Heterogeneous traffic estimation with particle filtering

Yanbing Wang[1] and Daniel B. Work[1]

Abstract

This article considers the state estimation problem for heterogeneous traffic (a mixed flow composed of vehicles with distinct driving behaviors). Heterogeneous traffic is characterized by loose lane discipline and various vehicle sizes. The interactions between vehicle classes display unique features that are distinct from lane-based homogeneous traffic flows. Modeling and estimation methods for heterogeneous traffic, however, still remain relatively unexplored. This article adopts a particle-filtering approach to sequentially estimate the traffic state with a heterogeneous traffic flow model that allows overtaking and creeping behaviors. Numerical experiments are introduced to evaluate the estimator performance, indicating that the filter can reduce estimation errors by up to 48% when compared to pure forward simulation of the model.

I. Introduction

While traffic management applications rely on traffic estimation and control for homogeneous traffic, a new generation of tools may be needed to facilitate management of increasingly complex heterogeneous traffic. This is because the diverse mobility ecosystem may have significantly different operating rules compared to a passenger-car-only traffic environment. Consequently an open question is how to accurately model and estimate traffic conditions when the flow is heterogeneous.

The main contribution of this work is to develop a traffic state estimator for heterogeneous traffic that allows overtaking and creeping between vehicle classes. The traffic state estimator uses a creeping traffic flow model [1] for the forward prediction step, and a particle filter to track the traffic state. Numerical experiments demonstrate the potential of the filter to estimate the traffic state when complex behaviors exist in the flow, and significantly reduce the error compared to pure forward simulations of an approximate model. While the examples considered here are numerical experiments, it represents a first step towards traffic state estimation of complex flows.

[1]Department of Civil and Environmental Engineering and the Institute for Software Integrated Systems, Vanderbilt University, Nashville, TN 37240

The remainder of the work proceeds as follows. Traffic modeling and estimation techniques are reviewed in Section II; the mathematical preliminaries of the multi-class creeping model and the particle filter are presented in Section III. Section IV describes the numerical experiments, estimation results, and the performance evaluation. Finally, the conclusion and potential future works are highlighted in Section V.

II. Related work

We first summarize the prior literature related to heterogeneous traffic modeling, and traffic estimation techniques.

A. Modeling heterogeneous traffic

After the seminal *Lighthill-Whitham-Richards* (LWR) homogeneous traffic flow model [2], [3], many model extensions have been introduced for lane-following traffic that considers vehicles with distinct driving behaviors and vehicle sizes. For example, Wong & Wong [4] distinguished vehicle classes by assuming a unique velocity function for each class. Chanut & Leclercq [5] considered a two-class flow where the slow vehicles (such as trucks) behave like a moving bottleneck. In the work by Ngoduy & Liu [6], each vehicle class is defined by the desired free flow speed, allowing overtaking in free flow to occur, while in the congested regime, all vehicles travel at the same speed. In the Fastlane model of van Lint et al. [7] (later extended Schreiter et al. [8]), the traffic dynamics are calculated in terms of state-dependent *passenger-car equivalents* (PCE), rather than constant PCE values assumed in earlier works. Chalons et al. [9] proposed a coupled microscopic-macroscopic model to account for the effect of large and slow-moving vehicles. These models characterize lane following dynamics when vehicles with different sizes and driving behaviors are present, and can allow various forms of overtaking to be captured.

Compared to lane-based traffic, modeling of heterogeneous traffic characterized by loose lane discipline and a wide range of vehicle characteristics is relatively less studied. Benzoni-Gavage & Colombo [10] introduced the n-populations model by making a connection between the equilibrium speed of each class and the total occupied space. Nair et al. [11] developed the porous model which allows vehicles to move through the "pores" defined by the free space between other vehicles in a disordered flow. The porous model specifies the relationship between density and speed with respect to the distribution of the pore space created by larger vehicles. This assumption allows not only overtaking, but also creeping to be captured. Inspired by these models, the creeping model [1] explicitly defines class-specific velocity functions and jam densities to capture both overtaking and creeping features. More recently, Gashaw et al. [12] re-examined the porous model and provided an analytical expression for the space distribution such that the model is specifically tailored to a mixed flow of cars and powered two-wheelers.

There are many other methods to model heterogeneous traffic such as using multi-class multi-lane mesoscopic modeling (Costeseque & Duret [13]), and cellular automata modeling (Mallikarjuna & Rao [14]). Interested readers can refer to the extended reviews in [15]–[18].

B. *Traffic state estimation*

Model based traffic state estimation and data assimilation approaches are often posed as online filtering problems. The approaches rely on Bayesian inference to estimate the most likely state by combining the uncertain model prediction with the noisy measurement data. Szeto & Gazis [19] first proposed the *Kalman filter* (KF) and the *extended Kalman filter* (EKF) for the application on traffic state estimation. The EKF is an extension of the KF for differentiable nonlinear systems and has been broadly applied to traffic state estimation [20]–[22]. For non-differentiable models such as the CTM or its extensions, the *unscented Kalman filter* (UKF) and the *ensemble Kalman filter* (EnKF) are also applied [23]–[25]. These Kalman-based filters, however, are minimal variance estimators which limits their application on traffic estimation problems that can generate multi-modal error distributions [25], even though error bounds can be derived [26]. Therefore, a fully Monte Carlo sampling-based filter, the particle filter [27], [28] is adopted [29]–[32]. Readers can refer to a complete review of traffic estimation techniques and the coupled flow models [33].

III. Preliminaries

In this section, we briefly review the two-class creeping model and the Bayesian state estimation framework for the construction of a model based estimator.

A. *A two-class creeping model*

The two-class creeping model [1] is a system of scalar conservation laws that governs the flow of each vehicle class:

$$\frac{\partial \rho_j(x,t)}{\partial t} + \frac{\partial \rho_j(x,t) V_j(r(x,t))}{\partial x} = 0, \;\; j \in \{1,2\}, \tag{1}$$

where $\rho_j(x,t)$ denotes the density of each vehicle class (indexed by j) at time t and space x. The velocity function for each class $V_j(\cdot)$ is distinct for each vehicle class and depends on the total density $r = \sum_j \rho_j$. For simplicity, in this work we consider the following velocity functions:

$$V_j(r) = v^m \left(1 - \frac{r}{r_j^m}\right), \; j \in \{1,2\}, \tag{2}$$

where v^m is the speed limit common to all road users. The class specific jam densities $r_j^m \in \{r_1^m, r_2^m\}$ control the total density r at which the individual vehicle classes come to a complete stop. If $r_1^m \neq r_2^m$, then one vehicle class will be able to creep through traffic while the other class is completely stopped. In the simplified setting considering a linear velocity function, the three parameters completely define the two class creeping flow.

A discrete approximation of (1) is defined as follows:

$$\rho_{i,j}^{k+1} = \rho_{i,j}^k + \frac{\Delta t}{\Delta x}\left(F_{i-1/2,j}^k - F_{i+1/2,j}^k\right), \; j \in \{1,2\}, \tag{3}$$

where $\rho_{i,j}^k$ represents the density of class j in the ith cell at time k. The terms $F_{i-1/2,j}^k$ and $F_{i+1/2,j}^k$ are the inflow and outflow of class j traffic via the upstream and downstream boundaries of cell i at time k. Computing the boundary flow $F_{i-1/2,j}^k$ requires solving a Riemann problem between two adjacent boundary cells, see [1] for a complete description.

B. Bayesian traffic state estimation

The Bayesian approach to traffic state estimation evaluates the posterior distribution of the system state given a prior state estimate and measurement data. The state of the system $\boldsymbol{x}^k$ for model (3) is defined as:

$$\boldsymbol{x}^k = \left[\rho_{1,1}^k, \ldots, \rho_{i_{max},1}^k, \rho_{1,2}^k, \ldots, \rho_{i_{max},2}^k\right]^T, \tag{4}$$

where i_{max} is the number of cells in the discretization. The sequential estimation problem based on the state space model and the measurement model is shown in Figure 1. The discrete-time creeping model defined in (3) is denoted as $f(\cdot)$ and it propagates the traffic state to the next time step. $\boldsymbol{y}^k$ is the observed state obtained from the measurement equation $h(\cdot)$, which defines the mapping between the traffic state $\boldsymbol{x}^k$ and the measurements $\boldsymbol{y}^k$ at time k. The terms $\boldsymbol{w}^k \sim \mathcal{N}(0,R)$ and $\boldsymbol{v}^k \sim \mathcal{N}(0,Q)$ denote the unbiased process noise and measurement noise at time k with assumed covariance matrices R and Q.

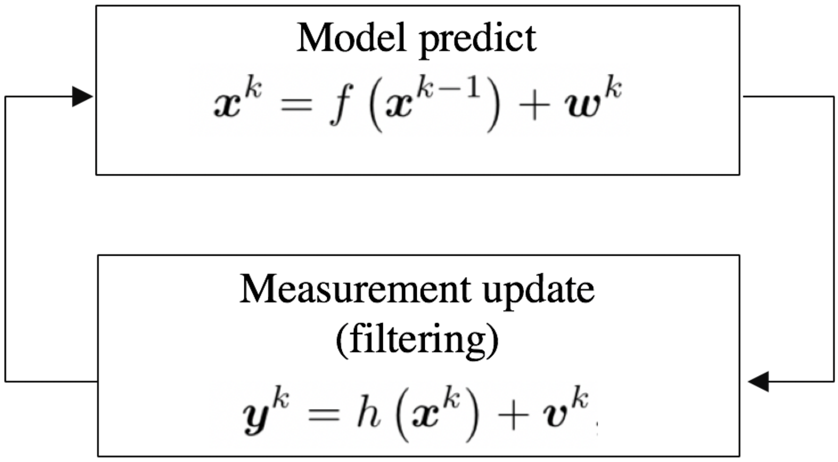

Fig. 1: Sequential state estimation diagram.

The particle filter begins with a collection of L samples (referred to as particles) from the initial state probability density function $p(\boldsymbol{x}^0)$, where $\boldsymbol{x}^0$ is a random variable representing the state vector at time $k = 0$. Each realization of the state vector is denoted as $\boldsymbol{x}_l^0$ (l = $1, \cdots, L$). At each time step $k = 1, 2, \cdots, k_{max}$, all the particles are propagated to the next time step using the discrete time process model $f(\cdot)$, then re-weighted based on the likelihood after measurements are obtained. The particle filter approximates the conditional probability distribution of $\boldsymbol{x}^k$ given measurements $\boldsymbol{Y}^k = [\boldsymbol{y}^1, \boldsymbol{y}^2, \ldots, \boldsymbol{y}^k]$, denoted as $p(\boldsymbol{x}^k|\boldsymbol{Y}^k)$, so that the state $\boldsymbol{x}^k$ is recursively updated according to:

$$\begin{aligned} p(\boldsymbol{x}^k|\boldsymbol{Y}^{k-1}) &= \int p(\boldsymbol{x}^k|\boldsymbol{x}^{k-1})p(\boldsymbol{x}^{k-1}|\boldsymbol{Y}^{k-1})d\boldsymbol{x}^{k-1} \\ p(\boldsymbol{x}^k|\boldsymbol{Y}^k) &= \frac{p(\boldsymbol{y}^k|\boldsymbol{x}^k)p(\boldsymbol{x}^k|\boldsymbol{Y}^{k-1})}{p(\boldsymbol{y}^k|\boldsymbol{Y}^{k-1})}, \end{aligned} \tag{5}$$

where $p(\boldsymbol{x}^k|\boldsymbol{Y}^{k-1})$ and $p(\boldsymbol{x}^k|\boldsymbol{Y}^k)$ are the prior and the posterior distribution of the state, respectively. A complete description of the particle filtering algorithm can be found in standard references such as [34].

IV. Numerical experiments

In this section, we assess the capability of the particle filter to track heterogeneous traffic flows in complex scenarios when creeping and overtaking occur. We first present two numerical experiments in which creeping traffic and overtaking traffic are observed, and then present a sensitivity analysis.

In each experiment, we use the creeping model (3) to evolve the *true state* (also referred to as the ground truth), which is to be estimated. Then, we run a forward model simulation with an *approximate model*, which also uses a creeping model (3), but with initial conditions, boundary conditions, and model parameters that differ from what is used in the true state. In the approximate model, these errors are intentionally introduced to capture the fact that in real deployments on real world experiments, the models often retain errors in parameter choices and initial and boundary conditions, see [35]. Finally we use the same approximate model within the particle filtering framework to estimate the state using noisy measurements of the true state. Because the same approximate model is used in both the forward simulation and within the particle filter, we can compare the total reduction in error as a result of the measurement correction step in the particle filter.

To quantify the error, we compute the *mean absolute error* (MAE) between *i*) the true state and the state computed using pure forward simulation of the approximate model; and *ii*) the true state and the state estimated using the particle filter. Let $\hat{\rho}_{i,j}^k$ denote the simulated or estimated traffic state of vehicle class j at time k, and let $\rho_{i,j}^k$ denote the true state. The MAE for each vehicle class is computed as:

$$MAE_j = \frac{1}{k_{max} \times i_{max}} \sum_{i=1}^{i_{max}} \sum_{k=1}^{k_{max}} \mid \rho_{i,j}^k - \hat{\rho}_{i,j}^k \mid . \tag{6}$$

A. *Overview of experiments*

In the two experiments presented next, we consider a stretch of a roadway discretized into $i_{max} = 40$ cells and the experiments are run for $k_{max} = 160$ time steps. The roadway is shared between two vehicle classes with the density of small creeping vehicles denoted by class $j = 1$, and the large vehicles are denoted by class $j = 2$.

For particle filtering runs, we consider that noisy density measurements are obtained in an upstream, intermediate, and downstream cells (indexed by $i = 3, 20,$ and 37), and we fix the number of particles at $L = 1,500$. The process noise and the initial noise are assumed to be Gaussian zero mean and a standard deviation of 0.1 and 0.06, respectively. A strong spatial correlation is assumed in the covariance. The measurement noise follows a normal distribution with zero mean and 0.08 standard deviation. The boundary noises assumed in the estimator are drawn from a Weibull distribution with the *scale parameter* set as 0.06 and the *shape parameter* set as 4. The Weibull distribution is used to prevent negative density state realizations on the boundary. Source code for the numerical experiments is available at https://github.com/yanb514/heterogeneous_traffic_estimation.git.

parameter	**value (true model)**	**value (approximate model)**
v^m	1.8	1.7
r_1^m	1.5	1.4
r_2^m	1.0	1.0

TABLE I: Model parameters for experiment 1: creeping traffic.

B. Experiment 1: Creeping traffic

This experiment is designed such that creeping traffic is observed in the true state evolution. The initial conditions and boundary conditions are set such that the large vehicles begin ahead of the small creeping vehicles. The downstream boundary condition is sufficiently large to cause the large vehicles to come to a complete rest, while the smaller vehicles are still able to advance.

The problem setup is as follows. The parameters of the true model, and the approximate model used for pure forward simulation and for filtering are set according to the values in Table I.

The initial condition for the true model is given as

$$\rho_{i,1}^0 = \begin{cases} 0.5 & i \in [1, 14] \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

$$\rho_{i,2}^0 = \begin{cases} 0.7 & i \in [15, 40] \\ 0 & \text{otherwise,} \end{cases} \tag{8}$$

which places the smaller faster vehicles initially behind the larger slower ones. The true upstream boundary conditions are assumed to follow a time varying square wave oscillating at low densities:

$$\begin{aligned} \rho_{0,1}^k &= \text{sgn}\left(\sin(0.07\ k)\right) \times 0.06 + 0.06 \\ \rho_{0,2}^k &= \text{sgn}\left(\sin(0.07\ k)\right) \times 0.07 + 0.07. \end{aligned} \tag{9}$$

As the traffic evolves and the vehicles approach the end of the domain, the downstream boundary conditions cause a bottleneck for the large vehicles to be activated. The true downstream conditions are set as $\rho_{i_{max}+1,1}^k = 0$, and $\rho_{i_{max}+1,2}^k = 1$. This scenario permits the small vehicles to continue to advance even as the large vehicles begin accumulating upstream of the bottleneck and eventually come to a complete stop (see Figure 2).

In the approximate model used for forward simulation and for estimation, and the initial conditions have errors compared to the true model (in addition to the parameter changes in Table I). The initial condition in the approximate model is set as:

$$\hat{\rho}_{0,1}^{0}=\begin{cases}0.6 & i\in[1,14]\\ 0 & \text{otherwise}\end{cases}\tag{10}$$

$$\hat{\rho}_{0,2}^{0}=\begin{cases}0.6 & i\in[15,40]\\ 0 & \text{otherwise},\end{cases}\tag{11}$$

while the boundary condition is assumed to follow:

$$\begin{aligned}\hat{\rho}_{0,1}^{k}&=\mathrm{sgn}(\sin(0.07k))\times 0.03+0.03\\ \hat{\rho}_{0,2}^{k}&=\mathrm{sgn}(\sin(0.07k))\times 0.03+0.03.\end{aligned}\tag{12}$$

Finally, in the approximate model we set an empty downstream for the small vehicles $\hat{\rho}_{i_{max}+1,1}^{k}=0$ and congestion for the large vehicles $\hat{\rho}_{i_{max}+1,2}^{k}=1$.

Note that the emphasis of the experiments is to illustrate the potential reduction in error achievable by running the filter on the (erroneous) approximate model, compared to simply evolving the state according to the approximate model. In this context, Figure 2 shows the distinction between the true state and the forward simulation by the approximate model. Without the measurement correction, the approximate model simply propagates the state forward with the assumed initial conditions and the model parameters, and further deviates from the true state when the time-varying boundary conditions take place.

The results of the particle filter are shown in Figure 3 at several snapshots in time. The optimal estimates (green lines) are the mean value of particle estimates (black dots), and are shown to trace the true state (black lines) closely throughout the experiment. The large vehicles accumulate at the downstream boundary around $k=20$, and become fully congested at $k=140$. The small vehicles continue to slowly maneuver through the gaps of large vehicles to the front of the queue, both in the true state and in the estimator.

To quantify the estimation performance, the MAE of the particle filter using the approximate model is compared to the MAE of the forward simulation of the approximate model. As shown in Table II, the particle filter provides a significant reduction in MAE compared to a pure forward simulation of the

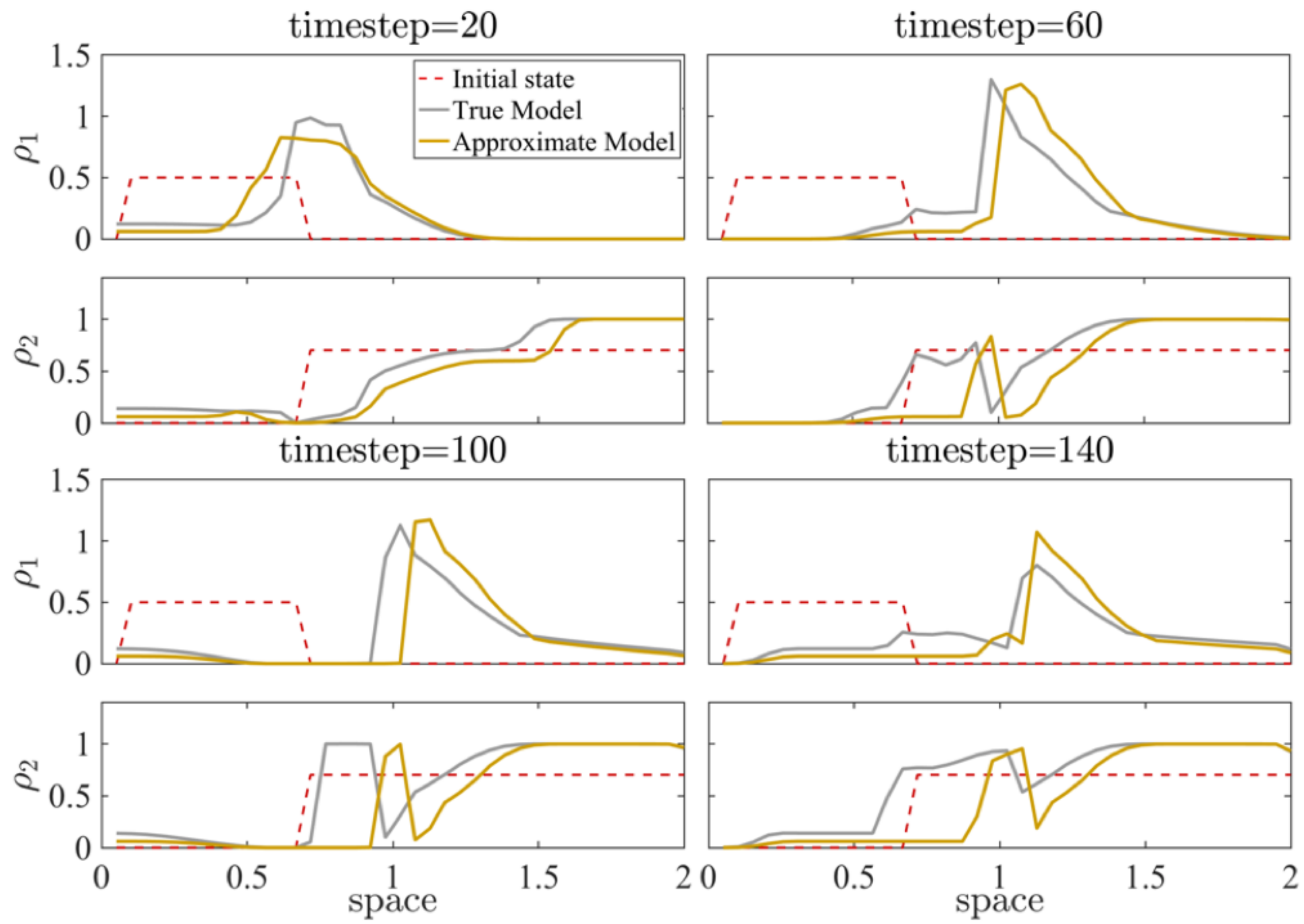

Fig. 2: Snapshots of the true state evolution and the forward simulation in experiment 1: creeping traffic.

State variable	**Forward sim (MAE)**	**Filter (MAE)**	**MAE reduction (%)**
$\rho_{i,1}^{k}$	0.093	0.081	12.9
$\rho_{i,2}^{k}$	0.159	0.106	33.3

TABLE II: State MAE under forward simulation and with particle filtering in Experiment 1: creeping traffic.

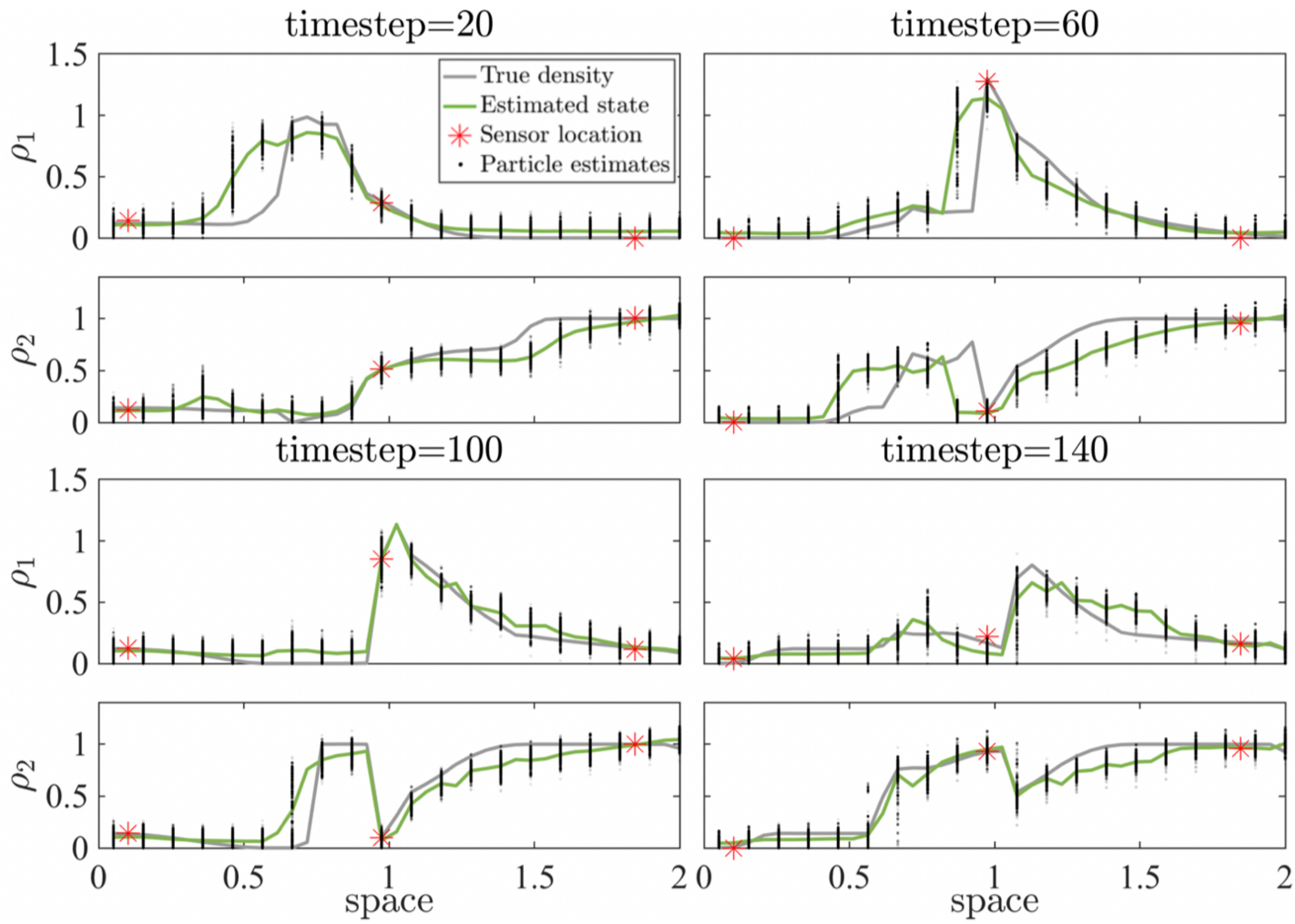

Fig. 3: Snapshots of the true state evolution and the particle filtering estimate in experiment 1: creeping traffic.

approximate model. The reduction of error on the first vehicle class is a more modest 12.9%, while the improvement on the second vehicle class, which has higher error overall is more substantial (33.3%). The experiment overall demonstrates that that the particle filter has the potential to correct for modeling errors and track both vehicle classes even when creeping occurs.

C. Experiment 2: Overtaking traffic

The second experiment assesses the potential of the particle filter to reduce the errors when overtaking occurs. The setup for the experiment is as follows. Like the creeping experiment, the small fast vehicles initially begin behind the larger slower ones, but the downstream boundary conditions are chosen such

parameter	value (true model)	value (approximate model)
v^m	1.8	1.9
r_1^m	1.5	1.6
r_2^m	1.0	1.0

TABLE III: Model parameters for experiment 2: Overtaking traffic.

that both vehicle classes exit the domain as the true state evolves.

In this experiment, the parameters of the true and approximate models are set as defined in Table III.

To achieve overtaking (but not creeping, as both vehicle classes have positive velocities), the true initial condition is set to be:

$$\rho_{i,1}^0 = \begin{cases} 0.5 & i \in [1, 14] \\ 0 & \text{otherwise} \end{cases} \tag{13}$$

$$\rho_{i,2}^0 = \begin{cases} 0.7 & i \in [15, 28] \\ 0 & \text{otherwise,} \end{cases} \tag{14}$$

and in the approximate model we set:

$$\hat{\rho}_{i,1}^0 = \begin{cases} 0.6 & i \in [1, 14] \\ 0 & \text{otherwise} \end{cases} \tag{15}$$

$$\hat{\rho}_{i,2}^0 = \begin{cases} 0.6 & i \in [15, 28] \\ 0 & \text{otherwise.} \end{cases} \tag{16}$$

The upstream boundary conditions in the true and approximate models are set at low density square waves defined in (9) and (12). The downstream boundary conditions are set to be freeflowing in both the true and approximate models so that the vehicles are able to exit the domain.

The results of the overtaking experiment with a pure forward simulation are shown in Figure 4. The small vehicles start at the back of the large vehicles, but move faster than the larger vehicles. Eventually the small vehicles overtake the large vehicles as both classes exit the domain. In the absence of measurements to correct the modeling errors, the forward simulation is not able to accurately track the state. In contrast, Figure 5 illustrates that the particle filter is able to produce an accurate estimation

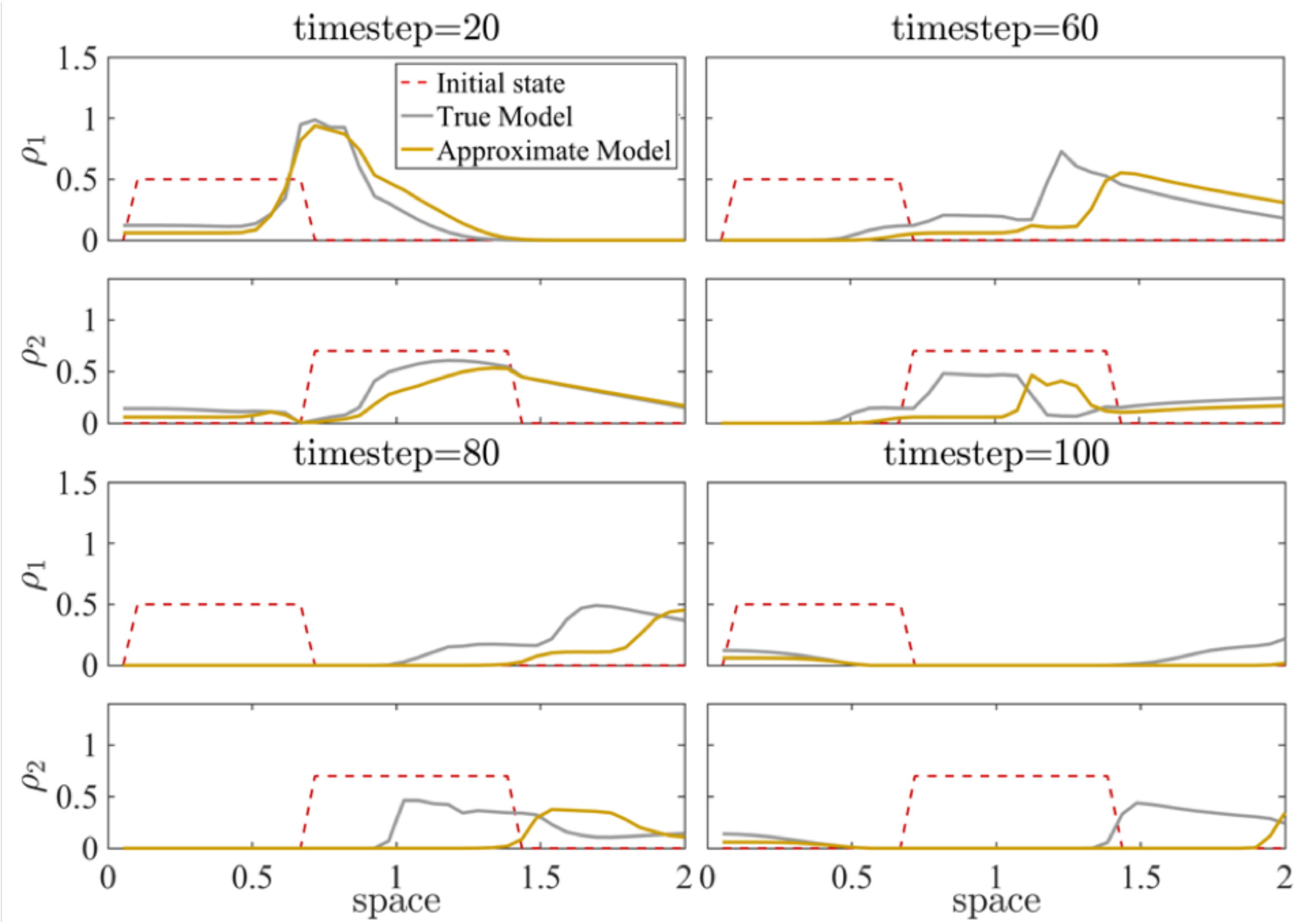

Fig. 4: Snapshots of the true state evolution and the forward simulation in experiment 2: overtaking traffic.

State variable	**Forward sim (MAE)**	**Filter (MAE)**	**MAE reduction (%)**
$\rho_{i,1}^k$	0.075	0.048	35.9
$\rho_{i,2}^k$	0.098	0.051	48.2

TABLE IV: State MAE under forward simulation and with particle filtering in experiment 2: overtaking traffic

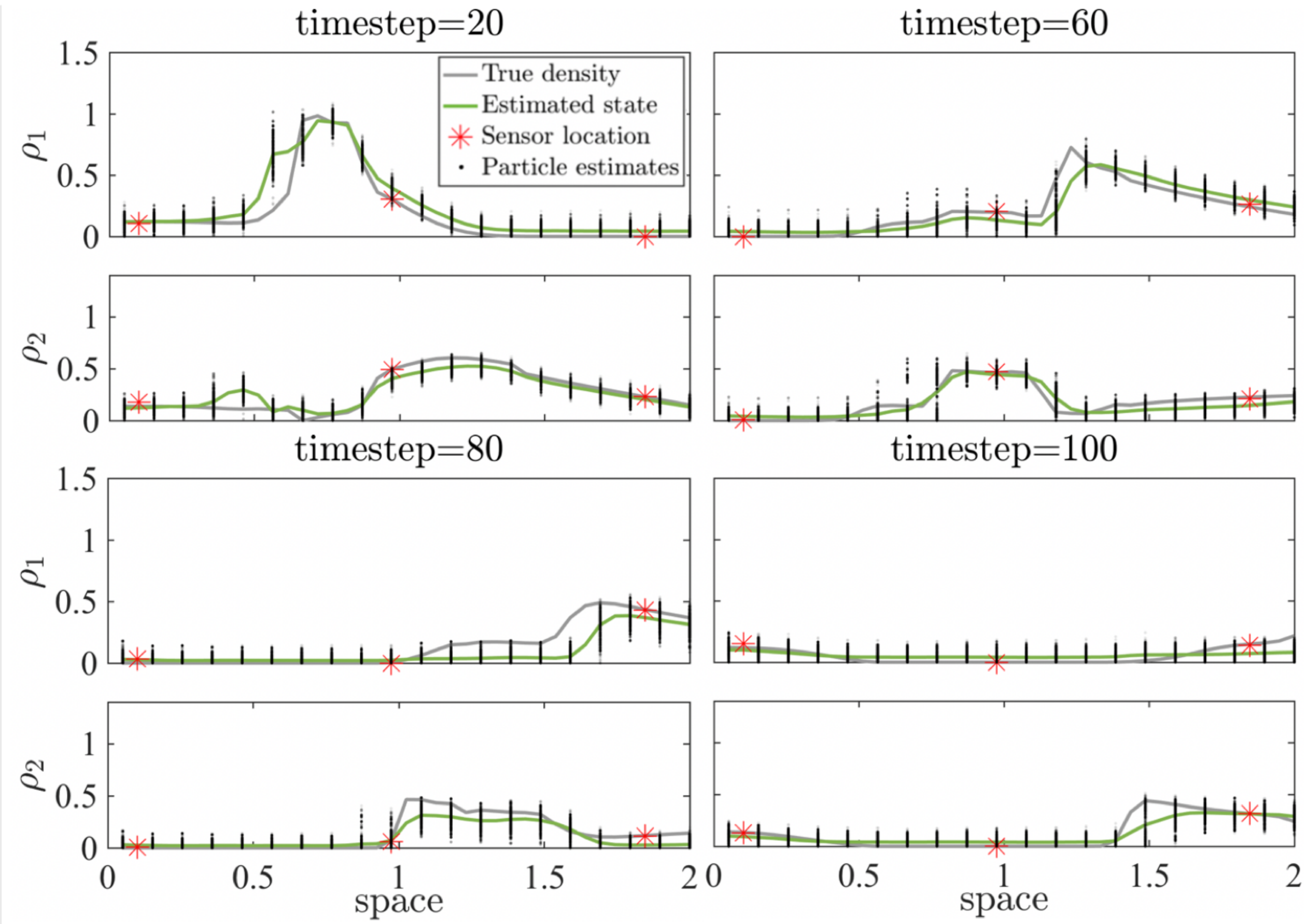

Fig. 5: Snapshots of the true state evolution and the particle filtering estimate in experiment 2: overtaking traffic.

which traces the true state evolution closely. Table IV summarizes the MAE of each vehicle class under the forward simulation and the particle filter, and indicates that the filter is able to reduce the MAE by 35-48%. This illustrates that the particle filter can be used on heterogeneous traffic in which overtaking occurs to capture more accurate traffic state estimates compared to pure forward simulation.

D. Sensitivity analysis

Finally, we examine the sensitivity of the estimator to the changes in the model parameters in both the creeping and overtaking experiments as described in this study. In each experiment, all parameters, initial conditions, and boundary conditions are fixed as described in the previous subsections except for

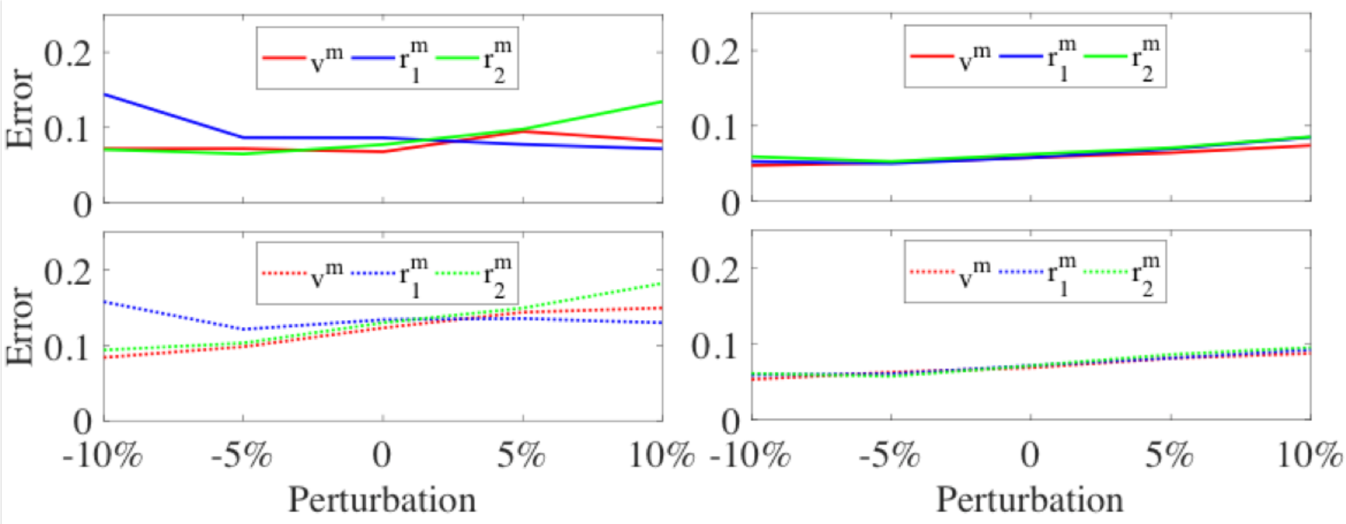

Fig. 6: MAE of small-vehicle (upper) and large-vehicle (lower) estimation with respect to the variation of model parameters for experiment 1: creeping traffic (left) and experiment 2: overtaking traffic (right).

a single parameter which is perturbed. Multiple runs of the particle filter are conducted to get an average performance result for each set of model parameters in each test. Each model parameter is perturbed by $-10\%, -5\%, 5\%$ and 10%, and the total runs of the particle filter is (3 model parameters)×(5 variations for each model parameter)×(3 particle filter runs per perturbation)×(2 experiments) = 90 runs. As shown in Figure 6, in the overtaking experiment the estimation performance quantified by the MAE is almost unaffected by the model parameters within the parameter perturbations tested. On the other hand, the MAE is observed to be more sensitive to the change of r_1^m in the creeping experiment on the lower end, and r_2^m on the higher end.

V. Conclusion

This article considers a particle filtering approach to estimate complex traffic in which overtaking or creeping can occur. Compared to the more common problems of homogeneous flow or heterogeneous flow with strict lane adherence, in this work the traffic state is governed by a model which does not require lane discipline. Using two numerical examples, it is demonstrated that the particle filter can reduce state estimation errors when compared to open loop simulations.

This article is a starting point for further work. For example, this work did not consider an exhaustive set of experiments under which the performance of the filter could be assessed, which should be considered in future work. Moving towards realistic deployment settings, the functional form of the velocity function and the calibration of the model parameters will also be important questions to consider. If data from field studies becomes available, it would be interesting to assess the filtering performance in practice in addition to the numerical experiments considered in this work.

VI. ACKNOWLEDGEMENT

This material is based upon work supported by the National Science Foundation under Grant No. CMMI-1853913 and the USDOT Dwight D. Eisenhower Fellowship program under Grant No. 693JJ31945012.

REFERENCES

[1] S. Fan and D. Work. A heterogeneous multiclass traffic flow model with creeping. *SIAM Journal on Applied Mathematics*, 75(2):813–835, 2015.

[2] 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(1178):317–345, 1955.

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

[4] G. C. K. Wong and S. C. Wong. A multi-class traffic flow model - an extension of LWR model with heterogeneous drivers. *Transportation Research Part A*, 36(9):827–841, 2002.

[5] S. Chanut and L. Leclercq. Introducing the effects of slow vehicles in a LWR two-flow traffic model. In Piet H. L. Bovy Michael Schreckenberg Serge P.Hoogendoorn, Stefan Luding and Dietrich E. Wolf, editors, *Traffic and Granular Flow '03*, pages 241–246, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg.

[6] D. Ngoduy and R. Liu. Multiclass first-order simulation model to explain non-linear traffic phenomena. *Physica A: Statistical Mechanics and its Applications*, 385(2):667 – 682, 2007.

[7] J.W.C. van Lint, S. P. Hoogendoorn, and M. Schreuder. Fastlane: New multiclass first-order traffic flow model. *Transportation Research Record: Journal of the Transportation Research Board*, 2088, 12 2008.

[8] H. van Lint T. Schreiter and S. Hoogendoorn. Multi-class ramp metering: Concepts and initial results. In *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 885–889, Oct 2011.

[9] M. L. Delle Monache C. Chalons and P. Goatin. A conservative scheme for non-classical solutions to a strongly coupled PDE-ODE problem. *Interfaces and Free Boundaries*, 19(4):553–570, 2017.

[10] S. Benzoni-Gavage and R. M. Colombo. An n-populations model for traffic flow. *European Journal of Applied Mathematics*, 14:587–612, 2002.

[11] H. S. Mahmassani R. Nair and E. Miller-Hooks. A porous flow approach to modeling heterogeneous traffic in disordered systems. *Procedia - Social and Behavioral Sciences*, 17:611 – 627, 2011. Papers selected for the 19th International Symposium on Transportation and Traffic Theory.

[12] P. Goatin S. Gashaw and J. Härri. Modeling and Analysis of Mixed Flow of Cars and Powered Two Wheelers. *Transportation research. Part C, Emerging technologies*, pages 1–44, 2018.

[13] G. Costeseque and A. Duret. Mesoscopic multiclass traffic flow modeling on multi-lane sections. In *95th annual meeting transportation research board - TRB*, page 27, Washington DC, United States, January 2016.

[14] C. Mallikarjuna and K. R. Rao. Cellular automata model for heterogeneous traffic. *Journal of Advanced Transportation*, 43(3):321–345, 2009.

[15] A. Ferrara, S. Sacone, and S. Siri. *First-Order Macroscopic Traffic Models*, pages 47–84. Springer International Publishing, Cham, 2018.

[16] M. Garavello, K. Han, and B. Piccoli. *Models for Vehicular Traffic on Networks*, volume 9 of *AIMS Series on Applied Mathematics*. American Institute of Mathematical Sciences, 2016.

[17] F. van Wageningen-Kessels, H. van Lint, K. Vuik, and S. Hoogendoorn. Genealogy of traffic flow models. *EURO Journal on Transportation and Logistics*, 4(4):445–473, Dec 2015.

[18] F. Van Wageningen-Kessels. Multi-class continuum traffic flow models: Analysis and simulation methods. *None*, 2013-03-20.

[19] M. W. Szeto and D. C. Gazis. Application of Kalman filtering to the surveillance and control of traffic systems. *Transportation Science*, 6(4):419–439, 1972.

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

[21] J. W. C. Van Lint, S. P. Hoogendoorn, and A. Hegyi. Dual EKF state and parameter estimation in multi-class first-order traffic flow models. *IFAC Proceedings Volumes*, 41(2):14078 – 14083, 2008. 17th IFAC World Congress.

[22] R. Babuska A. Hegyi, D. Girimonte and B. De Schutter. A comparison of filter configurations for freeway traffic state estimation. In *2006 IEEE Intelligent Transportation Systems Conference*, pages 1029–1034, Sep. 2006.

[23] D. Ngoduy and A. Sumalee. Adaptive estimation of noise covariance matrices in unscented Kalman filter for multiclass traffic flow model. *Transportation Research Record*, 2188(1):119–130, 2010.

[24] 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. In *2008 47th IEEE Conference on Decision and Control*, pages 5062–5068, Dec 2008.

[25] S. Blandin, A. Couque, A. Bayen, and D. Work. On sequential data assimilation for scalar macroscopic traffic flow models. *Physica D: Nonlinear Phenomena*, 241(17):1421 – 1440, 2012.

[26] Y. Sun and D. B. Work. Error bounds for Kalman filters on traffic networks. *Networks & Heterogeneous Media*, 13(2):261–295, 2018.

[27] A. Doucet and A. Johansen. A tutorial on particle filtering and smoothing: Fifteen years later. *Handbook of Nonlinear Filtering*, 12, 01 2009.

[28] Z. Chen. Bayesian filtering: From kalman filters to particle filters, and beyond. *Statistics*, 182, 01 2003.

[29] R. K. Boel L. Mihaylova and A. Hegyi. Freeway traffic estimation within particle filtering framework. *Automatica*, 43:290–300, 02 2007.

[30] L. Mihaylova, A. Hegyi, A. Gning, and R. K. Boel. Parallelized particle and gaussian sum particle filters for large-scale freeway traffic systems. *IEEE Transactions on Intelligent Transportation Systems*, 13(1):36–48, March 2012.

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

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

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

[34] D. Simon. *Optimal State Estimation*. John Wiley & Sons, Inc., 2006.

[35] J. Kaipio and E. Somersalo. *Statistical and computational inverse problems*, volume 160. Springer Science & Business Media, 2006.