for which GPS data are available to the model (1): if $p_j(t)$ with $j = 1, \dots, N_c$ denotes the position of the j th car, then the closed system

$$\rho_t(x, t) + [\rho(x, t)v(\rho(x, t))]_x = 0, \quad \frac{dp_j}{dt} = v(\rho(p_j, t)), \quad j = 1, \dots, N_c \quad (3)$$

describes the density $\rho(x, t)$ and the positions $p_j(t)$ for all (x, t) . We can now assimilate position and velocity data of the n cars as well as sensor data at the N_s fixed locations q_i for $i = 1, \dots, N_s$ through the linear observation operator

$$(\rho, (p_j)_{j=1, \dots, N_c}) \mapsto ((\rho(q_i, t))_{i=1, \dots, N_s}, (p_j(t))_{j=1, \dots, N_c}).$$

In this paper, we will discuss the efficacy and accuracy of Lagrangian data assimilation for sensor and GPS data in traffic flow. We will also test how robust our algorithm is and whether it can be used to estimate parameters. Specifically, we will investigate how sensitive the proposed algorithm is with respect to different traffic scenario and whether it behaves differently when implemented using ensemble Kalman filters or particle filters. We will also use input from microscopic traffic flow models and from real data to test how well our algorithm works when the observations do not come from the underlying model (1); we will also show that the algorithm works well to estimate the typically unknown parameters that appear in the velocity-density function (2).

Our results show that the proposed algorithm works well and is robust with respect to both filter method and traffic flow scenario. It also seems to handle real data and data generated by microscopic models well as long as the traffic flow to be estimated is compatible with the computational model that is used to assimilate the data: if the computational model is not capable of generating the traffic flow one wants to observe, data assimilation will fail.

The remainder of this paper is organized as follows. In §2, we review the microscopic and macroscopic models that form the basis of our investigation. In §3, we briefly review both ensemble Kalman and particle filters and discuss their capability to estimate parameters. In §4, we outline our algorithms for assimilating sensor and GPS data into macroscopic traffic models; we also discuss parameter estimation, localization, inflation and resampling. Our main results are presented in §5, and we conclude with a discussion of our results and open problems in §6.

2 Overview of traffic models

In this section, we describe the microscopic and macroscopic traffic models that will be used in conjunction with data assimilation in the forthcoming numerical analysis.

2.1 Microscopic model: optimal velocity

Microscopic models attempt to model traffic through differential equations governing the evolution of the individual cars' positions and velocities. In general, these models allow for each car to adjust its velocity and acceleration based on the position and/or velocity of its neighbors. One such model is the optimal-velocity (OV) model introduced in [7], in which each driver adjusts their acceleration to bring their velocity to an optimal velocity that depends on the headway, that is the distance from the vehicle in front. We focus on the formulation of this model as in [19] in which the position $p_n(t)$ of the n th vehicle evolves according to

$$\tau \ddot{p}_n + \dot{p}_n = V_g \left[\mathcal{V} \left(\frac{p_{n+1} - p_n - \bar{s}}{\ell_0} \right) \right] + V_0, \quad (4)$$

where the dot symbol denotes d/dt . Here τ is the reaction time, and the right hand side of the equation is the so-called optimal velocity function. The adjustment $\mathcal{V}$ depends on the difference of the headway $p_{n+1} - p_n$ from an optimal headway $\bar{s}$ that encompasses the car length plus a safety distance: a common choice for this function is $\mathcal{V}(u) = \tanh(u)$, which we will use throughout this paper. Since $\mathcal{V}(0) = 0$, the quantity V_0 represents an optimal velocity at which the car drives when the headway is equal to the optimal headway $\bar{s}$. The quantity V_g represents a velocity gain, and by considering an infinite headway, we have $\mathcal{V}(\infty) = 1$, so that the quantity

$V_g + V_0$ gives an effective speed limit for the drivers. The parameter ℓ_0 is a characteristic length scale which describes the pace at which the driver attempts to achieve the optimal velocity. We note that there have been studies of a variety of extensions of (4), such as allowing drivers to react also to the preceding car ([16, 17]), and allowing the optimal headway $\bar{s}$ to vary deterministically or stochastically in time ([18, 19]).

In this work, we focus on an extension introduced in [10] in which the drivers adjust their velocity according to not only the headway but also the relative velocity to the car in front. In this model, to each driver is associated a new dynamic variable $s_n(t)$, the individual target headway, and the optimal-velocity equation (4) is augmented with a second equation that describes the evolution of $s_n(t)$.

$$\begin{aligned}\tau \ddot{p}_n &= V_g \tanh\left(\frac{p_{n+1} - p_n - s_n}{\ell_0}\right) + V_0 - \dot{p}_n \\ \alpha \dot{s}_n &= \frac{V_g}{\ell_0} (\bar{s} - s_n) - \beta (\dot{p}_{n+1} - \dot{p}_n),\end{aligned}\tag{5}$$

Here the term $\bar{s} - s_n$ describes the relaxation of the individual target headway $s_n(t)$ to the optimal headway $\bar{s}$, while the term $\beta(\dot{p}_{n+1} - \dot{p}_n)$ takes into account that drivers may increase speed if the car in front does so. The parameters α and β are dimensionless: α is a measure of the overall adjustment time of the individual headways, while β measures how proactively drivers react to changes of the relative velocity. Note that setting $(\alpha, \beta) = 0$ recovers the standard optimal-velocity model (4).

We consider the situation of N cars driving on a circular road of fixed length L under the generalized OV model described above which is achieved by adding the periodicity conditions

$$p_{n+N} = p_n + L, \quad s_{n+N} = s_n. \tag{6}$$

It was shown in [10] that the model (5) admits both free-flow and traffic jam (travelling wave) solutions. In this paper, we take the following choice of the parameters $V_g, V_0, \ell_0, \bar{s}, \tau$ which were obtained from Japanese highway data ([6, 19]):

$$V_g = 37.57 \text{mi/hr}, \quad V_0 = 34.30 \text{mi/hr}, \quad \ell_0 = 38.15 \text{ft}, \quad \tau = 0.5 \text{s}, \quad \bar{s} = 82.00 \text{ft}, \tag{7}$$

and we consider two different scenarios of highway traffic, to be described below, obtained by different choices of the parameters α, β, N, L and initial conditions.

The first scenario is that of a long circular road consisting of a large number of cars engaged in free flow interacting with a small traffic jam region. We set $(\alpha, \beta) = (1, 4)$ and the length $L = 22$ miles. To initialize the positions of the cars, we place cars equally spaced along the road with a fixed density of 40 cars per mile except for a 2 mile stretch in which the density is increased to 130 cars per mile for a total of $N = 1060$ cars. This gives that the initial headways are $s_0^f = 131.95 \text{ft}$ in the free flow region and $s_0^j = 40.61 \text{ft}$ in the traffic jam region. The initial velocities of the cars are taken to be

$$v_0^f = V_g \tanh\left(\frac{s_0^f - \bar{s}}{\ell_0}\right) + V_0, \quad v_0^j = V_g \tanh\left(\frac{s_0^j - \bar{s}}{\ell_0}\right) + V_0, \tag{8}$$

for the free flow and jam regions, respectively. We initialize the system with the above conditions and compute the local density at each time step using a circular kernel density estimator in Matlab. In terms of density, the initial condition resembles a square wave which evolves into a shock with decreasing amplitude over time (Figure 2). In addition, we plot flux versus density, which gives what is known as the fundamental diagram.

The second scenario describes the propagation of a traffic jam or travelling wave solution. We set $(\alpha, \beta) = (4, 0.01)$ and the length $L = 2.5$ miles. We place $N = 150$ cars offset from an equal spacing of length $(n-1)L/N$ on the road with initial positions and velocities

$$p_n(0) = \frac{(n-1)L}{N} + 10 \sin\left(\frac{2\pi(n-1)}{N}\right), \quad v_n(0) = V_g \tanh\left(\frac{s_0 - \bar{s}}{\ell_0}\right) + V_0 \tag{9}$$

where $s_0 = 87.97\text{ft}$ is the initial average headway. Figure 3 shows the evolution of the local density of this system. After a short time, a traffic jam, or travelling wave, emerges that travels backwards against the flow of traffic.

2.2 Macroscopic model: Lighthill–Whitham

In this section, we introduce the core macroscopic model which will be used in conjunction with data assimilation techniques to predict traffic flow. The model proposed by Lighthill and Whitham [31] is a scalar first-order conservation law based on the assumption that the number of cars is conserved between two points in the absence of sources and sinks. In this model, the traffic is described by the vehicle density $\rho(x, t)$ which is a continuous function of location x and time t . The resulting equation for the vehicle density $\rho(x, t)$ with periodic boundary conditions is then given by:

$$\partial_t \rho(x, t) + \partial_x \phi(\rho(x, t)) = 0, \quad (x, t) \in [0, L] \times [0, T], \quad (10)$$

$$\rho(x, 0) = \rho_0(x), \quad \rho(0, t) = \rho(L, t), \quad (11)$$

where $\phi(\cdot)$ is the flux function defined on the interval $[0, \rho_{\max}]$, and $\rho_{\max}$ is the maximal density. The term $\rho_0(x)$ denotes the initial data. The function $\phi(\rho)$ expresses the vehicle flux in terms of density, i.e. the fundamental diagram. Assuming that the velocity can be modeled as a function $v(\rho)$ of density ρ in $[0, \rho_{\max}]$, then the flux function can be written as $\phi(\rho) = \rho v(\rho)$.

The function $v(\rho)$ is a modelling choice: examples include the Daganzo–Newell velocity function [44] and the Greenshields affine velocity function [23]. In this paper, we use the linear velocity function by Greenshields:

$$v_G(\rho) = v_{\max} \left(1 - \frac{\rho}{\rho_{\max}} \right), \quad (12)$$

where $v_{\max}$ is the maximum (free flow) velocity, and $\rho_{\max}$ is the maximum (jam) density. In order to take the drivers' response time into consideration (i.e. $v = \phi/\rho$ does not always hold), we would introduce a diffusion term ([49]). By substituting the velocity function and adding the diffusion term to the conservation equation (10), we obtain the viscous Lighthill–Whitham model:

$$\rho_t + v_{\max} \left(1 - \frac{2\rho}{\rho_{\max}} \right) \rho_x = \epsilon \rho_{xx}, \quad (13)$$

where ϵ is the diffusion coefficient, which acts as a smoothing factor in the model. The viscous Lighthill–Whitham model can be used to describe normal traffic flow in the absence of ramps, traffic lights, construction zones, etc. In this model, a shock wave is generated gradually with characteristic speed c given by the Rankine–Hugoniot condition as:

$$c = \frac{\phi(\rho_+) - \phi(\rho_-)}{\rho_+ - \rho_-} =: \frac{[\phi]}{[\rho]}. \quad (14)$$

Here positive speed means that the shock is moving in the downstream direction. We can use the velocity field in the macroscopic model to simulate the movement of individual cars on the road. Assuming $\{p_i(t)\}_{i=1,2,\dots,N_c}$ are the position functions of individual cars, their positions will evolve according to the ordinary differential equation

$$\frac{dp_i}{dt} = v(\rho(p_i, t)), \quad i = 1, 2, \dots, N_c.$$

In this sense, the model is able to provide a microscopic description of the traffic dynamics.

In order to visualize the typical dynamics associated with (13), we present a simulation of normal traffic flow in which a shock develops on a ring road of length $L = 50$ miles. We set parameters $\rho_{\max} = 45$ cars/mile, $v_{\max} = 75$ miles/hour and $\epsilon = 0.1$ and the initial density condition $\rho_0(x) = 0.5\rho_{\max} + 0.4\rho_{\max}\text{sech}(x - L/2)$. A traffic-flow system is initialized with the above conditions and updates every minute. It is observed that the initial density function $\rho_0(x)$ evolves into a shock travelling backward with decreasing amplitude over time (see Figure 4).

The macroscopic model can be used to simulate other real-world traffic scenarios by adding additional terms to (13). In addition to normal traffic flow, we also construct another four traffic scenarios for investigation: on-/off-ramps, traffic lights and stationary as well as travelling bottlenecks. The specific constructions and descriptions of these traffic scenarios are given in §5.

Figure 2: *Evolution of a square wave into a shock in the microscopic system.*

Figure 3: *Emergence of a travelling wave in the microscopic system.*

Figure 4: *Evolution into a shock in the macroscopic system.*

3 Data assimilation and filters

The goal of data assimilation is to accurately predict the states and parameters of a numerical model by incorporating available information obtained from observations into the model. In our work, this technique is carried out to estimate the traffic states and uncertain parameters in the traffic flow models described above. A general introduction to data assimilation and filters is provided in this section.

We first review two common sequential data assimilation methods, the particle filter (PF) and the ensemble Kalman filter (EnKF). The goal of each of these methods is to estimate the Bayesian prior and posterior probability distributions using an ensemble of possible states, as they change in time. First, define the state of interest to be $\mathbf{X}$, and the observation $\mathbf{Y}$ to be a noisy function of the state: $\mathbf{Y} = h(\mathbf{X}) + \eta$, where the observation error is normally distributed $\eta \sim \mathcal{N}(0, \mathbf{R})$ with $\mathcal{N}(\cdot, \cdot)$ denoting the Gaussian distribution. Then, we are interested in estimating the posterior probability distribution $p(\mathbf{X}|\mathbf{Y})$ given a prior distribution $p(\mathbf{X})$ based on our current knowledge of the state, and a likelihood distribution $p(\mathbf{Y}|\mathbf{X})$ of the observations $\mathbf{Y}$ given the state $\mathbf{X}$, based on our knowledge of the noise in the observations. Bayes' rule gives the true posterior distribution in this case:

$$p(\mathbf{X}|\mathbf{Y}) = \frac{p(\mathbf{Y}|\mathbf{X})p(\mathbf{X})}{p(\mathbf{Y})}, \quad (15)$$

where $p(\mathbf{Y}|\mathbf{X})$ is Gaussian likelihood with $\mathbf{Y}|\mathbf{X} \sim \mathcal{N}(h(\mathbf{X}), \mathbf{R})$. In both the PF and EnKF methods, these distributions are approximated by a weighted ensemble of the state $\mathbf{X}$ given by $\{\mathbf{X}_i, w_i\}_{i=1}^{N_e}$ which implies the distribution

$$\sum_{i=1}^{N_e} w_i \delta(\mathbf{X} - \mathbf{X}_i) \quad \text{with} \quad \sum_{i=1}^{N_e} w_i = 1, \quad (16)$$

where $\delta(\mathbf{X} - \mathbf{X}_i)$ is the Dirac delta centered at $\mathbf{X}_i$. Generally for the EnKF, $w_i = 1/N_e$. Between the observation times, the weights are kept fixed and the state variables are evolved according to the dynamics of the system. The main difference between the two methods comes at the time when observations are available. This is described in the next two subsections. We will use the notation that $\{\mathbf{X}_i^f, w_i^f\}_{i=1}^{N_e}$ is the ensemble from the prior distribution $p(\mathbf{X})$ whereas $\{\mathbf{X}_i^a, w_i^a\}_{i=1}^{N_e}$ is from the posterior distribution $p(\mathbf{X}|\mathbf{Y})$.

3.1 Particle filter

The posterior is approximated by updating the weights but leaving the particle positions fixed, i.e., $\mathbf{X}_i^a = \mathbf{X}_i^f := \mathbf{X}_i$. The updated weights are obtained by applying Bayes' rule to the weights as follows:

$$w_i^a = \frac{p(\mathbf{Y}|\mathbf{X}_i)w_i^f}{\sum_{j=1}^{N_e} p(\mathbf{Y}|\mathbf{X}_j)w_j^f}, \quad (17)$$

that is, the updated weights are found by multiplying the likelihood of that particle by the previous weight and normalizing to sum to 1. This is the simplest implementation of the PF, also known as sequential importance

sampling ([13, 21]). For a more complete derivation of the particle filter and the proposal distributions, see [12, 40, 42].

Due to the finite nature of the approximation and the recursive updating of the weights, sequentially applying this algorithm eventually leads to one particle with very high weight, while the rest of the particles have almost zero weight (so-called “filter divergence” or “weight collapse”). To avoid this, various resampling methods may be used ([43]). The basic idea behind each of these methods is to monitor when a predetermined threshold is hit (for example, when the “effective sample size” becomes small ([30])), and to then resample the particles from the discrete approximation of the posterior distribution and reset all weights to $1/N_e$. We approximate the effective sample size to be

$$N_{\text{eff}} \approx \frac{1}{\sum_{i=1}^{N_e} w_i^2} = \begin{cases} N_e & \text{for } w_j = 1/N_e \\ 1 & \text{for } w_j = \delta_{ij} \end{cases} \quad (18)$$

as in [30]. We will apply resampling when $N_{\text{eff}} < N_{\text{eff}}^{\text{thresh}}$ for a predetermined threshold $N_{\text{eff}}^{\text{thresh}}$ that will typically be a small fraction of the total number of particles N_e .

One major drawback of the PF is that it has been shown to fail in high dimensions ([41]). Thus, in the case where the state of interest is a spatially-discretized function over some domain, the PF is intractable. However, when the state dimension is small enough and the number of particles is large enough, the PF can provide an accurate approximation to the exact Bayesian posterior distribution. This is especially useful if this distribution is skewed or multimodal, which is often the case when the dynamical system governing the state is nonlinear.

3.2 Ensemble Kalman filter

Like the PF, the EnKF [14, 15] employs an ensemble of state vectors $\{\mathbf{X}_i\}_{i=1\dots N_e}$ to represent the posterior distribution; however, unlike the PF, the ensemble members are equally weighted for the entire assimilation window. Instead of updating the weights at analysis times, the members themselves are updated according to an ensemble approximation of the traditional Kalman filter update step, given here by the so-called perturbed observation EnKF [9, 15, 28]

$$\mathbf{X}_i^a = \mathbf{X}_i^f + \mathbf{K}(\mathbf{Y} - \mathbf{H}\mathbf{X}_i^f + \eta_i), \quad (19)$$

$$\mathbf{K} = \mathbf{P}^f \mathbf{H}^T (\mathbf{H} \mathbf{P}^f \mathbf{H}^T + \mathbf{R})^{-1}, \quad (20)$$

where $\mathbf{X}_i^f$ is the forecast of the i^{th} ensemble member, $\mathbf{X}_i^a$ is the i^{th} updated (analysis) ensemble member, $\mathbf{H}$ is the observation operator, $\mathbf{K}$ is the Kalman gain matrix, $\mathbf{R}$ is the covariance of the observation error, and $\eta_i \sim \mathcal{N}(0, \mathbf{R})$ are the observation perturbations. $\mathbf{P}^f$ is the forecast ensemble covariance given by

$$\mathbf{P}^f = \frac{1}{N_e - 1} \sum_{i=1}^{N_e} (\mathbf{X}_i^f - \bar{\mathbf{X}}^f)(\mathbf{X}_i^f - \bar{\mathbf{X}}^f)^T, \quad \bar{\mathbf{X}}^f = \frac{1}{N_e} \sum_{i=1}^{N_e} \mathbf{X}_i^f. \quad (21)$$

When the observation operator $\mathbf{H}$ is nonlinear, instead of calculating $\mathbf{H}$ and $\mathbf{P}^f$ separately our approach is to treat $\mathbf{P}^f \mathbf{H}^T$ and $\mathbf{H} \mathbf{P}^f \mathbf{H}^T$ as whole terms as follows:

$$\begin{aligned} \mathbf{H} \mathbf{P}^f \mathbf{H}^T &= \frac{1}{N_e - 1} \sum_{i=1}^{N_e} (\mathbf{Z}_i^f - \bar{\mathbf{Z}}^f)(\mathbf{Z}_i^f - \bar{\mathbf{Z}}^f)^T, \quad \bar{\mathbf{Z}}^f = \frac{1}{N_e} \sum_{i=1}^{N_e} \mathbf{Z}_i^f, \\ \mathbf{P}^f \mathbf{H}^T &= \frac{1}{N_e - 1} \sum_{i=1}^{N_e} (\mathbf{X}_i^f - \bar{\mathbf{X}}^f)(\mathbf{Z}_i^f - \bar{\mathbf{Z}}^f)^T, \quad \bar{\mathbf{X}}^f = \frac{1}{N_e} \sum_{i=1}^{N_e} \mathbf{X}_i^f, \end{aligned}$$

where $\bar{\mathbf{Z}}^f = h(\bar{\mathbf{X}}^f)$, which represents the determinant observations without noise η . There has been a lot of work towards improving the EnKF to make it feasible for very high dimensional systems; in particular, covariance inflation and localization have provided significant improvement of the performance of the EnKF (see [4, 24, 28, 29]). Since the ensemble size is often much smaller than the dimension of the state, spurious

correlation will arise in the sample covariance matrices. Localization is a method in which these spatially long-range correlations are diminished: in particular, this is often implemented via a Schur product of the sample covariance matrix and some sort of cutoff matrix; see [20] for a commonly-used localization function, the Gaspari-Cohn correlation function.

Another problem that may arise with the version of the EnKF described here is artificial collapse of the covariance matrix. In other words, the ensemble members may be overly tight around the mean. To address this, the covariance may be “inflated” before the Kalman update step is performed. This inflation could be as simple as multiplying the covariance matrix by a scalar larger than 1, but more sophisticated methods, such as adaptive inflation [2, 3], have also been developed.

3.3 Parameter estimation

Finally, we briefly describe three approaches for how data assimilation may be used to estimate parameters in addition to the state of interest. One approach is the augmented-vector method, in which the parameters to be estimated are appended to the state vector ([5, 32]) and are updated concurrently. Another approach is the “separated” method [8], in which the state is updated first, followed by updating the parameters, using localization on the state and non-global parameters. The third approach applies a two-stage filter [39] that consists of applying the EnKF to the state estimate and the PF to the parameters.

There are then several methods that may be used to model the evolution of the parameters between assimilation steps. The parameters may be kept fixed between time steps (persistence model, see [11]), but this can lead to degeneracy (sample attrition) when using PF methods. In [32], the authors suggest an artificial evolution of parameters between observations, in which the parameter evolution model is given by

$$\theta_{t+1} = \theta_t + \xi_{t+1}, \quad (22)$$

$$\xi_{t+1} \sim \mathcal{N}(0, W_{t+1}), \quad (23)$$

where W_{t+1} is a specified covariance matrix. In [46], multiplicative parameters were estimated by evolving the parameters under a temporally smoothed version of the persistence model. It was shown that this results in smooth temporal variation of the parameters, which in turn prevents blow-up of the model.

In this paper, the augmented-vector approach is used to estimate parameters, and artificial evolution is introduced to time-varying parameters.

4 Data assimilation implementation for mmodelling traffic flow

In this section, a complete system of traffic dynamics consisting of state and observation models will be built in order to apply the data assimilation techniques outlined above to traffic flow estimation. In addition, we construct the discretized version of the system which will be used for numerical calculation, and specify some details regarding the EnKF and PF, such as localization, inflation and resampling.

4.1 Eulerian and Lagrangian observations

There are two main categories of observation models in traffic state estimation: one is Eulerian observation in which the data is collected from stationary sensors on the road such as cameras, induction loops, and radar, and the other is Lagrangian observation, which includes the position and velocity data from vehicles equipped with a GPS device or cell phone. In the following part, we use the viscous Lighthill–Whitham model (13) as our traffic state model.

4.1.1 Continuous observation model

Stationary sensors along the road can be used to count the number of cars passing by and thus provide vehicle flux data as Eulerian observations. The complete traffic and observation system can be expressed as:

$$\begin{aligned}\rho_t + \phi(\rho)_x &= \epsilon \rho_{xx}, \\ h(\rho) &= \phi(\rho(q_i, t)) + \eta_i^\rho(t), \quad i = 1, 2, \dots, N_s,\end{aligned}\tag{24}$$

where $\{q_i\}_{i=1,2,\dots,N_s}$ are the locations of the sensors on the road, $\phi(\rho(q_i, t))$ are the flux data at the location of the i th sensor at time t , and $\eta_i^\rho(t)$ is the time-dependent observation noise for the i th sensor. The fundamental diagram shows a nonlinear relationship between vehicle flux and vehicle density, so the observation operator h is nonlinear in the Eulerian observation model (24). As mentioned previously, the EnKF provides a posterior distribution close to a Gaussian distribution, so the nonlinear operator h may influence the efficacy of EnKF. In addition, the positions of the sensors are fixed and hence only provide observations at the same locations. However, one significant advantage of Eulerian observations is that modern traffic sensors are very sensitive and the probability of missing a car is often less than 0.1% ([1]); therefore the observation noise η_i^ρ is very small.

Aside from sensors, velocity and position data can be obtained directly from probe cars through GPS devices or cell phones which can indirectly reflect the traffic situation. The Lagrangian observation model we propose can now be expressed as:

$$\begin{cases} \rho_t + \phi(\rho)_x = \epsilon \rho_{xx} \\ \dot{p}_j = v(\rho(p_j, t)) \quad j = 1, 2, \dots, N_c, \\ h_1(\rho, p_j) = p_j(t) + \eta_j^p \quad j = 1, 2, \dots, N_c \\ h_2(\rho, p_j) = v(\rho(p_j(t), t)) + \eta_j^v \quad j = 1, 2, \dots, N_c, \end{cases}\tag{25}$$

where $\{p_j(t)\}_{j=1,2,\dots,N_c}$ and $\{v(\rho(p_j(t), t))\}_{j=1,2,\dots,N_c}$ are the positions and velocities of cars on the road at time t . The observation noise terms η_i^p and η_i^v represent the position error and velocity error, respectively, of GPS devices, which depend on the carrier's signal, weather conditions, etc. We take uniform noise $\delta(\eta_i^p) = 5.12\text{m}$ ([22]) and $\delta(\eta_i^v) = 0.0707\text{m/s}$ ([34, 36]) according to the current accuracy of GPS devices. The use of Lagrangian observations has some disadvantages: in general, η_j^p and η_j^v are relatively large when compared to the observation noise of sensors, whose missing probability is only 0.1%. In addition, the overuse of GPS data may give rise to privacy issues. However, one significant advantage of Lagrangian observation is that vehicles are able to provide traffic information at different locations instead of fixed positions when moving on the road.

We emphasize that (24) and (25) can be combined in a straightforward fashion to simultaneously assimilate Eulerian and Lagrangian data by adding the observation function h defined in (24) to the system (25): we will test the resulting algorithm in §5.1.

4.1.2 Discretized observation model

Traffic flow models can be cast in a discrete state-space form, which makes them suitable for recursive data-assimilation techniques. Generally, these models can be formulated as follows:

$$\begin{aligned}\mathbf{X}_t &= f(\mathbf{X}_{t-1}, \theta_{t-1}), \\ \mathbf{Y}_t &= h(\mathbf{X}_{t-1}, \theta_{t-1}) + \boldsymbol{\eta}_t, \quad \boldsymbol{\eta}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_t).\end{aligned}$$

The road is discretized into M equally distributed mesh points $\{x_i\}_{i=1,2,\dots,M}$ for numerical calculation, and $\rho_i(t)$ denotes the density on the i th mesh point located at x_i at time t . When solving the macroscopic traffic model, the spectral method is used to compute the first and second derivatives in space x , and the Runge-Kutta method is applied to approximate the derivative in time t .

In the Eulerian observation model (24), the state of interest is the density value on the mesh points, i.e. $\mathbf{X}_t = [\rho_1(t), \rho_2(t), \dots, \rho_M(t)]$, and the observation is the flux, i.e. $\mathbf{Y}_t = [\phi(q_1, t), \phi(q_2, t), \dots, \phi(q_{N_s}, t)] + \boldsymbol{\eta}_t$, where $\boldsymbol{\eta}_t$ represents the noise of flux observations. In order to simplify the numerical calculations, the sensors are

appropriately set to lie on the mesh points. The observation operator h is obviously nonlinear, which can be handled by the proposed approach in §3.2 when using EnKF.

In the Lagrangian observation model (25), the state of interest consists of the density field and cars' positions, i.e. $\mathbf{X}_t = [\rho_1(t), \rho_2(t), \dots, \rho_M(t), p_1(t), p_2(t), \dots, p_{N_c}(t)]$. The observation can be taken as the cars' positions or velocities or both, i.e. $\mathbf{Y}_t = [p_1, p_2, \dots, p_{N_c}] + \boldsymbol{\eta}^p$, or $\mathbf{Y}_t = [v_1, v_2, \dots, v_{N_c}] + \boldsymbol{\eta}^v$ or $\mathbf{Y}_t = [p_1, p_2, \dots, p_{N_c}, v_1, v_2, \dots, v_{N_c}] + [\boldsymbol{\eta}^p, \boldsymbol{\eta}^v]$. One challenge of assimilating Lagrangian observations is that the velocity field is defined on mesh points and there is no guarantee that the observed cars will fall exactly on these points. To account for this, we use a trigonometric interpolation approach [27] in our numerical algorithm to approximate the velocity of a car at a given position.

4.2 Localization, inflation and resampling

4.2.1 Localization function in EnKF

The traditional localization, such as the Gaspari-Cohn function in [20], works on the forecast covariance matrix $\mathbf{P}^f$. From the formula of EnKF, we see that the matrix $\mathbf{P}^f$ only appears in the Kalman gain matrix $\mathbf{K}$. Thus we propose a more direct localization approach that works on the Kalman gain matrix $\mathbf{K}$ rather than the forecast covariance matrix $\mathbf{P}^f$. According to (20), the matrix $\mathbf{K}$ has $\dim(\mathbf{X})$ rows and $\dim(\mathbf{Y})$ columns, so the Kalman gain matrix can be treated as the correlation relationship between the state $\mathbf{X}$ and the observation $\mathbf{Y}$.

Our approach defines the localization coefficient by columns. The j th column in the matrix $\mathbf{K}$ represents the correlation between states and the j th observation from sensors or cars. We assume that the j th observation will directly or indirectly impact the traffic states physically nearby and have negligible impact on the traffic states far away.

In Eulerian observation system, the mesh points within 0.5 miles of the sensors are assumed to be affected by the observations. Thus we formulate an exponential decay localization function for the j th column in matrix $\mathbf{K}$:

$$\exp\{-d|x_i - (q_j + s)|\} \quad \text{with} \quad |x_i - q_j| < 0.5\text{mile},$$

where x_i is the coordinate of the i th mesh point, q_j is the coordinate of j th sensor, d is a decay coefficient describing the decay speed of localization and s is a shift coefficient centering the localization at $q_j + s$. The cars detected by the sensors will move ahead during time step ΔT , and this forward distance is the shift coefficient s . We set the parameters $d = 0.5$ and $s = 0.35\text{mile}$.

In the Lagrangian observation system, the assumption and formula for the localization function is similar to that of Eulerian observations. Considering the position and velocity data collected from GPS devices are in-time data and the delay is not significant, we set the shift coefficient $s = 0\text{mile}$. Here we choose $d = 1.2$ for the decay coefficient.

4.2.2 Inflation algorithm in EnKF

Two sets of inflation algorithms are used for the EnKF: constant scalar inflation and adaptive inflation. Constant scalar inflation is used for the simulations of multiple traffic scenarios. Since the localization algorithm does not tighten the Kalman gain matrix, we use the identity inflation matrix for the simulated scenarios. However, when assimilating real traffic data from Minnesota Department of Transportation, we use adaptive inflation. A detailed description of this adaptive inflation algorithm is available in [3].

4.2.3 Resample algorithm in PF

The ‘‘weight collapse’’ phenomenon of the PF can be avoided by a resampling algorithm, such as probabilistic resampling, residual resampling, stochastic universal resampling, or Monte Carlo Metropolis Hastings resampling to name a few. In this paper, we apply the Monte Carlo Metropolis Hastings [25] to the PF. The predetermined threshold is set as 5% of the total particle number N_e as in (18), i.e. $N_{\text{eff}}^{\text{thresh}} = 0.05N_e$. We also add jitter noise

to each particle after resampling in order to spread them out, which is similar to the inflation process in the EnKF.

4.3 Parameter estimation for traffic flow model

When parameter estimation is activated, the augmented vector is formed by concatenating the states of interest $\mathbf{X}$ with the uncertain parameters θ , i.e. $[\mathbf{X}; \theta]$, and the observation matrix is of the form $[\mathbf{H}, 0]$, where $\mathbf{H}$ transforms the state $\mathbf{X}$ to the observation $\mathbf{Y}$. The maximal density $\rho_{\max}$ and maximal velocity $v_{\max}$ are the core parameters in the macroscopic traffic model, and other uncertain parameters will be taken into consideration depending on the specific traffic scenarios, which will be discussed with details in the description of relevant simulations. Because the parameters to estimate are of low dimension, localization and inflation are not necessary for parameter estimation when applying the EnKF. The localization and inflation matrices for parameters are set to the identity matrix by default in our data assimilation algorithm.

4.4 Performance criteria

In this paper, we use both absolute and relative root-mean-square-errors (abbreviated as RMSE) as performance indices to assess different scenarios. The absolute performance index at time t_k is defined as:

$$\text{RMSE}_A = \sqrt{\frac{\sum_{i=1}^M (\rho_i(t_k) - \hat{\rho}_i(t_k))^2}{M}} \quad (26)$$

and the corresponding relative performance index is defined as:

$$\text{RMSE}_R = \frac{\text{RMSE}_A}{\rho_{\max}} \quad (27)$$

where $\rho_{\max}$ is the maximal density in the macroscopic model. Here M is the number of mesh points of the considered highway stretch, $\rho_i(t_k)$ denotes the true density at i th mesh point x_i at time t_k , and $\hat{\rho}_i(t_k)$ denotes the corresponding estimation from data assimilation. For those traffic scenarios with unknown maximal density $\rho_{\max}$, the denominator in the relative RMSE is replaced by the average $\sum_{i=1}^M \rho_i(t_k)/M$ of the true density.

5 Main results

The main goal of this work is to apply data assimilation to macroscopic and microscopic traffic flow models to estimate traffic states and parameters using the theory described in the previous sections. Specifically, we will investigate

1. assimilation of Eulerian and Lagrangian data in dynamic traffic flow;
2. efficacy of Lagrangian data assimilation;
3. impact of sensor location on data assimilation;
4. parameter estimation of traffic flow models;
5. data assimilation of microscopic traffic data using macroscopic models;
6. parameter estimation for real traffic data.

Note that the first three topics are investigated under the assumption that the estimator has the exact values of model parameters, and the emphasis is placed on the capability of traffic state estimators. However, in the last three sets of simulations, the parameter estimation is activated with the intent of investigating the capability of parameter estimators.

5.1 Data assimilation of Eulerian and Lagrangian data

First, we demonstrate that the proposed algorithm can be used to assimilate Eulerian, Lagrangian, and combined Eulerian-Lagrangian observations into a macroscopic model to accurately estimate traffic states. To test the

Figure 5: *Deceleration factor* $a(x, t; x_1^\ell)$.

algorithm, we consider a ring road with a traffic light for our simulations and assimilate sensor and GPS data. In order to better evaluate the performance of the algorithm, we will compare the results also to those obtained by simulating the macroscopic model with the same initial data and parameter values but without assimilating observations.

In the macroscopic model, we simulate the traffic flow on a ring road of length $L = 50$ miles, which is equally divided into 256 mesh points for the numerical computation. We use the following parameter values: maximal density $\rho_{\max} = 45$ cars/mile, maximal density $v_{\max} = 75$ miles/hour, and viscous coefficient $\epsilon = 0.1$, and set the initial density condition $\rho_0(x) = 0.5\rho_{\max} + 0.4\rho_{\max}\text{sech}(x - L/2)$. The Eulerian and Lagrangian observations are collected and assimilated every 1 minute, and the entire simulation has 180 total updates. For the Eulerian observation system, we place $N_s = 8$ stationary sensors equidistantly along the ring road. For the Lagrangian observation system, we pick $N_c = 15$ cars with GPS devices that are initially equally spaced along the ring road. This is the basic setup for the macroscopic model that will also be used in §5.2-§5.4.

In order to create a dynamic, time-dependent traffic flow, we place a traffic light on the road to force vehicles to decelerate and accelerate periodically, resulting in periodic oscillations of the vehicle density ρ along the road. The resulting system is given by

$$\rho_t + (\rho v_G(\rho(x, t)) a(x, t; x_1^\ell))_x = \epsilon \rho_{xx},$$

$$a(x, t; x_1^\ell) = \begin{cases} 0.5 & (x, t) \in (x_1^\ell - L_1^y, x_1^\ell) \times T_1^y \\ 0 & (x, t) \in (x_1^\ell - L_1^r, x_1^\ell) \times T_1^r \\ (x_1^\ell - L_1^r - x)/L_1^r & (x, t) \in (x_1^\ell - 2L_1^r, x_1^\ell - L_1^r) \times T_1^r \\ 1 & \text{otherwise,} \end{cases} \quad (28)$$

where $x_1^\ell = 25$ miles is the position of the traffic light, and $a(x, t; x_1^\ell)$ is the deceleration rate of the drivers. The function $a(x, t; x_1^\ell)$ is periodic in time t with period T : its graph is shown in Figure 5. The traffic light cycle has a period of $T = 6$ minutes, which consists of yellow light time T_1^y , red light time T_1^r and green light time T_1^g . During yellow lights, drivers within L_1^y miles (see Figure 5(a)) are decelerating. During red lights, cars within L_1^r miles (see Figure 5(b)) are stopped, and those following are decelerating. During green lights T_1^g (see Figure 5(c)), traffic flows normally. We set the lights length $(|T_1^y|, |T_1^r|, |T_1^g|) = (10, 190, 400)$ seconds and response distances $(L_1^y, L_1^r) = (1, 0.8)$ miles in order to generate significantly oscillating traffic flow.

We carried out simulations using the algorithm described in §4 with EnKF and PF for (i) Eulerian observations only, (ii) Lagrangian observations only, (iii) both Eulerian and Lagrangian observations, and (iv) no observations. The simulation results based on the traffic lights scenario are shown in Figure 6: panels (a) and (b) contain the results for EnKF and PF, respectively. The relative RMSE for the cases (i)-(iii) ranges between 1% and 2% after 3 hours, while the relative RMSE for case (iv) (no observations) is above 7%. The results suggest that our algorithm is capable of assimilating any combination of Eulerian and Lagrangian data and performs well independently of whether we use EnKF or PF.

5.2 Efficacy of Lagrangian data assimilation

In practice, either position or velocity data can be collected through GPS devices. Thus, we are interested in the efficacy and accuracy of different Lagrangian observations. According to the statistical data of GPS devices

Figure 6: Traffic state estimation in the traffic light scenario (28). Shown is the relative RMSE for assimilating no data (yellow dashed dot), Eulerian data (blue dashed), Lagrangian data (magenta dotted), and both Eulerian and Lagrangian data (green solid) using (a) EnKF and (b) PF.

on the internet, the error of position data has a standard deviation around 5.12 meters [22], while the error of velocity data has a standard deviation around 0.0707 m/s [34, 36]. In this section, both EnKF and PF are used to assimilate three groups of different Lagrangian observations in a normal traffic scenarios: (1) positions of vehicles, (2) velocities of vehicles, (3) combined positions and velocities of vehicles. The basic setup for the macroscopic model is the same as that in §5.1, and the viscous Lighthill–Whitham macroscopic model (13) is used to generate normal traffic flow for simulation.

The simulation results are shown in Figure 7. Figure 7(a) shows the relative RMSE of Lagrangian data assimilation for different observation data using EnKF. The error for assimilating position data (solid blue) approaches 3.5%, the error for assimilating velocity data (dashed blue) approaches 0.4% and the error for assimilating both data (dotted blue) approaches 0.1%. In contrast, Figure 7(b) shows the comparison based on PF: the error for assimilating velocity data (dashed magenta) approaches 0.25%, and the other two lines (solid and dotted magenta) approach 0.4%. The phenomenon that assimilating both is less accurate initially is related to the “curse of dimensionality” of PF. The dimension of observations is doubled when assimilating both data, but we are unable to correspondingly increase the ensemble size to compensate.

For both EnKF and PF, we can see that assimilating velocity data is more accurate compared to assimilating position data, which is consistent with the fact that velocity data has smaller observation noise than position data. However, PF has a stronger tolerance of observation noise since the difference of error between assimilating velocity data and position data is much smaller compared to those by EnKF. In addition, the performance of EnKF can be improved with combined observation data, while the performance of PF is limited by the less accurate position data.

5.3 Impact of sensor location on data assimilation

When the Department of Transportation installs sensors on the road, an important consideration is where to install them in order to optimize the amount of traffic information collected by them. In this section, we are interested in studying the impact of the configuration of sensors in the presence of on-/off-ramps or bottlenecks in order to efficiently place sensors near ramps and bottlenecks. The basic setup for the macroscopic model is the same as that in §5.1. We will design two traffic scenarios for investigation: an on-/off-ramps scenario and a stationary bottleneck scenario.

In the on-/off-ramps scenario, we take on-ramps and off-ramps into consideration by adding source and sink

Figure 7: Traffic state estimation comparison for Lagrangian data assimilation based on different observation data from normal traffic flow modelled by (13). Shown is the relative RMSE for (a) EnKF and (b) PF, where we assimilate position data only (solid), velocity data only (dotted), and combined position and velocity data (dashed).

terms to the viscous Lighthill–Whitham equation (13):

$$\rho_t + (\rho v_G(\rho))_x = \epsilon \rho_{xx} + \sum_{i \in \mathcal{I}} \rho_i^{\text{on}}(t) f(x - x_i^{\text{on}}) - \sum_{j \in \mathcal{J}} \rho_j^{\text{off}}(t) f(x - x_j^{\text{off}}) \quad (29)$$

where $\{x_i^{\text{on}}\}_{i \in \mathcal{I}}$ and $\{x_j^{\text{off}}\}_{j \in \mathcal{J}}$ are the positions of on-ramps and off-ramps, and $\{\rho_i^{\text{on}}(t)\}_{i \in \mathcal{I}}$ and $\{\rho_j^{\text{off}}(t)\}_{j \in \mathcal{J}}$ are time-dependent fluxes of on-ramps and off-ramps respectively. In this scenario, we place one on-ramp at $x_1^{\text{on}} = 37.5$ miles and one off-ramp at $x_1^{\text{off}} = 12.5$ miles on the road, and set constant flow $\rho_i^{\text{on}}(t) = 2\rho_{\max}$ for the on-ramp and $\rho_i^{\text{off}}(t) = 4\rho_{\max}$ for the off-ramp. Instead of using a delta function $\delta(x - x_i^{\text{on}})$ for sources and $\delta(x - x_i^{\text{off}})$ for sinks, we introduce sech functions $f(x - x_i^{\text{on}}) = \text{sech}(x - x_i^{\text{on}})$ and $f(x - x_i^{\text{off}}) = \text{sech}(x - x_i^{\text{off}})$ to represent the spread effect of ramps. In this scenario, spatial shocks are generated around the on-ramp and off-ramp. Specifically, a density bump appears around the on-ramp, while a density valley appears around the off-ramp. Three comparative experiments are designed where a sensor is installed (1) at the ramp, (2) 0.5 miles upstream to the ramp, and (3) 0.5 miles downstream to the ramp.

In the stationary bottleneck scenario, we aim to model, for instance, construction zones or traffic accidents. By incorporating the bottleneck factor into the viscous Lighthill–Whitham equation, we obtain the formula of macroscopic traffic as:

$$\rho_t + (\rho v_G(\rho) a(x - x_1^b))_x = \epsilon \rho_{xx} \quad (30)$$

where x_1^b is the location of the bottleneck. The quantity $a(x - x_1^b)$ is the bottleneck factor, which can be written as the product of severity coefficient and spread effect $a(x - x_1^b) = c f(x - x_1^b)$. The severity coefficient c reflects the degree of influence of the construction zone or traffic accident, and $f(x - x_1^b)$ represents the spread effect of the bottleneck. We set the bottleneck at $x_1^b = 25$ miles, severity coefficient $c = 1$ and spread effect $f(x) = 1 - 0.5\text{sech}(x)$. In this scenario, a stationary density bottleneck appears around x_1^b . Three comparative experiments are designed where a sensor is installed (1) at the bottleneck, (2) 1 mile upstream to the bottleneck, and (3) 1 mile downstream to the bottleneck.

The simulation results for ramps are shown separately in Figure 8(a) and 8(b), and those for the bottleneck are shown in Figure 8(c). In Figure 8(a), it is observed that the performance of data assimilation is improved by moving a sensor from downstream to upstream of the on-ramp. While in Figure 8(b), unlike the on-ramp case, the performance is improved by moving a sensor from upstream to downstream of the off-ramp. In Figure 8(c), it shows that the performance is improved by moving a sensor from upstream to downstream of the bottleneck. By studying the shape of density on the ring road, we find that a density bump is generated gradually upstream

Figure 8: *Evaluation of various sensors configurations using EnKF. Results for different sensor locations near (a) on-ramps, (b) off-ramps, and (c) bottlenecks are shown, where the solid curves correspond to sensor locations upstream to the target location, the dotted curves represent sensors at the target location, and the dashed curves represent locations downstream to the target location.*

to the on-ramp. Therefore, installing a sensor slightly upstream can provide more traffic information and thereby improve the performance of EnKF. In contrast, a density valley is generated gradually downstream to the off-ramp or the bottleneck, so installing a sensor slightly downstream can provide more traffic information.

In summary, the locations of sensors do impact the performance of data assimilation when using EnKF: the simulation results suggest installing sensors upstream of on-ramps and downstream of off-ramps and bottlenecks. In contrast, the PF performs well regardless of the locations of sensors (simulation results not shown).

5.4 Parameter estimation of traffic-flow models

In previous subsections, we showed that our algorithm successfully assimilates traffic states when the parameters in the underlying traffic model are known. However, in practical applications, these parameters are not known and need to be estimated when traffic data are assimilated. Hence, we are interested in the efficacy of data assimilation when traffic parameter estimation is activated. In this part, we construct two traffic scenarios for investigation: on-/off-ramps scenario and travelling bottleneck scenario.

The basic setup for macroscopic model is the same as §5.1. In on-/off-ramps scenario, the unknown parameters include maximal density $\rho_{\max}$, maximal velocity $v_{\max}$, the fluxes of on-ramp ρ_1^{on} and the fluxes of off-ramp ρ_1^{off} . All of these parameter are constants in the whole simulation. In travelling bottleneck scenario, the position of the bottleneck is time-dependent, and the macroscopic model can be described as:

$$\rho_t + (\rho v_G(\rho) a(x - x_1^b(t)))_x = \epsilon \rho_{xx} \quad (31)$$

where $x_1^b(t)$ is the location of travelling bottleneck, which is assumed to move back and forth periodically according to the formula $x_1^b(t) = L/2 + L/4 \cos(2\pi t/3)$. The unknown parameters include maximal density $\rho_{\max}$, maximal velocity $v_{\max}$, taken to be constant, and the bottleneck location $x_1^b(t)$, which is time-dependent.

As shown in Figure 9(a), the relative RMSE by using EnKF (blue dashed line) and PF (magenta dashed line) hovers around 3% and 2%, respectively, after 3 hours of assimilating Eulerian observations. In Figure 9(b), the relative RMSE by using EnKF (blue dash line) and PF (magenta dash line) is around 10% and 20%, respectively, after 3 hours of assimilating Lagrangian observations. Observations of the flux, which is the product of density and velocity, can balance the estimation of $\rho_{\max}$ and $v_{\max}$ and thereby provide accurate estimation with an error below 5%. However, Lagrangian observations, especially of velocity data, cannot provide as good a balance between density and velocity as Eulerian observations, and we indeed find that Lagrangian observations are less efficient when the parameters are unknown. We also see in Figure 9(c) and 9(d) that the PF estimator (magenta dashed) is more sensitive when considering the estimated fluxes as these oscillate around the true flux value near the ramps.

In the bottleneck scenario, the conclusions are similar as those in the ramps scenario. The relative RMSEs are below 5% in Figure 10(a) and below 9% in Figure 10(b), which means our approach has provided a very

Figure 9: *Estimation of traffic states and parameters in the ramps scenario: shown are the relative RMSE for assimilating (a) Eulerian and (b) Lagrangian observations as well as the results of parameter estimation for (c) Eulerian and (d) Lagrangian observations. Estimated are the maximal density ρ_{max} , the maximal velocity v_{max} , the flux of the on-ramp ρ_1^{on} , and the flux of the off-ramp ρ_1^{off} .*

good state estimator. In addition, Figure 10(c) and 10(d) demonstrate the dynamical tracking capability of our approach for the time-varying parameter $x_1^b(t)$.

In summary, our approach is capable of providing accurate estimations for both constant and time-varying parameters. Generally Lagrangian observations are more efficient when estimating traffic state, but Eulerian observations appear to be slightly more efficient when parameter estimation is activated.

5.5 Estimate microscopic traffic data using macroscopic model

The simulations shown above used the same traffic model for (1) generating the truth and (2) assimilating data in our algorithm. We are interested in testing the sensitivity with respect to the model used in the data assimilation algorithm in a controlled environment. In this section, we therefore generate the truth using the microscopic model introduced in §2.1 and assimilate data using the macroscopic Lighthill–Whitham equation. More specifically, we found in §2.1 two different traffic scenarios in the microscopic model, obtained by different choices of parameters and initial conditions: in the first scenario, a square wave evolves into a shock with gradually decreasing amplitude (a solution that can also be found in the macroscopic Lighthill–Whitham model), while the second scenario describes the emergence of a travelling wave with fixed amplitude (which cannot be generated by the Lighthill–Whitham model). In our simulations below, we will study each scenario separately using EnKF to assimilate Lagrangian observations from the microscopic model to estimate the macroscopic traffic density. To compare the predicted traffic densities with the truth, we transform the microscopic traffic data into continuous densities on the ring road.

Figure 10: Estimation of traffic states and parameters in the travelling bottleneck traffic scenario shown is the relative RMSE for assimilating (a) Eulerian and (b) Lagrangian observations and the results of parameter estimation for (c) Eulerian and (d) Lagrangian observations, where we estimate the maximal density ρ_{max} , the maximal velocity v_{max} , and the time-dependent bottleneck location x_1^b .

The simulation results for the first scenario are shown in Figure 11. We observe that the estimated density (green circled line) gradually converges to the true value (blue starred line). This is plausible as the viscous Lighthill–Whitham model can generate a travelling shock with decreasing amplitude as demonstrated in §2.2. With the aid of data assimilation, the macroscopic model provides an accurate estimation for the travelling shock wave transferred from microscopic traffic data.

The simulation results for the second scenario are shown in Figure 12. We observe that the estimated density (green circles) is smoothed out rapidly and does not converge to the true density value (blue pluses). Unlike the previous case, the macroscopic model cannot provide an accurate estimation even though data assimilation is used. This is not unexpected as the viscous Lighthill–Whitham model cannot generate a travelling square wave shape with stable amplitude and is therefore not capable of reproducing the correct density profile.

In summary, a macroscopic model may produce inaccurate results if the model is not capable of generating dynamical features present in the traffic flow data that need to be assimilated.

5.6 Parameter estimation for real traffic data

In addition to estimating simulated traffic states and parameters, we are interested in applying the developed approach to real traffic data: in this section, we use data from the Minnesota Department of Transportation [35].

A freeway strip of I-35E with length 7.175 miles between Main Street and Hwy 96E is considered. There are 14 consecutive detectors S1535-S1548 along this stretch, as well as 3 on-ramps and 1 off-ramp (see Figure 13). We

Figure 11: *Estimating microscopic data in scenario 1 using EnKF. True and estimated density values are represented by blue pluses and green circles, respectively.*

Figure 12: *Estimating microscopic data in scenario 2 using EnKF. True and estimated density values are represented by blue pluses and green circles, respectively.*

A Strip of Highway I-35E

Figure 13: A strip of highway I-35E in Minnesota. Cars are moving from left to right. Sensors, on-ramps, and off-ramps are labelled using blue stars, red pluses, and green circles, respectively.

Figure 14: Relative RSME (a) and parameter estimate (b) are shown for data assimilation of real traffic data taken during late night: observations are taken from all sensors (solid), half the sensors (dotted), or no sensors (dashed).

picked two representative time periods from a day for simulation: one is late night (00:01am-00:30am) when free flow is expected, and the other is rush hour (05:31pm-06:00pm) when traffic jams are expected. The Minnesota Department of Transportation provides traffic density, velocity and flow data collected from the sensors, and the density data of ramps. All traffic data are collected each minute.

We use a macroscopic model that takes the on- and off-ramps into account, and assimilate the flux data collected from the sensors using EnKF. We also estimate the unknown maximal density $\rho_{\max}$ and the maximal velocity $v_{\max}$. The density profiles of the ramps are artificially taken as unknown and are replaced by the average density value during a period. In contrast to the simulation presented in previous sections, we use an adaptive inflation algorithm [2] here.

The simulation results for late night traffic are shown in Figure 14. The relative RMSE by assimilating all observation approaches 15%, while the relative RMSE by not assimilating observations approaches 40%. The parameter estimation in 14(b) shows that estimated maximal velocity $\hat{v}_{\max}$ is approximately 85 miles/hour, and the maximal density $\hat{\rho}_{\max}$ decreases to below 30 cars/mile. This is a reasonable description of late night traffic.

The simulation results for rush hour traffic are shown in Figure 15. The approach results in a decrease of error from 15% to 8%. Since the flow of ramps is quite stable during rush hour, the average density value for ramps are close to the true density value which explains why our approach results in a less drastic improvement when compared to the previous case. In Figure 15(b), the estimated maximal velocity $\hat{v}_{\max}$ is still approximately 85 miles/hour, but the maximal density $\hat{\rho}_{\max}$ increases to 150 cars/mile, which is characteristic of traffic flow during rush hour.

Figure 15: *Relative RSME (a) and parameter estimate (b) are shown for data assimilation of real traffic data collected during rush hour: observations are taken from all sensors (solid), half the sensors (dotted), or no sensors (dashed).*

In summary, the efficacy of our approach has been validated by the real traffic data from Minnesota Department of Transportation. Estimation for traffic states is significantly improved in both late night and rush hour conditions, and the estimation for parameter is consistent with characteristics of traffic scenarios.

6 Conclusions and Discussion

In this paper, we studied three aspects of traffic flow prediction: mathematical traffic models, Eulerian and Lagrangian observations, and data assimilation techniques. Specifically, we reviewed microscopic and macroscopic traffic models and used both in our simulations. We presented a formulation that unifies the treatment of Eulerian and Lagrangian observations without requiring a reformulation of the underlying traffic model. We then studied how well this approach to data assimilation works for estimating traffic states as well as parameters using EnKF and PF.

The initial motivation was to develop a consistent data assimilation approach for both sensor and GPS data by appending the differential equations for the positions of vehicles to the traffic model. We also introduced new methods for calculating the Kalman gain matrix $\mathbf{K}$ and the localization function when implementing the EnKF (see §3.2 and §4.3). Unlike other data assimilation methods, our approach allows us to simultaneously assimilate Eulerian and Lagrangian data.

The Lagrangian data assimilation method was implemented in multiple traffic scenarios in order to estimate traffic states and unknown parameters. We demonstrated the accuracy and efficacy of the proposed approach using both simulated data and real traffic data from the Minnesota Department of Transportation. We found that Lagrangian data assimilation via PF is far less sensitive to errors in GPS data and the location of sensors compared to EnKF implementations (see §5.3 and §5.4).

In this work, the Lagrangian data assimilation was only applied to simplified traffic scenarios: normal traffic, on/off-ramps, traffic lights and bottlenecks. However, in practice the road network can be very complicated and may include intersections, multiple lanes, etc. An interesting extension would be to expand on the above framework in order to apply Lagrangian data assimilation to more complex and realistic traffic scenarios.

According to our study in §5.5, one limitation in using data assimilation is that the estimation accuracy is reduced if the underlying traffic model cannot reproduce the actual traffic flow. Therefore, an important aspect of using data assimilation in modelling traffic lies in developing accurate models of traffic flow which are capable of capturing the various phenomena observed in real traffic scenarios.

Since Lagrangian data assimilation is very accurate in estimating both constant and time-dependent parameters, a future goal could be to use parameter estimation to relay traffic information to drivers. For instance, the period of traffic lights could be estimated by Lagrangian data assimilation, and thus the waiting time can be provided to drivers. Data assimilation could also be used to predict the location and duration of congestion caused by slow trucks or road constructions, based on which drivers or route-guidance software could effectively plan trips.

Acknowledgements The authors gratefully acknowledge support by the National Science Foundation through grant DMS-1148284.

References

- [1] Federal Highway Administration, *Over-roadway sensor technologies*, <http://www.fhwa.dot.gov/policyinformation/pubs/vdstits2007/05pt2.cfm>.
- [2] J. L. Anderson, *An adaptive covariance inflation error correction algorithm for ensemble filters*, *Tellus* **59A** (2007), 210–224.
- [3] J. L. Anderson, *Spatially and temporally varying adaptive covariance inflation for ensemble filters*, *Tellus A* **61** (2009), no. 1, 72–83.
- [4] J. L. Anderson and S. L. Anderson, *A Monte Carlo implementation of the nonlinear filtering problem to produce ensemble assimilations and forecasts*, *Mon. Wea. Rev.* **127** (1999), no. 12, 2741–2758.
- [5] S. J. Baek, B. R. Hunt, E. Kalnay, and E. Ott, *Local ensemble Kalman filtering in the presence of model bias*, *Tellus* **58A** (2006), no. 3, 293–306.
- [6] M. Bando, K. Hasebe, K. Nakanishi, and A. Nakayama, *Analysis of optimal velocity model with explicit delay*, *Phys. Rev. E* **58** (1998), no. 5, 5429.
- [7] M. Bando, K. Hasebe, A. Nakayama, A. Shibata, and Y. Sugiyama, *Dynamical model of traffic congestion and numerical simulation*, *Phys. Rev. E* **51** (1995), 1035–1042.
- [8] T. Bellsky, J. Berwald, and L. Mitchell, *Nonglobal parameter estimation using local ensemble Kalman filtering*, *Mon. Wea. Rev.* **142** (2014), no. 6, 2150–2164.
- [9] G. Burgers, P. J. van Leeuwen, and G. Evensen, *Analysis scheme in the ensemble Kalman filter*, *Mon. Wea. Rev.* **126** (1998), 1719–1724.
- [10] P. Carter, P. Christiansen, Y. Gaididei, C. Gorria, B. Sandstede, M. Sorensen, and J. Starke, *Multi-jam solutions in traffic models with velocity-dependent driver strategies*, *SIAM J. Appl. Math.* **74** (2014), no. 6, 1895–1918.
- [11] D. P. Dee and A. M. Da Silva, *Data assimilation in the presence of forecast bias*, *Q. J. Roy. Meteor. Soc.* **124** (1998), no. 545, 269–295.
- [12] A. Doucet, N. De Freitas, and N. Gordon, *An introduction to sequential Monte Carlo methods*, *Sequential Monte Carlo methods in practice*, Springer, 2001, pp. 3–14.
- [13] A. Doucet, S. Godsill, and C. Andrieu, *On sequential Monte Carlo sampling methods for Bayesian filtering*, *Stat. Comput.* **10** (2000), 197–208.
- [14] G. Evensen, *Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics*, *J. Geophys. Res.: Oceans* **99** (1994), no. C5, 10143–10162.
- [15] G. Evensen, *The ensemble Kalman filter: theoretical formulation and practical implementation*, *Ocean dynamics* **53** (2003), no. 4, 343–367.

- [16] Y. Gaididei, R. Berkemer, C. Gorria, P. L. Christiansen, A. Kawamoto, T. Shiga, M. P. Sørensen, and J. Starke, *Complex spatiotemporal behavior in a chain of one-way nonlinearly coupled elements*, Discr. Cont. Dynam. Syst. S **4** (2011), no. 5, 1167–1179.
- [17] Y. B. Gaididei, R. Berkemer, J. G. Caputo, P. L. Christiansen, A. Kawamoto, T. Shiga, M. P. Sørensen, and J. Starke, *Analytical solutions of jam pattern formation on a ring for a class of optimal velocity traffic models*, New J. Phys. **11** (2009), 073012, 19 pp.
- [18] Y. B. Gaididei, C. Gorria, R. Berkemer, P. L. Christiansen, A. Kawamoto, M. P. Sørensen, and J. Starke, *Stochastic control of traffic patterns*, Netw. Heterog. Media **8** (2013), no. 1, 261–273.
- [19] Y. B. Gaididei, C. Gorria, R. Berkemer, A. Kawamoto, T. Shiga, P. L. Christiansen, M. P. Sørensen, and J. Starke, *Controlling traffic jams by time-modulating the safety distance*, Phys. Rev. E (2013).
- [20] G. Gaspari and S. E. Cohn, *Construction of correlation functions in two and three dimensions*, Q. J. Roy. Meteor. Soc. **125** (1999), no. 554, 723–757.
- [21] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, *Novel approach to nonlinear-non-Gaussian Bayesian state estimation*, IEE Proc-F **140** (1993), no. 2, 107–113.
- [22] GPS Gov, *GPS accuracy*, <http://www.gps.gov/systems/gps/performance/accuracy/>.
- [23] B. D. Greenshields, W. Channing, H. Miller, et al., *A study of traffic capacity*, Highway research board proceedings, vol. 1935, National Research Council (USA), Highway Research Board, 1935.
- [24] T. M. Hamill, J. S. Whitaker, and C. Snyder, *Distance-dependent filtering of background error covariance estimates in an ensemble Kalman filter*, Mon. Wea. Rev. **129** (2001), no. 11, 2776–2790.
- [25] W. K. Hastings, *Monte Carlo sampling methods using Markov Chains and their applications*, Biometrika **57** (1970), 97–109.
- [26] D. Helbing, *Traffic and related self-driven many-particle systems*, Rev. of Mod. Phys. **73** (2001), no. 4, 1067.
- [27] P. Henrici, *Applied and computational complex analysis, discrete Fourier analysis, Cauchy integrals, construction of conformal maps, univalent functions*, vol. 3, John Wiley & Sons, 1993.
- [28] P. L. Houtekamer and H. L. Mitchell, *Data assimilation using an ensemble Kalman filter technique*, Mon. Wea. Rev. **126** (1998), 796–811.
- [29] P. L. Houtekamer and H. L. Mitchell, *A sequential ensemble Kalman filter for atmospheric data assimilation*, Mon. Wea. Rev. **129** (2001), no. 1, 123–137.
- [30] A. Kong, J. Liu, and W. Wong, *Sequential imputations and Bayesian missing data problems*, J. Am. Stat. Assoc. **89** (1994), 278–288.
- [31] M. J. Lighthill and G. B. Whitham, *On kinematic waves. ii. a theory of traffic flow on long crowded roads*, Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences **229** (1955), no. 1178, 317–345.
- [32] J. Liu and M. West, *Combined parameter and state estimation in simulation-based filtering*, Sequential Monte Carlo methods in practice, Springer, 2001, pp. 197–223.
- [33] R. N. Miller, M. Ghil, and F. Gauthiez, *Advanced data assimilation in strongly nonlinear dynamical systems*, J. Atmos. Sci. **51** (1994), no. 8, 1037–1056.
- [34] GPS Information Net, *GPS speed*, <http://www.gpsinformation.net/main/gpsspeed.htm>.
- [35] Minnesota Department of Transportation, *Traffic forecasting & analysis*, <http://www.dot.state.mn.us/traffic/data/>.

- [36] GPS Action Replay, *GPS and speed measurement*, <http://gpsactionreplay.free.fr/index.php?menu=6&choice=5>.
- [37] P. I. Richards, *Shock waves on the highway*, Oper. Res. **4** (1956), no. 1, 42–51.
- [38] H. Salman, L. Kuznetsov, C. K. R. T. Jones, and K. Ide, *A method for assimilating Lagrangian data into a shallow-water-equation ocean model*, Mon. Wea. Rev. **134** (2006), no. 4, 1081–1101.
- [39] N. Santitissadeekorn and C. K. R. T. Jones, *Two-state filtering for joint state-parameter estimation*, arXiv preprint arXiv:1403.5989 (2014).
- [40] C. Snyder, *Particle filters, the optimal proposal and high-dimensional systems*, Proceedings of the ECMWF Seminar on Data Assimilation for Atmosphere and Ocean, 2011.
- [41] C. Snyder, T. Bengtsson, P. Bickel, and J. Anderson, *Obstacles to high-dimensional particle filtering*, Mon. Wea. Rev. **136** (2008), no. 12, 4629–4640.
- [42] C. Snyder, T. Bengtsson, and M. Morzfeld, *Performance bounds for particle filters using the optimal proposal*, Mon. Wea. Rev. **143** (2015), no. 11, 4750.
- [43] P. J. van Leeuwen, *Particle filtering in geophysical systems*, Mon. Wea. Rev. **137** (2009), 4089–4114.
- [44] D. B. Work, S. Blandin, O.-P. Tossavainen, B. Piccoli, and A. M. Bayen, *A traffic model for velocity data assimilation*, Appl. Math. Res. Express **2010** (2010), no. 1, 1–35.
- [45] D. B. Work, O.-P. Tossavainen, S. Blandin, A. M. Bayen, T. Iwuchukwu, and K. Tracton, *An ensemble Kalman filtering approach to highway traffic estimation using GPS enabled mobile devices*, 2008 47th IEEE Decis. Contr. P., IEEE, 2008, pp. 5062–5068.
- [46] X. Yang and T. Delsole, *Using the ensemble Kalman filter to estimate multiplicative model parameters*, Tellus **61A** (2009), no. 5, 601–609.
- [47] Y. Yuan, H. Van Lint, F. Van Wageningen-Kessels, and S. Hoogendoorn, *Network-wide traffic state estimation using loop detector and floating car data*, J. Intell. Transport. S.: Technology, Planning, and Operations **18** (2014), no. 1.
- [48] Y. Yuan, J. W. C. Van Lint, R. Eddie Wilson, F. Van Wageningen-Kessels, and S. P. Hoogendoorn, *Real-time Lagrangian traffic state estimator for freeways*, IEEE Trans. Intell. Transp. Syst **13** (2012), no. 1, 59–70.
- [49] H. M. Zhang, *Driver memory, traffic viscosity and a viscous vehicular traffic flow model*, Transport. Res. B-Meth. **37** (2003), no. 1, 27–41.