

Highway Traffic State Estimation Using Improved Mixture Kalman Filters for Effective Ramp Metering Control

Xiaotian Sun, Laura Muñoz, and Roberto Horowitz
Department of Mechanical Engineering
University of California at Berkeley
Berkeley, CA 94720-1742
Emails: {sunx, lmunoz, horowitz}@me.berkeley.edu

Abstract—In this paper, we use a cell transmission model based switching state-space model to estimate vehicle densities and congestion modes at unmeasured locations on a highway section. The mixture Kalman filter algorithm, which is based on sequential Monte Carlo method, is employed to approximately solve the difficult problem of inference on a switching state-space model with an unobserved discrete state. We propose a scheme to prevent the risk of weight underflow and to introduce forgetting. The estimation results show that comparable accuracies can be achieved using either a small or a large number of sampling sequences, thus make it possible to carry out efficient online filtering. Underflow prevention and forgetting improves estimation accuracy in our examples. On average, a mean percentage error of approximately 10% is achieved for the vehicle density estimation. The estimation performance is consistent with data sets from various days.

I. INTRODUCTION

Congestion on urban highway networks in metropolitan areas occurs regularly and causes inefficient operation of highways, waste of resources, and increased air pollution. A widely used method to prevent and/or relieve highway congestion is to control the demand by means of on-ramp metering. Many ramp metering strategies, such as time-of-day (TOD) tables, static local occupancy response, ALINEA [1], etc., have been proposed, tested and deployed on highway networks at various locations.

In order to effectively control the on-ramp flows, traffic state information, such as vehicle density and the presence or absence of nearby congestion, has to be made available to the ramp metering controller. However, cost and other limitations prevent sensory devices being installed and maintained at all desired locations. Therefore, these traffic states must be estimated using the available data.

In this paper, we design and implement a traffic state estimator based on an improved version of the so-called mixture Kalman filter [2] and demonstrate the effectiveness and efficiency of the estimator. The paper is organized as follows: in Section II we introduce the cell transmission model [3], [4], and a switching mode model [5] that is based on it. We then briefly describe the switching state-space model and an approximate inference algorithm on

it, called mixture Kalman filtering [2] in Sections III and IV. Weight underflow prevention and a forgetting scheme to improve the performance of the mixture Kalman filter is proposed in Section V. In Sections VI and VII we detail the application of the mixture Kalman filter to traffic state estimation and analyze the estimation results. A summary and some directions for future work are given in Section VIII.

II. HIGHWAY TRAFFIC MODELS

Highway traffic is an interesting yet complex phenomenon to model. Many efforts have been made to establish and validate both microscopic, *e.g.*, car-following, and macroscopic, *e.g.*, hydrodynamics-based, models. However, many of these models are computationally too expensive to be used for online estimation of the traffic state in a large-scale highway network.

The cell transmission model (CTM) [3], [4] is an analytically simple model, yet it captures many important traffic phenomena, such as queue build-up and dissipation, backward propagation of congestion waves, etc. The CTM is a discrete model, both in space and time. It divides the highway into small segments, which are called “cells”, as shown in Fig. 1. The traffic flow q_i going into a cell i is considered constant between two consecutive times t and $t+1$ and is determined by the following relationship:

$$q_i(t) = \min\{v_{f,i-1}\rho_{i-1}(t), w_{c,i}(\rho_{J,i} - \rho_i(t)), Q_{M,i}\}, \quad (1)$$

where for a cell i , $\rho_i(t)$ is the average vehicle density between times t and $t+1$, $\rho_{J,i}$ is the jam density, *i.e.*, the maximum vehicle density allowed in cell i , $v_{f,i}$ is the free-flow speed, $w_{c,i}$ the backward congestion wave propagation speed, and $Q_{M,i}$ is the flow capacity, *i.e.*, the maximum possible flow. This relationship is derived from a simplified flow-density relation, which in traffic engineering is often called a *fundamental diagram* in, as shown in Fig. 2.

The three terms involved in the minimization in (1) can be interpreted as follows. The first term $v_{f,i-1}\rho_{i-1}(t)$ is the flow that can be supplied by the cell $i-1$. The second term is the flow that can be absorbed by the cell i . And the third term $Q_{M,i}$ is the maximum possible flow from cell $i-1$ to i .

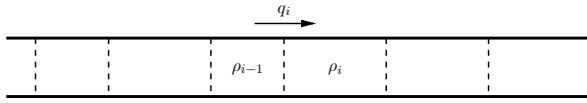


Fig. 1. A highway section divided into cells.

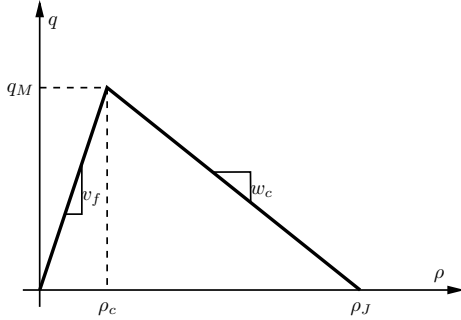


Fig. 2. A simplified fundamental diagram (flow-density relation) used by the cell transmission model.

Although the cell transmission model is much simpler than many other higher order hydrodynamics-based partial differential models, the nonlinear nature of the flow-density relation still makes it difficult to analyze and to use as a basis for the design of traffic controllers Muñoz *et al.* [5] piecewise linearized the cell transmission model and derived a CTM-based switching-mode model, which depends on the traffic congestion status in each subsection of the highway. This switching mode model includes several modes, *i.e.*, discrete states. In each mode, the vehicle densities in the cells evolve according to a different set of linear difference equations. Among these modes, two are of greatest importance: pure free-flow and full congestion. In this paper, we further simplify this model by considering only these two modes and neglecting all other mixed cases, such as the mode wherein half the cells in a section are in free-flow and half are in congestion. This is partially justified by the fact that the highway sections are short and the mixed cases are often transient. We write down this simplified switching state-space model with two discrete states: free-flow and congestion, as follows.

When the entire highway section is in free-flow mode, the first term in (1) dominates, and the difference equations¹ are

$$\begin{bmatrix} \rho_1 \\ \rho_2 \\ \rho_3 \\ \rho_4 \end{bmatrix} (t+1) = \begin{bmatrix} 1 - \frac{v_{f1}T_s}{l_1} & 0 & 0 & 0 \\ \frac{v_{f1}T_s}{l_2} & 1 - \frac{v_{f2}T_s}{l_2} & 0 & 0 \\ 0 & \frac{v_{f2}T_s}{l_3} & 1 - \frac{v_{f3}T_s}{l_3} & 0 \\ 0 & 0 & \frac{v_{f3}T_s}{l_4} & 1 - \frac{v_{f4}T_s}{l_4} \end{bmatrix} \begin{bmatrix} \rho_1 \\ \rho_2 \\ \rho_3 \\ \rho_4 \end{bmatrix} (t) + \begin{bmatrix} \frac{T_s}{l_1} & 0 & 0 & 0 \\ 0 & 0 & \frac{T_s}{l_2} & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\frac{T_s}{l_4} \end{bmatrix} \begin{bmatrix} q_{m1} \\ q_{m2} \\ r \\ f \end{bmatrix} (t) \quad (2)$$

$$= A(1)\rho(t) + B_q(1)q(t), \quad (3)$$

where q_{m1} and q_{m2} are the mainline entering and exiting flows, respectively, r and f are the on-ramp and off-ramp

¹For simplicity of notation, we use a section of 4 cells as an example.

flows, respectively, l_i is the length of cell i , and T_s is the sampling time.

When the entire highway section is in congestion mode, the second term in (1) dominates, and the difference equations are

$$\begin{bmatrix} \rho_1 \\ \rho_2 \\ \rho_3 \\ \rho_4 \end{bmatrix} (t+1) = \begin{bmatrix} 1 - \frac{w_{c1}T_s}{l_1} & \frac{w_{c2}T_s}{l_1} & 0 & 0 \\ 0 & 1 - \frac{w_{c2}T_s}{l_2} & \frac{w_{c3}T_s}{l_2} & 0 \\ 0 & 0 & 1 - \frac{w_{c3}T_s}{l_3} & \frac{w_{c4}T_s}{l_3} \\ 0 & 0 & 0 & 1 - \frac{w_{c4}T_s}{l_4} \end{bmatrix} \begin{bmatrix} \rho_1 \\ \rho_2 \\ \rho_3 \\ \rho_4 \end{bmatrix} (t) + \begin{bmatrix} 0 & 0 & \frac{T_s}{l_1} & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\frac{T_s}{l_3} \\ 0 & -\frac{T_s}{l_4} & 0 & 0 \end{bmatrix} \begin{bmatrix} q_{m1} \\ q_{m2} \\ r \\ f \end{bmatrix} (t) + \begin{bmatrix} \frac{w_{c1}T_s}{l_1} & -\frac{w_{c2}T_s}{l_1} & 0 & 0 \\ 0 & \frac{w_{c2}T_s}{l_2} & -\frac{w_{c3}T_s}{l_2} & 0 \\ 0 & 0 & \frac{w_{c3}T_s}{l_3} & -\frac{w_{c4}T_s}{l_3} \\ 0 & 0 & 0 & \frac{w_{c4}T_s}{l_4} \end{bmatrix} \begin{bmatrix} \rho_{J1} \\ \rho_{J2} \\ \rho_{J3} \\ \rho_{J4} \end{bmatrix}, \quad (4)$$

$$= A(2)\rho(t) + B_q(2)q(t) + B_J(2)\rho_J, \quad (5)$$

where ρ_{Ji} is the jam density (maximum allowable density) in cell i .

III. SWITCHING STATE-SPACE MODELS

A switching state-space model can be thought of as a combination of two popular statistical models: the hidden Markov model (HMM) and the linear state-space model (SSM). This combined class of models is used in many applications, such as modeling discrete event systems, fault detection in dynamic systems, and piecewise linearization of nonlinear systems, to name a few. They are often referred to by different names in different fields, such as stochastic hybrid systems [6] in control, hybrid dynamic Bayesian networks [7], and jump Markov linear systems [8]. Within this paper, we shall refer to this model as the *switching state-space model*. In Fig. 3, we represent this model graphically using the popular Bayesian network representation. In the figure, the arrows represent conditional dependence. The square nodes are discrete and the round ones are continuous states, and shaded nodes are observed.

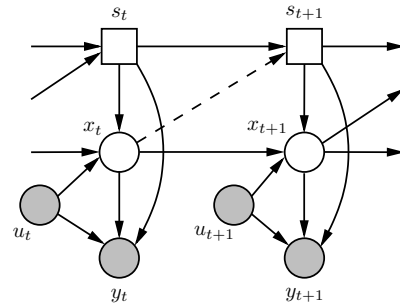


Fig. 3. The graphic representation of a switching state-space model.

We define the switching state-space model as follows.

$$x_{t+1} = A(s_{t+1})x_t + B(s_{t+1})u_{t+1} + F(s_{t+1})v_{t+1}, \quad (6)$$

$$y_t = C(s_t)x_t + D(s_t)u_t + G(s_t)w_t, \quad (7)$$

where $t = 0, 1, 2, \dots$ is the discrete time, $s_t \in S$ is the unobserved discrete, finite-state random variable, $x_t \in \mathbb{R}^{n_x}$ is the unobserved continuous random variable, $y_t \in \mathbb{R}^{n_y}$ is the continuous observation, $u_t \in \mathbb{R}^{n_u}$ is the known exogenous input, $v_t \sim \mathcal{N}(0, I_{n_v})$ and $w_t \sim \mathcal{N}(0, I_{n_w})$ are the unobserved Gaussian white noise, and $A(s_t)$, $B(s_t)$, $C(s_t)$, $D(s_t)$, $F(s_t)$, and $G(s_t)$ are the time-varying real matrices, with proper dimensions, that are functions of the finite-state variable s_t .²

Unlike HMM and SSM, there is no known exact inference algorithm for the switching state-space model, due to the exponentially increasing sample space S^t as time grows. However, many approximate inference algorithms have been proposed. In this paper, we use a sequential Monte Carlo approach, called mixture Kalman filtering [2], which we describe briefly in the next section for completeness.

IV. THE MIXTURE KALMAN FILTER ALGORITHM

For the switching state-space model as defined in (6) and (7), if the discrete state sequence were given from time 0 to t , then the continuous state could be easily estimated by a (non-stationary) Kalman filter, since the sequences of the system matrices A , B , C , D , F , and G would be known. Therefore, conditioned on s_t ,³ the following distributions are readily available through a Kalman filter: $p(x_t | y_t, \mathbf{u}_t, s_t)$, $p(x_{t+1} | y_t, \mathbf{u}_{t+1}, s_{t+1})$, and $p(y_{t+1} | y_t, \mathbf{u}_{t+1}, s_{t+1})$.

Theoretically, the distribution of x_t can be obtained by marginalizing the joint distribution of x_t and s_t as follows:

$$p(x_t | y_t, \mathbf{u}_t) = \int_{S^t} p(x_t | y_t, \mathbf{u}_t, s_t) p(s_t | y_t, \mathbf{u}_t) ds_t. \quad (8)$$

However, as we mentioned earlier, the prohibitive difficulty of inference on a switching state-space model is that the space S^t grows exponentially as t increases, which makes the integral in (8) computationally impossible.

The mixture Kalman filter algorithm [2] utilizes a sequential Monte Carlo method in which a fixed finite number, M , of sequences, $s_t^{(m)}$, $m = 1, 2, \dots, M$, of the discrete state are sampled carefully from the space S^t according to some sort of predictive probability, and the integral in (8) is approximated by a finite sum, *i.e.*,

$$p(x_t | y_t, \mathbf{u}_t) \approx \frac{\sum_{m=1}^M p(x_t | y_t, \mathbf{u}_t, s_t^{(m)}) p(s_t^{(m)} | y_t, \mathbf{u}_t)}{\sum_{m=1}^M p(s_t^{(m)} | y_t, \mathbf{u}_t)} \quad (9)$$

$$= \sum_{m=1}^M \xi_t^{(m)} p(x_t | y_t, \mathbf{u}_t, s_t^{(m)}), \quad (10)$$

where

$$\xi_t^{(m)} := \frac{p(s_t^{(m)} | y_t, \mathbf{u}_t)}{\sum_{m=1}^M p(s_t^{(m)} | y_t, \mathbf{u}_t)}, \quad (11)$$

are normalized time-varying weights that represent the conditional probability.

²In the following discussion, we will not distinguish between a random variable and its realization.

³For simplicity of notation, we make the following convention: a normal font (medium weighted) variable, *e.g.*, s_t , denotes a single sample of that variable at time t , while a bold variable, *e.g.*, \mathbf{s}_t , denotes a sequence of samples from time 0 to t , *i.e.*, $\mathbf{s}_t = \{s_0, s_1, \dots, s_t\}$.

Once a new observation y_{t+1} is available, the weights $\xi_{t+1}^{(m)}$ are updated in such a way that favors the sample sequences with larger likelihoods $p(y_{t+1} | y_t, \mathbf{u}_t, s_t^{(m)})$. For this purpose, incremental weights are defined by

$$\zeta_{t+1}^{(m)} := p(y_{t+1} | y_t, \mathbf{u}_t, s_t^{(m)}), \quad (12)$$

and the weights are updated as follows

$$\xi_{t+1}^{(m)} = \frac{\zeta_t^{(m)} \zeta_{t+1}^{(m)}}{\sum_{m=1}^M \zeta_t^{(m)} \zeta_{t+1}^{(m)}}. \quad (13)$$

The predictive sampling probability $\mu_{t+1}^{(m)}(s)$ is defined as

$$\begin{aligned} \mu_{t+1}^{(m)}(s) &\propto p(s_{t+1}^{(m)} = s, y_{t+1} | s_t^{(m)}, y_t, \mathbf{u}_{t+1}) \\ &= p(y_{t+1} | s_{t+1}^{(m)} = s, s_t^{(m)}, y_t, \mathbf{u}_{t+1}) p(s_{t+1}^{(m)} = s | s_t^{(m)}, y_t, \mathbf{u}_{t+1}). \end{aligned} \quad (14)$$

In (14),

$$p(y_{t+1} | s_{t+1}^{(m)} = s, s_t^{(m)}, y_t, \mathbf{u}_{t+1}) = \mathcal{N}(\hat{y}_{t+1|t}^{(m)}, Q_{t+1|t}^{(m)}), \quad (15)$$

where $\hat{y}_{t+1|t}^{(m)}$ is the *a priori* expectation and $Q_{t+1|t}^{(m)}$ is the *a priori* covariance of y_{t+1} , which can be obtained from the conditional Kalman filters given $s_{t+1}^{(m)} = s$, $s_t^{(m)}$, y_t , and \mathbf{u}_{t+1} .

With the definition of $\mu_{t+1}^{(m)}(s)$, the incremental weights

$$\begin{aligned} \zeta_{t+1}^{(m)} &= p(y_{t+1} | y_t, \mathbf{u}_t, s_t^{(m)}) \\ &= \sum_{s \in S} p(y_{t+1} | s_{t+1}^{(m)} = s, s_t^{(m)}, y_t, \mathbf{u}_t) p(s_{t+1}^{(m)} = s | s_t^{(m)}, y_t, \mathbf{u}_t) \end{aligned} \quad (16)$$

$$= \sum_{s \in S} \mu_{t+1}^{(m)}(s). \quad (17)$$

For any measurable function $h(x_t)$, its conditional expectation

$$\mathbb{E}[h(x_t) | y_t, \mathbf{u}_t] = \int_{\mathbb{R}^{n_x}} h(x_t) p(x_t | y_t, \mathbf{u}_t) dx_t \quad (18)$$

is approximated by

$$\hat{\mathbb{E}}[h(x_t) | y_t, \mathbf{u}_t] = \sum_{m=1}^M \xi_t^{(m)} \int_{\mathbb{R}^{n_x}} h(x_t) p(x_t | y_t, \mathbf{u}_t, s_t^{(m)}) dx_t. \quad (19)$$

As a special case, the continuous state estimate is obtained by

$$\hat{x}_{t|t} = \sum_{m=1}^M \xi_t^{(m)} \hat{x}_{t|t}^{(m)}, \quad (20)$$

where $\hat{x}_{t|t}^{(m)}$ is the *a posteriori* state estimation from a conditional Kalman filter given a sample sequence $s_t^{(m)}$. This estimation is referred to as *mixture estimation* because it is a weighted sum of the M conditional Kalman estimations.

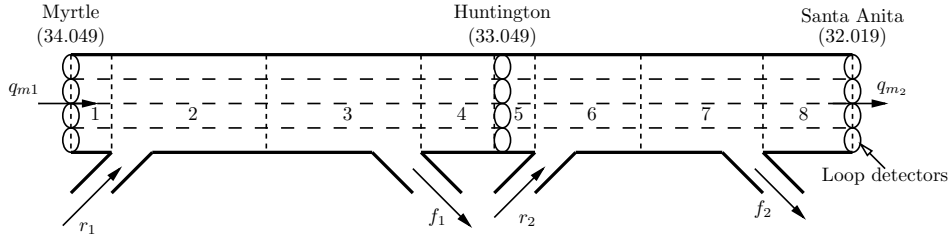


Fig. 4. Cell configuration for the section of Interstate 210 West in Pasadena, CA, from Myrtle to Santa Anita.

V. WEIGHT UNDERFLOW PREVENTION AND FORGETTING

In the implementation of the mixture Kalman filter algorithm, we find that after a certain period of time, most of the weights $\xi_t^{(m)}$ approach 0, while only a few are of modest magnitudes. This phenomenon is a direct result of the weight update scheme (13)–(17) and may have several undesirable effects:

- 1) The contributions of those sample sequences with tiny weights to the mixture estimate are negligible. Therefore, the effective number of sample sequences is reduced and computation time and memory are wasted to maintain these sample sequences.
- 2) There exists an underflow risk for these tiny weights due to limited floating point precision. If underflow does happen to some of the weights, their corresponding sample sequences become permanently inactive.
- 3) The influence of the early history persists. It is difficult for the tiny weights to grow back to within a modest range of magnitudes, even if the current likelihoods (12) favor these sample sequences.

We propose to introduce a lower bound to the weights $\xi_t^{(m)}$ to address these potential problems. We choose a small positive number $\varepsilon \ll 1$ and set the weight lower bound

$$\underline{\xi} = \frac{\varepsilon}{M}, \quad (21)$$

where M is the number of the sample sequences. After updating the weights according to (13)–(17), we lower bound the weights

$$\xi_{t+1}^{(m)} = \max\{\xi_{t+1}^{(m)}, \underline{\xi}\}, \quad (22)$$

for all m and re-normalize them.

This simple procedure prevents the underflow phenomenon. It also acts as a forgetting factor for the weights update in the sense that it stops the influence of the early history once the effect of the history reduces the weight to the minimum level. It makes the weights recover more quickly once their corresponding sample sequences are favored by the current measurements. On the other hand, it introduces only minimal estimation error because its total effect on the weights is bounded by the small number ε .

VI. APPLICATION TO HIGHWAY TRAFFIC ESTIMATION

Vehicle density and traffic flow data are collected on California highways by loop detectors buried in the pavement. Our object in this study is a section of Interstate 210 West in Pasadena, CA, from Myrtle to Santa Anita, with postmiles

from 34.049 to 32.019, as shown in Fig. 4. This section is divided into 8 cells. The loop detectors are located at the two boundaries and in the middle, as shown in the figure. Therefore, we consider the two boundary flows q_{m1} and q_{m2} , and the densities in cells 1, 5, and 8 are measured. In addition, the ramp flows r_1 , r_2 , f_1 , and f_2 are also measured. However we do not make the measured density ρ_5 available to the model. This provides us a way to compare the estimated and measured densities and enable us to evaluate the performance of the mixture Kalman filter.

With the above configuration, we have $A(1), A(2) \in \mathbb{R}^{8 \times 8}$, with similar structure as the “ A ” matrices in (2) and (4),

$$C(1) = C(2) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad (23)$$

and

$$D(1) = D(2) = 0. \quad (24)$$

We also let $B = [B_q \ B_J]$ and set $B_J(1) = 0$. In addition, we define

$$F(s) = \sigma_v(s)I_{n_x}, \quad (25)$$

and

$$G(s) = \sigma_w(s)I_{n_y}, \quad (26)$$

where $\sigma_v(s)$ and $\sigma_w(s)$ are the standard deviations of the white noise v and w in the two modes, respectively, and $s \in S = \{1, 2\}$ (1 is the free-flow mode and 2 the congestion mode).

In order to sample the sequences and update the weights, the predictive sampling probabilities as defined in (14) have to be computed. We simplify this expression as follows.

$$\begin{aligned} & p(s_{t+1}^{(m)} = s \mid s_t^{(m)}, \mathbf{y}_t, \mathbf{u}_{t+1}) \\ &= \int_{\mathbb{R}^{n_x \times n_t}} p(s_{t+1}^{(m)} = s \mid \mathbf{x}_t, s_t^{(m)}, \mathbf{y}_t, \mathbf{u}_{t+1}) p(\mathbf{x}_t \mid s_t^{(m)}, \mathbf{y}_t, \mathbf{u}_{t+1}) d\mathbf{x}_t \end{aligned} \quad (27)$$

$$= \int_{\mathbb{R}^{n_x \times n_t}} p(s_{t+1}^{(m)} = s \mid x_t, s_t^{(m)}) p(x_t, \mathbf{x}_{t-1} \mid s_t^{(m)}, \mathbf{y}_t, \mathbf{u}_t) d\mathbf{x}_t \quad (28)$$

$$= \int_{\mathbb{R}^{n_x}} p(s_{t+1}^{(m)} = s \mid x_t, s_t^{(m)}) p(x_t \mid s_t^{(m)}, \mathbf{y}_t, \mathbf{u}_t) dx_t, \quad (29)$$

where in the step from (27) to (28) we utilize the conditional independence fact that given s_t and \mathbf{x}_t , s_{t+1} is independent of \mathbf{y}_t and \mathbf{u}_{t+1} , and the Markov assumption we made in the graphical model in Fig. 3, *i.e.*,

$$p(s_{t+1} \mid s_t, \mathbf{x}_t) = p(s_{t+1} \mid s_t, x_t). \quad (30)$$

In this step, we also use the fact that \mathbf{x}_t and \mathbf{u}_{t+1} are conditionally independent. The step from (28) to (29) is marginalization over \mathbf{x}_{t-1} .

The two factors in (29) can be obtained as follows. From the conditional Kalman filters given $s_t^{(m)}$, \mathbf{y}_t , and \mathbf{u}_t ,

$$p(\mathbf{x}_t | s_t^{(m)}, \mathbf{y}_t, \mathbf{u}_t) = \mathcal{N}(\hat{x}_{t|t}^{(m)}, P_{t|t}^{(m)}), \quad (31)$$

where $\hat{x}_{t|t}^{(m)}$ is the *a posteriori* expectation and $P_{t|t}^{(m)}$ is the *a posteriori* covariance of the continuous state x_t . And $p(s_{t+1}^{(m)} = s | x_t, s_t^{(m)})$ is the *a priori* transition probability.

For simplicity, we neglect the conditional dependence between s_{t+1} and x_t , *i.e.*, the dashed arrow shown in Fig. 3. Therefore, the discrete state transition probability in (30) is simply given by a table

$$p(s_{t+1} | s_t, x_t) = p(s_{t+1} | s_t) = \begin{array}{c|cc} & s_{t+1} = 1 & s_{t+1} = 2 \\ \hline s_t = 1 & 0.95 & 0.05 \\ s_t = 2 & 0.05 & 0.95 \end{array}. \quad (32)$$

This transition probability makes the discrete state jump less often. It corresponds to a mean transition time of $380T_s$, which is roughly 31 minutes.

We are also able to obtain an approximate maximum *a posteriori* (MAP) estimation of the discrete state s_t using the following approximation of the *a posteriori* distribution of s_t :

$$p(s_t | \mathbf{y}_t, \mathbf{u}_t) = E[1_s(s_t) | \mathbf{y}_t, \mathbf{u}_t] \quad (33)$$

$$= \int_{S_t} 1_s(s_t) p(s_t | \mathbf{y}_t, \mathbf{u}_t) ds_t \quad (34)$$

$$\approx \sum_{m=1}^M \xi_t^{(m)} 1_s(s_t^{(m)}), \quad (35)$$

where

$$1_s(s_t) := \begin{cases} 1 & \text{if } s_t = s, \\ 0 & \text{otherwise,} \end{cases} \quad (36)$$

is the indicator function of s_t . The sequence of the MAP estimates for the discrete state is

$$\hat{s}_{t,\text{MAP}} = \arg \max_s p(s_t = s | \mathbf{y}_t, \mathbf{u}_t). \quad (37)$$

Once we have the MAP discrete state estimation, we are able to estimate the continuous state by a non-stationary Kalman filter conditioned on $\hat{s}_{t,\text{MAP}}$. We denote this estimate by $\hat{x}_{t|t}^{\hat{s}_{t,\text{MAP}}}$ and informally call it the “MAP” estimation of the continuous state.

VII. RESULTS

The traffic flow and vehicle density data were obtained through the PeMS database [9] for four different days. In our implementation of the mixture Kalman filter, different values of M are used in order to experiment with the effect of the number of sample sequence. We also experiment with the effect of the weight underflow prevention and forgetting scheme that we proposed. Figures 5–7 show a few examples of the estimation results. In the plots, the estimated vehicle densities, as well as the measured ones, and

the MAP estimation of the congestion mode are shown. The detailed information, such as the date of the data, estimation parameters, etc., can be found in the captions of the figures. In the congestion mode subplot (s_{MAP}), 1 represents free-flow and 2 congestion.

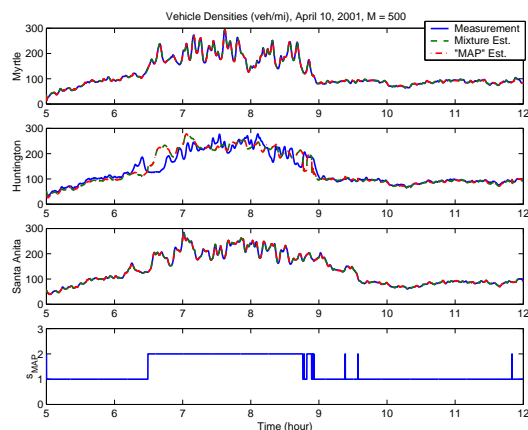


Fig. 5. Measured and estimated densities, with estimated traffic congestion mode. April 10, 2001. $M = 500$. Underflow prevention and forgetting not employed.

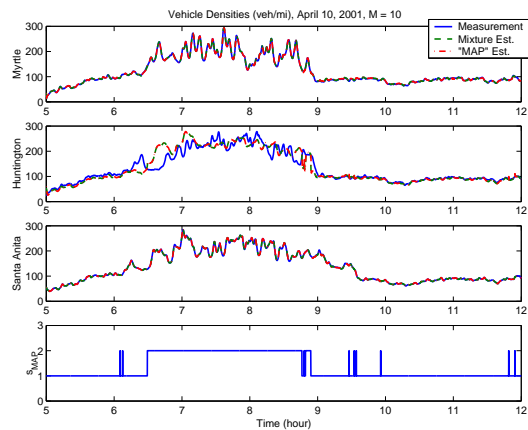


Fig. 6. Measured and estimated densities, with estimated traffic congestion mode. April 10, 2001. $M = 10$. Underflow prevention and forgetting not employed.

In order to quantitatively compare the performance of different estimates, such as those with different values of M , the mixture estimates and the “MAP” estimates, and the estimates with or without weight underflow prevention and forgetting, we define a mean percentage error measure

$$E_{\text{MPE}} = \frac{1}{T+1} \sum_{t=0}^T \left| \frac{\hat{x}_{t|t} - x_t}{x_t} \right|. \quad (38)$$

In particular, here we only consider the density and its estimation in cell 5, *i.e.*, ρ_5 and $\hat{\rho}_5$, because we have the measurement. The error measures, averaged from the results of several runs, for different estimates are listed in Table I.

It can be seen from the table that the algorithm achieves satisfactory results. On average, the mean percentage errors are about 10%. The performance is consistent with data sets from various days. With a small number of sample

TABLE I

THE MEAN PERCENTAGE ERROR MEASURES FOR THE VARIOUS ESTIMATES OF ρ_5 ON DIFFERENT DAYS.

Forgetting	E_{MPE}	March 15, 2001		March 27, 2001		April 10, 2001		April 25, 2001	
		$\hat{x}_{t t}$	$\hat{x}_{t t}^{S_t,MAP}$	$\hat{x}_{t t}$	$\hat{x}_{t t}^{S_t,MAP}$	$\hat{x}_{t t}$	$\hat{x}_{t t}^{S_t,MAP}$	$\hat{x}_{t t}$	$\hat{x}_{t t}^{S_t,MAP}$
No	$M = 10$	0.10661	0.10691	0.11012	0.11059	0.09549	0.09590	0.10575	0.10612
	$M = 100$	0.10639	0.10659	0.10987	0.11061	0.09492	0.09542	0.10565	0.10592
	$M = 500$	0.10629	0.10652	0.10981	0.11033	0.09479	0.09506	0.10575	0.10594
Yes	$M = 10$	0.10634	0.10660	0.10978	0.11036	0.09477	0.09517	0.10554	0.10584
	$M = 100$	0.10625	0.10640	0.10976	0.11062	0.09453	0.09491	0.10563	0.10590
	$M = 500$	0.10611	0.10626	0.10972	0.11045	0.09464	0.09466	0.10565	0.10573

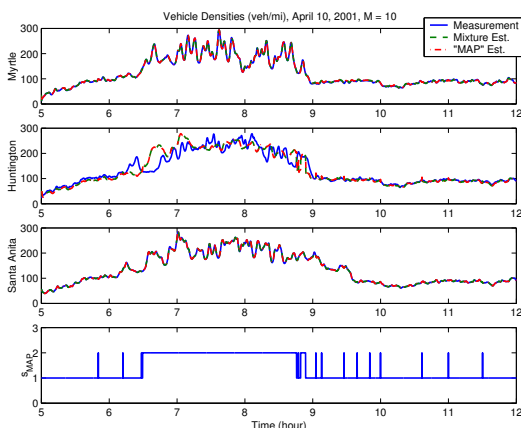


Fig. 7. Measured and estimated densities, with estimated traffic congestion mode. April 10, 2001. $M = 10$. Underflow prevention and forgetting employed.

sequences, e.g., $M = 10$, the algorithm achieves comparable estimation errors as with $M = 500$. This observation enables us to carry out efficient online filtering by using only a small number of sample sequences. It can also be seen from the table that the proposed weight underflow prevention and forgetting scheme improves the estimation accuracy slightly.

Another interesting observation is that the mixture estimation $\hat{x}_{t|t}$ performs slightly better than the MAP discrete sequence conditioned Kalman estimation $\hat{x}_{t|t}^{S_t,MAP}$ in general. This is because we only consider two modes in the switching state-space model and neglect other possible cases, such as the one that half of the section is in free-flow and half in congestion. The mixture estimation might account for these mixed cases better. However, since these mixed cases are transient and only last a short period of time, their contributions to the mean error are minimal.

VIII. SUMMARY AND FUTURE WORK

In this paper, we have applied an approximate inference algorithm called a mixture Kalman filter to a cell transmission based switching state-space model to estimate vehicle densities and congestion modes at unmeasured locations on a highway section. The sequential Monte Carlo based algorithm is effective and efficient. On average, a mean percentage error of $\sim 10\%$ is achieved, and is consistent over different days. The algorithm achieves comparable performance with a significantly smaller number of sample sequences. The proposed weight underflow prevention and

forgetting scheme improves estimation accuracy.

We are currently working on designing a ramp metering algorithm to relieve highway congestion and to improve capacity utilization. Analogous to the mixture Kalman filter, we are looking into a so-called *mixture controller*. Given a sample discrete state sequence $s_t^{(m)}$, a stabilizing controller can be designed under the framework of Markovian jump linear systems (MJLS) [10]. The mixture control command will be the weighted sum of the sample sequence conditioned control commands. Findings and results will be reported in future publications.

ACKNOWLEDGMENTS

This work was supported in part by the California PATH (Partners for Advanced Transit and Highways) under Task Order 4136.

REFERENCES

- [1] M. Papageorgiou, H. Hadj-Salem, and J.-M. Blosseville, "ALINEA: A local feedback control law for on-ramp metering," *Transportation Research Record*, no. 1320, pp. 58–64, 1991.
- [2] R. Chen and J. S. Liu, "Mixture Kalman filters," *Journal of the Royal Statistical Society, Series B—Statistical Methodology*, vol. 62, pp. 493–508, 2000.
- [3] C. F. Daganzo, "The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory," *Transportation Research Part B: Methodological*, vol. 28, no. 4, pp. 269–287, Aug. 1994.
- [4] —, "The cell transmission model, part II: Network traffic," *Transportation Research Part B: Methodological*, vol. 29, no. 2, pp. 79–93, Apr. 1995.
- [5] L. Muñoz, X. Sun, R. Horowitz, and L. Alvarez, "Traffic density estimation with the cell transmission model," in *Proceedings of the 2003 American Control Conference*, Denver, CO, USA, June 2003, pp. 3750–3755.
- [6] J. Hu, J. Lygeros, and S. Shankar, "Towards a theory of stochastic hybrid systems," in *Proceedings of the Third International Workshop on Hybrid Systems: Computation and Control*, ser. Springer Lecture Notes in Computer Science, N. Lynch and B. H. Krogh, Eds., vol. 1790, Pittsburgh, PA, USA, Mar. 2000, pp. 160–173.
- [7] M. I. Jordan, *An Introduction to Probabilistic Graphical Models*, 2002, in preparation.
- [8] A. Logothetis and V. Krishnamurthy, "Expectation maximization algorithms for MAP estimation of jump Markov linear systems," *IEEE Transactions on Signal Processing*, vol. 47, no. 8, pp. 2139–2156, Aug. 1999.
- [9] C. Chen, K. Petty, A. Skabardonis, P. Varaiya, and Z. Jia, "Freeway performance measurement system: Mining loop detector data," *Transportation Research Record*, no. 1748, pp. 96–102, 2001.
- [10] Y. Ji, H. J. Chizeck, X. Feng, and L. K. A., "Stability and control of discrete-time jump linear systems," *Control Theory and Advanced Technology*, vol. 7, no. 2, pp. 247–270, June 1991.