Bayesian Inference for Time Delay Systems with Application to Connected Automated Vehicles

Xunbi A. Ji, Tamás G. Molnár, Alex A. Gorodetsky, and Gábor Orosz

***Abstract*— In this paper, a Bayesian inference problem is set up to infer the time delay and the resistance models from the dynamics of a connected automated vehicle. The delayed rejection adaptive Metropolis Markov chain Monte Carlo method is applied to obtain the posterior distributions of time delay and resistance parameters simultaneously using experimental data. The estimations of the time delay are shown to be consistent among multiple datasets and different resistance models. The distribution of the posterior indicates that there exist multiple modes in time delay, corresponding to different behaviors in acceleration and deceleration.**

I. INTRODUCTION

Time delay and resistance forces are essential for modeling the dynamics of vehicles with high fidelity. Time delays usually show up as the human reaction time, engine actuation time and communication delays in connected vehicle systems [1]. In [2] different car-following models were evaluated using real-world data and simulations and it was shown that including time delays improves the fidelity of these models. On the other hand, resistance forces play an important role in vehicle performance and energy consumption [3]. We are interested in identifying the overall time delay and the coefficients of resistance forces for the longitudinal dynamics of a connected automated vehicle (CAV) based on real experimental data.

Bayesian inference has shown its power in parameter estimation in the field of car-following dynamics of human-driven vehicles: it can be used in both model selection [4] and calibration [5]. For example, hierarchical Bayesian estimation was applied in [6] for calibrating a car-following model with time-varying parameters related to vertical gradient. In [7] a time-varying car-following model was established via particle filtering while in [8] particle filtering was used for online parameter estimation. Inferring the time delay was not considered in the above-mentioned works, but it was introduced into the model as a given parameter in [5] and driver reaction time was identified through cross-correlation calculation in [9]. In [10], sensor delay for an automated vehicle was identified as a discrete parameter via trials of recursive least squares method. In this work, we consider the time delay as a continuous parameter in the longitudinal dynamics of a CAV and we infer the delay together with resistance coefficients using Bayesian inference.

Applying Bayesian inference for time delay systems is not a common practice, since the time delay is a special parameter that changes the relationship between the states. In [11] the authors gave an example of learning a delay differential equation with Gaussian process based on Markov Chain Monte Carlo sampling method. Sequential time delay updating with particle filter was used for multi-path global navigation satellite systems in [12]. A special distributed time delay was considered in [13] in the field of biochemical processes. In our application, the time delay determines which past state affects the state derivative of the CAV, and we compute the corresponding likelihood of the delay from data. To handle this, we introduce a general Bayesian network for time delay systems, and we identify different modes in the delay parameter through sampling methods. We also show the consistency of time delay among different resistance models.

The paper is constructed as follows. We introduce the longitudinal dynamics and the longitudinal controller of the connected automated vehicle with different resistance terms and the data from real experiments in Section II. In Section III, we set up the Bayesian inference problem for time delay systems in two alternative ways based on different noise considerations, and we implement the Delayed Rejection Adaptive Metropolis (DRAM) method for time delay systems with process noise. In Section IV, we analyze the parameter distributions given by DRAM for different data sets and resistance models. In Section V, we summarize our results and give directions for future work.

II. PROBLEM SETTING

We consider the scenario shown in Fig. 1(a) where a connected automated vehicle (CAV) follows a connected human-driven vehicle (CHV). The CAV measures its own position and velocity, while it is also aware of the position and velocity of the leading CHV via vehicle-to-vehicle (V2V) communication. These data are used in the longitudinal controller of the CAV. This scenario was realized in the experiments detailed in [14], [15].

For the CAV, the longitudinal dynamics are modeled by the following delayed double integrator model with saturation:

$$\begin{aligned} \dot{h}(t) &= v_{\mathrm{L}}(t) - v(t), \\ \dot{v}(t) &= -p\big(v(t)\big) + \mathrm{sat}\big(u(t-\tau), v(t)\big), \end{aligned} \tag{1}$$

X. A. Ji and G. Orosz are with the Department of Mechanical Engineering and G. Orosz is also with the Department of Civil and Environmental Engineering, University of Michigan, Ann Arbor, MI 48109, USA; xunbij@umich.edu, orosz@umich.edu.

A. A. Gorodetsky is with the Department of Aerospace Engineering, University of Michigan, Ann Arbor, MI 48109, USA; goroda@umich.edu.

T. G. Molnár is with Department of Mechanical and Civil Engineering, California Institute of Technology, Pasadena, CA 91125, USA; tmolnar@caltech.edu.

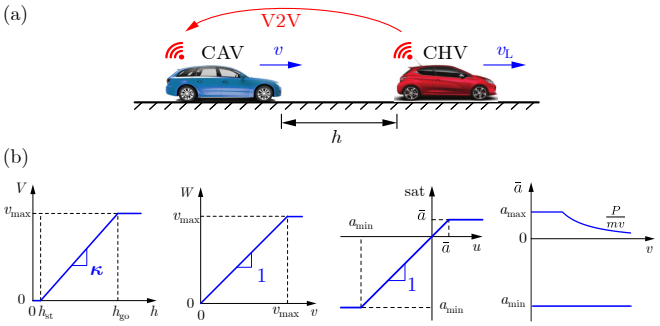

Fig. 1. (a) Experimental setup where a connected automated vehicle (CAV) follows a connected human-driven vehicle (CHV) based on the data transmitted through vehicle-to-vehicle (V2V) connectivity. (b) Nonlinearities used in the model.

where h is the distance headway between the vehicles, v is the speed of the CAV and v_{L} is the speed of the CHV; cf. Fig. 1(a). The control input u prescribes the CAV's desired acceleration and the saturation function

$$\mathrm{sat}(u,v)=\begin{cases} a_{\min} & u<a_{\min},\\ u & a_{\min}\le u\le \bar{a}(v),\\ \bar{a}(v) & u>\bar{a}(v),\end{cases} \qquad \bar{a}(v)=\min\left\{a_{\max},\frac{P}{mv}\right\} \tag{2}$$

includes the acceleration limits $a_{\min}$ and $a_{\max}$ while $\bar{a}(v)$ is determined by the power limit P and the mass m of the CAV. In this model, we consider the time delay τ associated with communication, computation and actuation. We also take into account resistances (such as air drag and rolling resistance) by a quadratic function

$$p(v)=0.001cv^2+0.01bv+0.1a. \tag{3}$$

Our aim is to infer the time delay τ and the coefficients c, b and a of the resistance terms from experimental data knowing the control algorithm of the CAV.

The desired acceleration of the CAV is assigned by the following control law [16], [17]:

$$u=\alpha\big(V(h)-v\big)+\beta\big(W(v_{\mathrm{L}})-v\big), \tag{4}$$

where α and β are control gains. The control law includes the range policy V that describes a desired speed according to the headway:

$$V(h)=\begin{cases}0 & h<h_{\mathrm{st}},\\ \kappa(h-h_{\mathrm{st}}) & h_{\mathrm{st}}\le h\le h_{\mathrm{go}},\\ v_{\max} & h>h_{\mathrm{go}},\end{cases} \tag{5}$$

and the speed policy W that gives a desired speed based on the lead CHV's speed:

$$W(v_{\mathrm{L}})=\begin{cases}v_{\mathrm{L}} & v_{\mathrm{L}}\le v_{\max},\\ v_{\max} & v_{\mathrm{L}}>v_{\max}.\end{cases} \tag{6}$$

The functions V, W, sat and $\bar{a}$ are depicted in Fig. 1 (b).

In the experiments [14], [15], controller (4) was used with parameters $\alpha=0.4\,\mathrm{s}^{-1}$, $\beta=0.5\,\mathrm{s}^{-1}$, $\kappa=0.6\,\mathrm{s}^{-1}$,

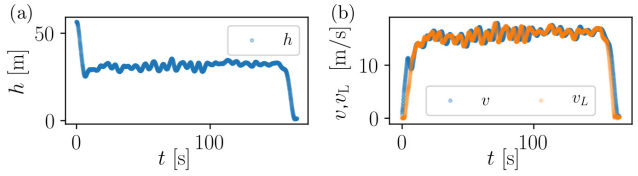

Fig. 2. Dataset with average velocity $v_{\mathrm{ave}}=15\,\mathrm{m/s}$.

$h_{\mathrm{st}}=5\,\mathrm{m}$, $v_{\max}=30\,\mathrm{m/s}$, $h_{\mathrm{go}}=h_{\mathrm{st}}+v_{\max}/\kappa=55\,\mathrm{m}$, $a_{\min}=-7\,\mathrm{m/s^2}$, $a_{\max}=3\,\mathrm{m/s^2}$, $P=50\,\mathrm{kW}$ and $m=1770\,\mathrm{kg}$. These are considered as known parameters, while the time delay τ and the air drag coefficients c, b and a are the unknowns to be determined from data. We chose 7 runs of experiments with different average velocity v_{ave} ranging from $2\,\mathrm{m/s}$ to $26\,\mathrm{m/s}$. One of the datasets with $v_{\mathrm{ave}}=15\,\mathrm{m/s}$ is shown in Fig. 2.

In what follows, we set up the Bayesian inference problem for the time delay system (1) and apply a Markov Chain Monte Carlo method to obtain the posterior distribution of τ, c, b and a for each dataset. We also compare different resistance models (that are the special cases of (3)) and analyze the inference results over the models and the datasets.

III. METHODOLOGY

In (1), we consider the state and input to be $X=[h,v]^\top$ and $U=v_{\mathrm{L}}$, respectively. Since the experimental data include both the position and the velocity of the two vehicles, the state X and the input U can all be observed. With X and U, the continuous time system (1) is of the form

$$\dot{X}(t)=f(X(t),X(t-\tau),U(t),U(t-\tau)). \tag{7}$$

We discretize the system using the measurement time step $\Delta t=0.1\,\mathrm{s}$, and the corresponding discrete time system can be written as

$$X^{k+1}=\Phi(\mathbf{X}^k,\mathbf{U}^k,\theta)=X^k+\Delta t f(\mathbf{X}^k,\mathbf{U}^k,\theta) \tag{8}$$

based on a forward Euler finite difference scheme, where

$$\begin{aligned}\mathbf{X}^k&=[h^k,v^k,h^{k-1},v^{k-1},\dots,h^{k-r},v^{k-r}]^\top,\\ \mathbf{U}^k&=[v_{\mathrm{L}}^k,v_{\mathrm{L}}^{k-1},\dots,v_{\mathrm{L}}^{k-r}]^\top.\end{aligned} \tag{9}$$

Here $\mathbf{X}^k$ and $\mathbf{U}^k$ contain the past states and inputs from time step $k-r$ to k, and r corresponds to the maximum allowed time delay $\tau_{\max}=r\Delta t$ while $\theta=[\tau,c,b,a]^\top$ contains the parameters we wish to learn.

Recall Bayes' rule for inference:

$$P(\theta|D)=\frac{\mathcal{L}(D|\theta)P(\theta)}{P(D)}, \tag{10}$$

where D is the given data, $P(D)$ is the marginal probability of the data, $\mathcal{L}(D|\theta)$ is the likelihood of the data under given parameters, $P(\theta)$ is the prior and $P(\theta|D)$ is the posterior. $P(D)$ acts as a normalizing constant, while the prior $P(\theta)$ represents the prior belief in the parameters before seeing the data and it can be used as regularization. The likelihood $\mathcal{L}(D|\theta)$ can be calculated for chosen parameters according

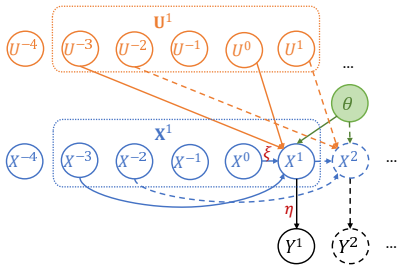

Fig. 3. Bayesian network of time delay systems.

to the noise model. The posterior $P(\theta|D)$ is what we are interested in computing.

The Bayesian network of time delay systems is represented in Fig. 3 for the example of a fixed time delay $\tau = 3\Delta t$ and maximum allowed time delay $\tau_{\max} = 4\Delta t$. The main difference between this network and Bayesian networks without delay is that the time delay τ affects the relationships between the states, i.e., state X^{k+1} not only depends on X^k and U^k but also on the preceding values of the state and input stored in $\mathbf{X}^k$ and $\mathbf{U}^k$. In this figure, ξ and η indicate the process noise and measurement noise, respectively. Since the states are fully-observed in our application, we assume that only one of the noise terms exists in the system so that the problem can be simplified. Namely, we consider two settings in the following subsection: (i) a time delay system with only measurement noise η and (ii) a time delay system with only process noise ξ. One can also consider both noises in the system [18], which is one of our future directions on the applications with time delay inference.

A. Posteriors for Time Delay Systems

If we assume that noise only exists in the measurement, based on (8) we can write the system equations as

$$\begin{aligned} X^k &= \Phi(\mathbf{X}^{k-1}, \mathbf{U}^{k-1}, \theta), \\ Y^k &= X^k + \eta, \quad \eta \sim \mathcal{N}(0, \Gamma), \end{aligned} \tag{11}$$

where Y^k is the measurement at time $t_k = k\Delta t$. For a given initial history, the states of the system are deterministic. In other words, X^k does not depend on the measurements. In this case, we can write the posterior as

$$P(\theta|D) \propto P(\theta) \prod_{k=1}^{N} P(Y^k|X^k), \tag{12}$$

where X^k is obtained by simulation for a given initial history and parameter set. As $P(Y^k|X^k) \sim \mathcal{N}(X^k, \Gamma)$, we can write the log-posterior as:

$$\log P(\theta|D) \propto \log P(\theta) - \frac{1}{2} \sum_{k=1}^{N} \left\| Y^k - X^k \right\|_{\Gamma}^2, \tag{13}$$

where $\|\cdot\|_{\Gamma}^2 = (\cdot)^\top (\Gamma)^{-1} (\cdot)$.

If we assume that the controller assigns the commanded acceleration based on measurement data and the measurements are filtered (which is the case in the experiment), then the error of the model can be viewed as a process noise ξ. Accordingly, the system can be described as

$$\begin{aligned} X^k &= \Phi(\mathbf{X}^{k-1}, \mathbf{U}^{k-1}, \theta) + \xi, \quad \xi \sim \mathcal{N}(0, \Sigma), \\ Y^k &= X^k. \end{aligned} \tag{14}$$

In this case, the future predicted states depend on the measurement as $X^k = \Phi(\mathbf{Y}^{k-1}, \mathbf{U}^{k-1}, \theta) + \xi$, where $\mathbf{Y}^{k-1}$ includes the history of measurements from time step $k-r-1$ up to time step $k-1$. Therefore, the posterior is written as

$$P(\theta|D) \propto P(\theta) \prod_{k=2}^{N} P(Y^k|\mathbf{Y}^{k-1}, \mathbf{U}^{k-1}, \theta), \tag{15}$$

where $P(Y^k|\mathbf{Y}^{k-1}, \mathbf{U}^{k-1}, \theta) \sim \mathcal{N}(\Phi(\mathbf{Y}^{k-1}, \mathbf{U}^{k-1}, \theta), \Sigma)$, that is, $P(Y^k - \Phi(\mathbf{Y}^{k-1}, \mathbf{U}^{k-1}, \theta)) \sim \mathcal{N}(0, \Sigma)$. Using the same difference scheme as in (8), we can rewrite this into

$$Y^k - Y^{k-1} - \Delta t f(\mathbf{Y}^{k-1}, \mathbf{U}^{k-1}, \theta) \sim \mathcal{N}(0, \Sigma). \tag{16}$$

Thus, we can write the log-posterior as

$$\begin{aligned} \log P(\theta|D) \propto \log P(\theta) \\ - \frac{1}{2} \sum_{k=2}^{N} \left\| \frac{Y^k - Y^{k-1}}{\Delta t} - f(\mathbf{Y}^{k-1}, \mathbf{U}^{k-1}, \theta) \right\|_{\Sigma/\Delta t^2}^2. \end{aligned} \tag{17}$$

We remark that for system (11), one needs to simulate the delayed dynamics for each choice of parameters to obtain the state X^k. It is computationally expensive if one uses sampling methods which require a large amount of simulations. As opposed, in system (14) we consider the difference of the measurements and the corresponding delay differential equation. This speeds up the sampling process significantly. Moreover, in this particular example with experimental data, the CAV made decisions based on measurements, and thus, it is more realistic to use the second system. Therefore, we focus on system (14) with only process noise to obtain the posterior distribution of the time delay and the resistance coefficients from the data. We achieve this by implementing the Delayed Rejection Adaptive Metropolis (DRAM) method.

B. DRAM for Time Delay Systems

We apply the Delayed Rejection Adaptive Metropolis Markov Chain Monte Carlo method [19] for parameter inference. In a standard Markov Chain Monte Carlo (MCMC) method, assuming we have the current sample (parameters) $\theta^{(i)}$, we propose a new sample θ based on proposal distribution, then we accept the new sample as the next sample $\theta^{(i+1)} = \theta$ with the probability $a(\theta^{(i)}, \theta)$ or reject it by setting $\theta^{(i+1)} = \theta^{(i)}$ with probability $1 - a(\theta^{(i)}, \theta)$. The distribution given by the resulting samples approaches the target distribution (the posterior distribution) as the number of samples goes to infinity. DRAM [19] combines delayed rejection [20] with adaptive Metropolis MCMC method [21]. Delayed rejection means that we do not reject the proposed sample for the first time but propose a second sample with a smaller scaled covariance instead. We delay the rejection decision, which increases the acceptance ratio and reduces

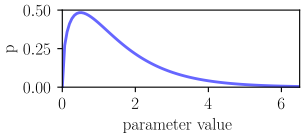

Fig. 4. Probability density function of the prior selected for all parameters.

the correlations between samples. Note that this delay is part of the sampling algorithm and is not related to the time delay τ we are trying to infer. The DRAM is adaptive because we utilize the covariance from obtained samples to propose new samples. The adaptive covariance in the proposal distribution helps to further reduce the correlation between samples. The DRAM used in this inference is described in Algorithm 1 with acceptance probabilities

$$\begin{aligned} a_1(\theta^{(i)},\theta_1) &= \min\left\{\frac{\pi(\theta_1)q_1(\theta^{(i)}|\theta_1)}{\pi(\theta^{(i)})q_1(\theta_1|\theta^{(i)})},1\right\},\\ a_2(\theta^{(i)},\theta_1,\theta_2) &= \min\left\{\frac{\pi(\theta_2)q_1(\theta_1|\theta_2)}{\pi(\theta^{(i)})q_1(\theta_1|\theta^{(i)})}\right.\\ &\quad\left.\times\frac{q_2(\theta^{(i)}|\theta_1,\theta_2)(1-a_1(\theta_2|\theta_1))}{q_2(\theta_2|\theta_1,\theta^{(i)})(1-a_1(\theta^{(i)}|\theta_1))},1\right\}. \end{aligned} \tag{18}$$

In this application, we use random walk proposal $q(\theta)$, and the posterior $\pi(\theta) = P(\theta|D)$. The prior distribution used for time delays should imply that the time delays are non-negative, thus, we choose the gamma prior shown in Fig. 4 for the time delay. In fact, the same prior is used for all other parameters since they are also non-negative and scaled to be of the same order of the magnitude as the delay via the scaling factors 0.001, 0.01 and 0.1 used in (3) for c, b, and a, respectively. Additionally, the time delay is limited below the maximum allowed value $\tau_{\max}$ to keep the length of prediction to be the same as the reference data. This can be achieved by choosing another specially designed prior distribution for the time delay, or by assigning $\pi(\theta) = 0$ for $\tau > \tau_{\max}$ as an extra regularization. We implemented the latter option for simplicity and set $\tau_{\max}$ large enough so that the effect of truncation becomes negligible (the area below the prior is larger than 0.99).

Data: Data D for inference
Result: Samples of θ distributed as $P(\theta|D)$
obtain a maximum a posteriori probability (MAP) estimate θ^{MAP} and initialize the first sample $\theta^{(0)} = \theta^{\mathrm{MAP}}$ with corresponding covariance $C^{(0)} = -(\nabla^2_\theta \log\pi(\theta^{\mathrm{MAP}}))^{-1}$
for $i = 0, ..., n$ **do**
  propose sample $\theta_1 = \theta^{(i)} + \zeta_1$, $\zeta_1 \sim \mathcal{N}(0, C^{(0)})$
  calculate the acceptance probability $a_1(\theta^{(i)}, \theta_1)$,
  draw a number $r_1 \sim \mathcal{U}[0,1]$,
  if $r_1 < a_1(\theta^{(i)}, \theta)$ **then**
    accept, $\theta^{(i+1)} = \theta_1$
  end
  else
    propose another sample $\theta_2 = \theta^{(i)} + \zeta_2$, $\zeta_2 \sim \mathcal{N}(0, 0.5C^{(0)})$
    calculate the acceptance probability $a_2(\theta^{(i)}, \theta_1, \theta_2)$,
    draw a number $r_2 \sim \mathcal{U}[0,1]$,
    if $r_2 < a_2(\theta^{(i)}, \theta_1, \theta_2)$ **then**
      accept, $\theta^{(i+1)} = \theta_2$
    end
    else
      reject, $\theta^{(i+1)} = \theta^{(i)}$
    end
  end
  update $C^{(i+1)} = g(C^{(i)})$
end
for $i = n+1, ...$ **do**
  follow the above steps except for proposing samples, use $\theta_1 = \theta^{(i)} + \zeta_1$, $\zeta_1 \sim \mathcal{N}(0, C^{(i)})$ and $\theta_2 = \theta^{(i)} + \zeta_2$, $\zeta_2 \sim \mathcal{N}(0, 0.5C^{(i)})$
end

Algorithm 1: Delayed Rejection Adaptive Metropolis (DRAM) algorithm [19].

IV. IMPLEMENTATION AND RESULTS

In the implementation, we choose the covariance matrix $\Sigma = \begin{bmatrix} 0.2^2 & 0 \\ 0 & 0.2^2 \end{bmatrix}$. The maximum allowed time delay $\tau_{\max}$ is set to $6\,\mathrm{s}$. We generate 10000 samples and discard the first half so that the remaining samples represent the posterior distribution. We consider three different cases of the quadratic resistance model (3) listed in Table I.

Model Name	Resistance Model $p(v)$
c	$0.001cv^2$
ca	$0.001cv^2 + 0.1a$
cba	$0.001cv^2 + 0.01bv + 0.1a$

TABLE I
DIFFERENT RESISTANCE MODELS.

We first combine all 7 datasets with different average velocities into one and infer the parameters from it. We show the marginal distributions given by 5000 samples for every parameter pair in Fig. 5. The maximum a posteriori probability (MAP) estimates are marked by red dots. The distributions of parameter τ and c are quite consistent among all the models, while c and b have a strong correlation in the cba model. Moreover, the distribution of the time delay τ has two modes, which indicates that there may exist two distinct time delays in the system, potentially corresponding to the different response times of the vehicle during acceleration and deceleration.

We also infer the parameters from each of the 7 individual datasets with different average velocity v_{ave}, and show how the data quality affects the inference. We plot the median value, 25th quantile and 75th quantile of the inferred parameters as a function of v_{ave} in Fig. 6. On panel (a), we can observe that the time delay consistently falls between $0.4\,\mathrm{s}$ to $0.75\,\mathrm{s}$ among all datasets, and there exist multiple modes in the marginal distribution of the delay since some sample distributions show a different peak from the one given by the MAP estimate. This further justifies that there might be two delays corresponding to different behaviors of the CAV, even for a single dataset. On panels (b) and (c), we can see large changes in the median values of c and b when the average velocity is low, although the coefficients should be independent of the velocity according to (3). The quality of inference on resistance coefficients depends on the average velocity. This implies that the resistance model is less reliable at low speeds and that the resistance forces do not affect the

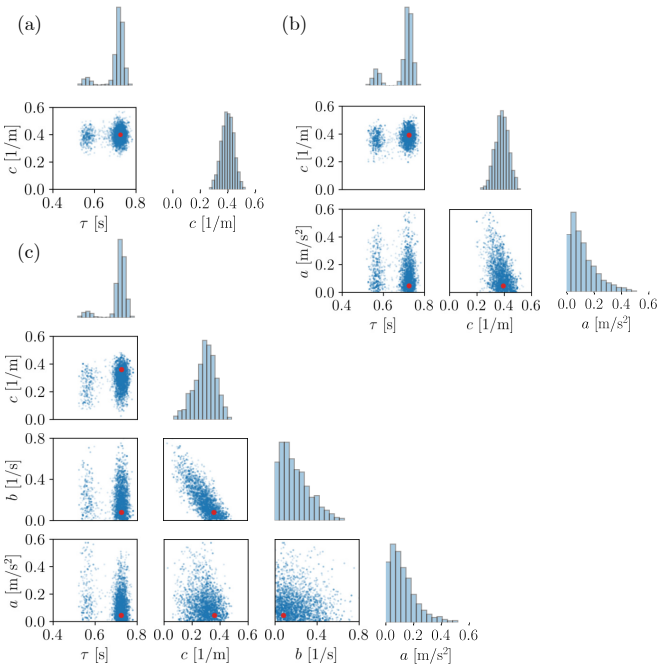

Fig. 5. Marginal distributions of the parameters obtained from the combined dataset: (a) c model; (b) ca model; (c) cba model. The MAP estimates are indicated by red dots.

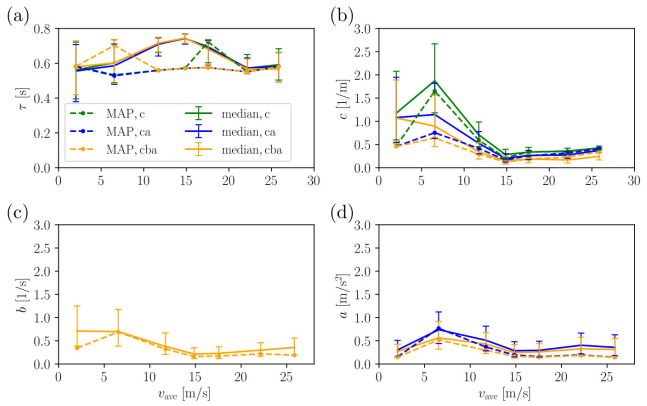

Fig. 6. Bayesian inference of 4 parameters using separate datasets: time delay τ and resistance coefficients c, b, and a, based on 7 datasets with different average velocity v_{ave}. Median, median plus minus 25th quantiles and MAP estimates are plotted for different resistance models.

car following behavior so significantly when the velocity is low. It also indicates that high velocity data are better for inferring the coefficients of resistance terms.

We plot the root mean square error (RMSE) of the state derivatives between those obtained from data and those given by the MAP and median estimates of the parameters in Fig. 7. The RMSE of $\dot{h}$ is exactly the same for each dataset regardless of the parameters. This is because $\dot{h}$ in (1) is not influneced by any parameter (both v_{L} and v are from data), and thus, the error in $\dot{h}$ cannot be controlled. The RMSE of $\dot{v}$ is affected by the parameters. The errors in all the models and estimates are close, they give overlapping curves.

We use the MAP estimates and 10 percent of the samples obtained for each dataset to simulate the system with the input (the speed of the lead CHV) taken from the experimental data. We show the simulation results for two datasets,

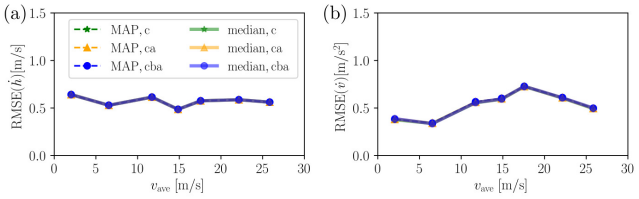

Fig. 7. RMSE in the state derivative given by the MAP and median estimates of the parameters for different resistance models. The estimates are obtained from separate datasets. Note that the curves obtained for the various scenarios overlap.

$v_{\mathrm{ave}} = 7\,\mathrm{m/s}$ and $v_{\mathrm{ave}} = 26\,\mathrm{m/s}$, with the cba model in Fig. 8. The simulation for $v_{\mathrm{ave}} = 7\,\mathrm{m/s}$ is plotted in Fig. 8(a) and (c). For this dataset the spreads of c, b and a are wide; see the yellow solid lines with error bars in Fig. 6. Yet, the simulation results lie within a narrow range as indicated by the shaded area. This demonstrates that uncertainties in the resistances are less relevant when the velocity is small. The simulation for $v_{\mathrm{ave}} = 26\,\mathrm{m/s}$ is plotted in Fig. 8(b) and (d). The simulation results enclose a wide region, and both the MAP simulation and the median of 500 simulations give small velocity error but large headway error between simulation and data. The headway error indicates that there exist unmodeled dynamics, and the dataset with large speed is more sensitive to errors in the state derivative. This sensitivity is caused by reaching the saturation limits depicted in Fig. 1(b) (where the range policy V saturates for large headway $h > h_{\mathrm{go}}$).

Although its control law was predefined, the CAV's response has uncertainties in real experiments. These uncertainties can be considered as part of the unmodeled dynamics, which cannot be captured by our current model. In order to increase the flexibility of the model and further reduce its error, calibrations are needed for more parameters. Furthermore, minimizing the error in the state derivative (which was the case in our method) does not always lead to minimal error in closed-loop simulations, since there is measurement noise in data and additional error can be introduced by numerical differentiation when obtaining the state derivative. To reduce the headway error in the simulation, one may need to calibrate more parameters, include closed-loop simulations when calculating the posterior, or infer both states and parameters together from data.

To further examine the MAP and median estimates inferred from each dataset, we simulate the system with the MAP and median estimates and plot the RMSE of headway and velocity in Fig. 9. Note that the simulation with the median estimate is different from the median of the 500 simulations in Fig. 8. First, we observe in Fig. 9 that the errors in both headway and velocity increase as the average velocity increases from $2\,\mathrm{m/s}$ to $18\,\mathrm{m/s}$. Most of the datasets have relatively small simulation RMSE in both headway and velocity except the dataset with largest average velocity $v_{\mathrm{ave}} = 26\,\mathrm{m/s}$ (for the reasons discussed above). Second, the three resistance models do not show significant difference in simulation, which indicates that using the dominant term

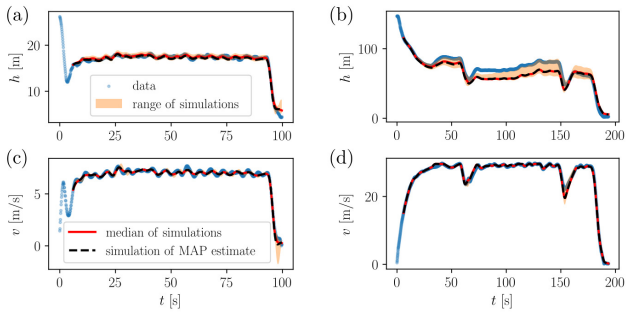

Fig. 8. Experimental data and simulation results for the cba model. The shaded area shows the range of simulations using 500 samples and solid red line gives the median of those simulations. The dashed black line is the simulation based on the MAP estimate. (a), (c) show the simulation results for dataset $v_{\text{ave}} = 7\,\text{m/s}$ with MAP estimate: $\tau = 0.70\,\text{s}$, $c = 0.64\,\text{m}^{-1}$, $b = 0.69\,\text{s}^{-1}$, $a = 0.52\,\text{m/s}^2$; while (b), (d) show the results for dataset $v_{\text{ave}} = 26\,\text{m/s}$ with MAP estimate: $\tau = 0.58\,\text{s}$, $c = 0.34\,\text{m}^{-1}$, $b = 0.19\,\text{s}^{-1}$, $a = 0.14\,\text{m/s}^2$.

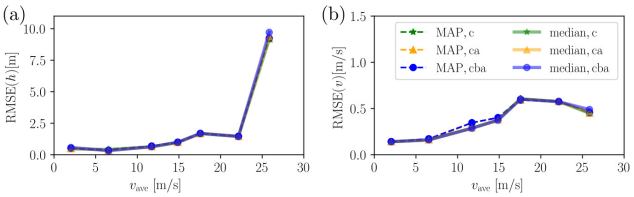

Fig. 9. Simulation error given by the MAP and median estimates of the parameters for different resistance expressions. The estimates correspond to separate datasets. The curves obtained for the various scenarios overlap.

$0.001cv^2$ with the appropriate value for c may be sufficient to capture the effect of resistance forces. Third, the median and the MAP estimate give similar simulation RMSE. This implies that the MAP estimate can be used as an alternative to the median value. Since the MAP estimate is much faster to obtain, it can be used for real-time parameter inference, potentially in a time-varying fashion; but this topic is beyond the scope of this paper.

V. CONCLUSIONS AND FUTURE WORK

Our results showed a successful implementation of the Delayed Rejection Adaptive Metropolis (DRAM) method for inferring the time delay and resistance coefficients of a connected automated vehicle from field data. We observed multiple modes in the distribution of the delay parameter indicating the existence of different time delays corresponding to acceleration and braking. The time delay estimations were consistent between all the datasets, giving a value around $0.5\,\text{s}$ to $0.7\,\text{s}$. The time delay did not have significant correlation with the resistance terms, while we observed strong correlation between linear and quadratic resistance coefficients. The DRAM sampling method gave the distribution of the parameters so that we could show the uncertainty of the prediction. The maximum a posteriori probability (MAP) estimate was computationally fast and could potentially be implemented for real-time inference while providing reasonable prediction. A future application can be to identify the lateral dynamics for connected automated vehicles.

REFERENCES

[1] H. Ozaki, "Reaction and anticipation in the car-following behavior," in *the 12th International Symposium on Theory of Traffic Flow and Transportation*, 1993, pp. 349–366.

[2] M. Pourabdollah, E. Bjärkvik, F. Fürer, B. Lindenberg, and K. Burgdorf, "Calibration and evaluation of car following models using real-world driving data," in *20th IEEE International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2017, pp. 1–6.

[3] Y. Li, L. Zhang, H. Zheng, X. He, S. Peeta, T. Zheng, and Y. Li, "Evaluating the energy consumption of electric vehicles based on car-following model under non-lane discipline," *Nonlinear Dynamics*, vol. 82, no. 1, pp. 629–641, 2015.

[4] C. P. I. van Hinsbergen, H. W. van Lint, S. P. Hoogendoorn, and H. J. van Zuylen, "Bayesian calibration of car-following models," *IFAC Proceedings Volumes*, vol. 42, no. 15, pp. 91–97, 2009.

[5] M. Rahman, M. Chowdhury, T. Khan, and P. Bhavsar, "Improving the efficacy of car-following models with a new stochastic parameter estimation and calibration method," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 5, pp. 2687–2699, 2015.

[6] M. Kasai, S. Shibagaki, and S. Terabe, "Application of hierarchical Bayesian estimation to calibrating a car-following model with time-varying parameters," in *IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2013, pp. 870–875.

[7] M. Kasai and J. Xing, "Use of particle filtering to establish a time-varying car-following model," *International Journal of Intelligent Transportation Systems Research*, vol. 17, no. 1, pp. 49–60, 2019.

[8] Y. Wang, G. Gunter, M. Nice, M. L. Delle Monache, and D. Work, "Online parameter estimation methods for adaptive cruise control systems," *IEEE Transactions on Intelligent Vehicles*, 2020.

[9] Z. Ding, D. Xu, H. Zhao, M. Moze, F. Aioun, and F. Guillemard, "Driver identification through multi-state car following modeling," in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*. IEEE, 2019, pp. 1580–1587.

[10] Y. Wang, G. Gunter, and D. B. Work, "Online parameter estimation of adaptive cruise control models with delays and lags," in *23rd IEEE International Conference on Intelligent Transportation Systems*. IEEE, 2020, pp. 1–6.

[11] B. Calderhead, M. Girolami, and N. D. Lawrence, "Accelerating Bayesian inference over nonlinear differential equations with Gaussian processes," in *Advances in Neural Information Processing Systems*, 2009, pp. 217–224.

[12] M. Lentmaier, B. Krach, and P. Robertson, "Bayesian time delay estimation of GNSS signals in dynamic multipath environments," *International Journal of Navigation and Observation*, vol. 2008, 2008.

[13] B. Choi, Y.-Y. Cheng, S. Cinar, W. Ott, M. R. Bennett, K. Josić, and J. K. Kim, "Bayesian inference of distributed time delay in transcriptional and translational regulation," *Bioinformatics*, vol. 36, no. 2, pp. 586–593, 2020.

[14] S. S. Avedisov, G. Bansal, A. K. Kiss, and G. Orosz, "Experimental verification platform for connected vehicle networks," in *21st IEEE International Conference on Intelligent Transportation Systems*, Maui, HI, USA, 2018, pp. 818–823.

[15] X. A. Ji, T. G. Molnár, S. S. Avedisov, and G. Orosz, "Feed-forward neural networks with trainable delay," in *Learning for Dynamics and Control*. PMLR, 2020, pp. 127–136.

[16] L. Zhang and G. Orosz, "Motif-based design for connected vehicle systems in presence of heterogeneous connectivity structures and time delays," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 6, pp. 1638–1651, 2016.

[17] J. I. Ge, S. S. Avedisov, C. R. He, W. B. Qin, M. Sadeghpour, and G. Orosz, "Experimental validation of connected automated vehicle design among human-driven vehicles," *Transportation Research Part C*, vol. 91, pp. 335–352, 2018.

[18] N. Galioto and A. A. Gorodetsky, "Bayesian system ID: optimal management of parameter, model, and measurement uncertainty," *Nonlinear Dynamics*, vol. 102, no. 1, pp. 241–267, 2020.

[19] H. Haario, M. Laine, A. Mira, and E. Saksman, "DRAM: efficient adaptive MCMC," *Statistics and Computing*, vol. 16, no. 4, pp. 339–354, 2006.

[20] A. Mira, "On Metropolis-Hastings algorithms with delayed rejection," *Metron*, vol. 59, no. 3-4, pp. 231–241, 2001.

[21] H. Haario, E. Saksman, and J. Tamminen, "An adaptive Metropolis algorithm," *Bernoulli*, vol. 7, no. 2, pp. 223–242, 2001.