NETWORKS AND HETEROGENEOUS MEDIA
©American Institute of Mathematical Sciences
Volume **8**, Number **3**, September **2013**
doi:10.3934/nhm.2013.8.803
pp. **803–824**

MARKOV CHAIN MONTE CARLO BASED INVERSE MODELING OF TRAFFIC FLOWS USING GPS DATA

Olli-Pekka Tossavainen

Microsoft
1065 La Avenida St
Mountain View, CA 94043, USA

Daniel B. Work

University of Illinois at Urbana-Champaign
1203 Newmark Civil Engineering Laboratory, 205 N. Mathews Ave
Urbana, IL 61801, USA

Abstract. In large scale deployments of traffic flow models, estimation of the model parameters is a critical but cumbersome task. A poorly calibrated model leads to erroneous estimates in data–poor environments, and limited forecasting ability. In this article we present a method for calibrating flow model parameters for a discretized scalar conservation law using only velocity measurements. The method is based on a Markov Chain Monte Carlo technique, which is used to approximate statistics of the posterior distribution of the model parameters. Numerical experiments highlight the difficulty in estimating jam densities and provide a new approach to improve performance of the sampling through re-parameterization of the model. Supporting source code for the numerical experiments is available for download at https://github.com/dbwork/MCMC-based-inverse-modeling-of-traffic.

1. Introduction.

1.1. Motivation.

This article is motivated by a practical problem encountered when deploying flow models at the scale of cities, states, or countries, for traffic monitoring and prediction (e.g. [25, 26]). When the models are deployed at large scales, it is necessary to load the network topology from commercial or open source maps, from which we can easily obtain properties of the roadways, such as speed limits, and the number of lanes. This information is used to construct prior estimates for the flow model parameters. However, any errors in the map database, especially on the number of lanes, can cause significant error in the ability of the flow model to correctly predict the future evolution of traffic. Because the maps are usually constructed for navigation purposes, and not for building network traffic flow models, these errors can go undetected until after the flow model is deployed. Thus, there is a real need to be able to compute an accurate posterior distribution for the parameters using traffic observations, which will reduce the sensitivity of the flow model traffic predictions on any errors in the map database.

As GPS data becomes ubiquitous, the coverage of systems that utilize traffic flow models can be dramatically increased. The proliferation of GPS data has enabled

2010 *Mathematics Subject Classification.* 90B20, 65C05, 65C40, 35L65, 65N21.
Key words and phrases. Parameter estimation, fundamental diagram, Markov Chain, Monte Carlo, traffic flow, cell transmission model.

private companies and research institutes to collect large amounts of GPS data, but little research has gone into developing effective methods to perform offline calibration of flow models using these data archives.

1.2. **Problem statement.** In this work we address the problem of determining the parameters of a traffic flow model using observations obtained from GPS data. The traffic dynamics are encoded in a mathematical model f which evolves the traffic state x^n (e.g. the traffic density or velocity) at time n, given some time–invariant model parameters θ. The traffic evolution and measurement generation is given by the following *evolution–observation* system,

$$\begin{cases} x^{n+1} &= f\left(x^n, \theta\right) \\ y^n &= g^n(x^n), \end{cases}$$

where y^n is a vector of measurements, and g^n is a time-varying observation operator, which maps the traffic state to the observed measurements.

Letting $y = \left(\left(y^1\right)^{\mathrm{T}}, \left(y^2\right)^{\mathrm{T}}, \cdots, \left(y^N\right)^{\mathrm{T}}\right)^{\mathrm{T}}$ denote the concatenated measurement vector, we define the *forward model,* h as the function which maps the model parameters to the observations,

$$y = h(\theta).$$

Solving the forward problem requires evolving the observation evolution system forward in time for a given model parameter vector θ.

We define the *inverse problem* (also called the calibration or parameter estimation problem) as the task of estimating the flow model parameters θ given the measurements y. In general, as well as in the specific problem investigated in this work, the inverse problem is ill–posed. For example, this can occur when the measurements are insufficient to uniquely determine the parameters, or because too many measurements lead the problem to be overdetermined. Thus, we formulate the inverse problem using Bayesian formalism [13], in which the solution of the inverse problem is the posterior probability density of the model parameters θ given observations y.

Because the posterior density contains a numerical solution of a flow model, we cannot write analytical expressions for the mean value and covariance of the posterior distribution. Instead we generate samples from the posterior density and use Monte Carlo techniques to estimate these statistics. The approach used in this work is based on *Markov Chain Monte Carlo* (MCMC) sampling of the posterior probability density [10].

In this work we restrict ourselves to the problem of estimating time–invariant parameters of the traffic flow model in consideration. However, time varying parameters can also be addressed through the proposed MCMC approach, provided an evolution equation for the parameters is specified. This is an important extension in practice, as unknown boundary conditions can be treated as time-varying parameters to be estimated. The accurate recovery of boundary condition parameters is also important to achieve good forecasts from the traffic model.

1.3. **Related work and contributions of this article.** To our best knowledge this is the first attempt to solve the time–invariant parameter estimation problem using only speed observations, by embedding a flow model in the inversion. This is in contrast to earlier methods [8, 18, 20, 21] in which the parameter estimation is performed directly on the fundamental diagram (i.e. flux function), using constrained least squares fitting or machine learning techniques, without the flow

model. Unlike the problem of estimating flow model parameters from fixed sensors, the temporal and spatial varying nature and the sparsity of the GPS measurements prevents inversion directly on the flux function, and motivates our approach. In cases where it is possible to obtain high resolution GPS data, the fundamental diagram can be directly calibrated using regression methods [7]. Nonlinear programming has also been used to calibrate traffic flow model parameters [6]. In [1], a repeated calibration of flow model parameters using nonlinear least squares method was used to improve model predictive control.

A class of online state and parameter estimation problems which work directly on the flow model has also been considered [2, 12, 17, 27, 29] in the presence of speed and density data from inductive loop detectors. In these works, the traffic state and the flow model parameters are estimated using an extended Kalman filter. The approaches show good performance when inductive loop detector data is available.

From a computational perspective, we also discuss the problem generating good samples from the posterior density using a Markov Chain Monte Carlo approach and our specific choice of a traffic flow model. We demonstrate that a variable change on the parameters to be estimated can significantly improve the posterior exploration, and the performance of the method.

The remainder of the article is organized as follows. In Section 2 we present the specific traffic evolution dynamics considered in this work. In Section 3 we review the construction of the posterior density and give the Markov Chain Monte Carlo algorithm used to estimate the flow model parameters. In Section 4 we describe the setting for our numerical experiments. We also discuss improvements to the MCMC method for the flow model considered in this work. In Section 5 we give the results of the simulations and in Section 6 we give conclusions and suggestions for future work.

2. **Traffic flow model.**

The traffic flow model employed in this work is the velocity evolution equation proposed in [28]. Here we repeat the essential parts of the model.

We model traffic flows using the *Lighthill-Whitham-Richards* (LWR) [16, 22] *partial differential equation* (PDE), which is a nonlinear hyperbolic conservation law that describes the evolution of the traffic density ρ on a stretch of highway of length L over a time horizon T. The LWR PDE can be written as:

$$\frac{\partial \rho(x,t)}{\partial t} + \frac{\partial \left(\rho(x,t) V_\theta\left(\rho\right)\right)}{\partial x} = 0, \;\; (x,t) \in (0,L) \times (0,T) \tag{1}$$

with initial and boundary conditions given by

$$\begin{cases} \rho(x,0) = \rho_0(x) \\ \rho(0,t) = \rho_l(t), \;\; \rho(L,t) = \rho_r(t), \end{cases}$$

and where $\rho_0(\cdot)$, $\rho_l(\cdot)$ and $\rho_r(\cdot)$ are the initial, left, and right boundary data, respectively.

A velocity function $V_\theta(\cdot)$ is defined to link the density to the average speed of traffic, thereby closing the LWR model. In this work we assume the Smulders [23] velocity function, which can be written as follows:

$$V_\theta(\rho) = \begin{cases} v_{\max}\left(1 - \frac{\rho}{\rho_{\max}}\right) & \text{if } \rho \leq \rho_c \\ -w_f\left(1 - \frac{\rho_{\max}}{\rho}\right) & \text{otherwise.} \end{cases}$$

The parameters of the Smulders velocity function, $\theta = (\rho_{\max}, v_{\max}, w_f)^{\mathrm{T}}$, are the maximum density $\rho_{\max}$, the maximum velocity $v_{\max}$, and the maximum negative wave speed w_f. For continuity of the flux function, the critical density ρ_c is defined as

$$\rho_c = \frac{\rho_{\max} w_f}{v_{\max}}.$$

Note that the Smulders velocity function is invertible, and the mapping from velocity v to density ρ can be written as

$$\rho = V_\theta^{-1}(v) = \begin{cases} \rho_{\max}\left(1 - \frac{v}{v_{\max}}\right) & \text{if } v \geq v_c \\ \rho_{\max}\left(\frac{1}{1+\frac{v}{w_f}}\right) & \text{otherwise,} \end{cases} \tag{2}$$

where $v_c = V_\theta(\rho_c)$ is the critical velocity.

As shown in [28], we can obtain a discrete velocity evolution equation, called the *Cell Transmission Model for velocity* (CTM-v), by applying a Godunov [11, 14] scheme directly to (1) and then applying a velocity inversion. The discretized evolution can be written as follows:

$$v_i^{n+1} = V_\theta\left(V_\theta^{-1}(v_i^n) - \frac{\Delta T}{\Delta x}\left(\tilde{G}_\theta(v_i^n, v_{i+1}^n) - \tilde{G}_\theta(v_{i-1}^n, v_i^n)\right)\right), \tag{3}$$

where ΔT is the discrete time step indexed by $n \in \{0, \ldots, n_{\max}\}$, Δx is the discrete space step indexed by $i \in \{0, \ldots, i_{\max}\}$, v_i^n is the traffic velocity in the i^{th} cell at time step n.

In (3), $\tilde{G}_\theta(\cdot, \cdot)$ is the Godunov velocity flux, which is given by

$$\tilde{G}_\theta(v_1, v_2) = \begin{cases} v_2 \rho_{\max}\left(\frac{1}{1+\frac{v_2}{w_f}}\right) & \text{if } v_c \geq v_2 \geq v_1 \\ v_c \rho_{\max}\left(1 - \frac{v_c}{v_{\max}}\right) & \text{if } v_2 \geq v_c \geq v_1 \\ v_1 \rho_{\max}\left(1 - \frac{v_1}{v_{\max}}\right) & \text{if } v_2 \geq v_2 \geq v_c \\ \min\left(V_\theta^{-1}(v_1)v_1, V_\theta^{-1}(v_2)v_2\right) & \text{if } v_1 \geq v_2. \end{cases}$$

The velocity evolution of the first and last cells in the domain v_0^n and $v_{i_{\max}}^n$ is governed by the following equations:

$$v_0^{n+1} = V_\theta\left(V_\theta^{-1}(v_0^n) - \frac{\Delta T}{\Delta x}\left(\tilde{G}_\theta(v_0^n, v_1^n) - \tilde{G}_\theta(v_l^n, v_0^n)\right)\right),$$
$$v_{i_{\max}}^{n+1} = V_\theta\left(V_\theta^{-1}(v_{i_{\max}}^n) - \frac{\Delta T}{\Delta x}\left(\tilde{G}_\theta(v_{i_{\max}}^n, v_r^n) - \tilde{G}_\theta(v_{i_{\max}-1}^n, v_{i_{\max}}^n)\right)\right),$$

where v_l^n and v_r^n are given by the prescribed left and right boundary conditions defined in terms of velocity. Extension of the model to networks is given in [28].

An important feature of the velocity dynamics assumed in this work (i.e. the Smulders velocity function) is that the wave speed s of a shock on a homogeneous roadway is not a function of the jam density. This can be shown as follows. The speed of a shock connecting the two states v_1 and v_2 is given by the *Rankine–Hugoniot* relation:

$$s = \frac{V_\theta^{-1}(v_1)v_1 - V_\theta^{-1}(v_2)v_2}{V_\theta^{-1}(v_1) - V_\theta^{-1}(v_2)}.$$

By writing the velocity inversion function (2) as $V_\theta^{-1}(v) = \rho_{\max} Z_\theta(v)$ where

$$Z_\theta(v) = \begin{cases} 1 - \frac{v}{v_{\max}} & \text{if } v \geq v_c \\ \frac{1}{1+\frac{v}{w_f}} & \text{otherwise} \end{cases}$$

the shock speed reduces to

$$s = \frac{Z_\theta(v_1)v_1 - Z_\theta(v_2)v_2}{Z_\theta(v_1) - Z_\theta(v_2)}, \tag{4}$$

which does not depend on the jam density.

Moreover, at the boundary of two links with changing parameters, conservation of vehicles across the junction is imposed by the Rankine–Hugoniot junction condition:

$$s = V_{\theta_1}^{-1}(v_1)v_1 - V_{\theta_2}^{-1}(v_2)v_2 = 0$$

where θ_1 and θ_2 denote the velocity function parameters of the incoming link and outgoing link. Defining α as the ratio of the jam densities, $\alpha = \rho_{\max,1}/\rho_{\max,2}$, the junction condition can be written as:

$$s = \alpha Z_{\theta_1}(v_1)v_1 - Z_{\theta_2}(v_2)v_2 = 0. \tag{5}$$

Thus, as long as the ratio of the jam densities between adjacent links satisfies (5), the Rankine–Hugoniot condition between links will be satisfied.

From (4) and (5) above, we can draw the following conclusions. First, within a link, the shock wave speed does not depend on $\rho_{\max}$. Second, when a discontinuity occurs at a link boundary, the jump is a function of the jam density ratio α, but not directly the jam density. From the perspective of estimating the value of the jam density on each link, this will significantly increase the degree of difficulty in generating good jam density samples from the posterior parameter distribution.

Note that for other choices of a velocity function, for example the California model [9], may introduce dependencies in (2) on the specific values of the jam densities. Regardless, examining the dependency of jump condition on the model parameters may be helpful for identifying other efficient sampling strategies. This is because parameter estimates which generate incorrect shock speeds are easily rejected in our proposed MCMC algorithm.

2.1. **Simulation of vehicle trajectories.** We model the evolution of GPS equipped vehicles as passive Lagrangian tracers, which evolve according to the macroscopic (average) velocity field. Hence, the j^{th} vehicle moves with the local traffic speed and updates its position χ_j according to:

$$\dot{\chi}_j(t) = V_\theta\left(\rho\left(\chi_j(t), t\right)\right). \tag{6}$$

Problems of the form (1) and *ordinary differential equation* (ODE) (6) have been studied in [4, 5], and a numerical method for approximating solutions on networks via an extension of the Godunov scheme was developed in [4]. The numerical method can be understood by recognizing the Godunov scheme as a *reconstruct evolve average* (REA) scheme [15]. In an REA scheme, the first step is to reconstruct a piecewise constant approximation of the density field. In the second step, the reconstructed density field is evolved exactly up to time ΔT by application of the Riemann solvers at each cell boundary. Finally, at time ΔT, the density in each cell is averaged, so that a piecewise constant density function can be reconstructed

and the process is repeated. The key idea in [4] is to update the vehicle position immediately after the *evolve* step but before the *average* step in the Godunov scheme, instead of updating the position after the average step has occurred.

To simulate vehicle trajectories in our numerical experiments, we implement the trajectory simulation scheme for the Godunov method [4] for the specific case of the Smulders velocity function on a linear network. We use a time step of $\Delta T = 1$ sec for the coupled ODE PDE system (1) and (6) to simulate the vehicle trajectories. The speed reported by the vehicle is assumed to be the space mean speed, computed by dividing the distance traveled during the time step, by the time step. This is done because in practice, instantaneous GPS speed measurements are typically averaged over some distance or time to reduce the noise of the reported speeds. Finally, a Gaussian measurement noise with standard deviation of 2 mph is added to the speed to create the synthetic measurements. The complete algorithm for the vehicle tracking scheme applied to the Smulders diagram is detailed in Appendix A.

3. Bayesian estimation of model parameters.

In the Bayesian setting, the solution to our flow model parameter estimation problem is a posterior density of the parameters, conditioned on the observed GPS measurements. In this section, we construct the parameter observation model and the posterior density for the unknown model parameters. Because of the nonlinearity of the observation operator, which itself contains a nonlinear flow model, we are unable to write an analytical expressions for quantities such as mean value of the posterior and its covariance. To circumvent this difficulty encountered in the estimation process, we propose a *Markov Chain Monte Carlo* (MCMC) method to explore the posterior density and generate random samples from it. This approach allows us to compute properties of the posterior distribution of the flow model parameters (e.g. moments of the distribution) using sample approximations, which can then be used as the calibrated flow model parameters and their uncertainty estimates.

3.1. Construction of the posterior density.

The observations used in our flow model parameter estimation problem are GPS speeds corrupted by measurement noise, which are obtained from vehicles traveling through the computational domain.

Let $y \in \mathbb{R}^m$ denote the vector of these observations. The observation model is assumed to be an additive noise model, which is written as

$$y = h(\theta) + \varepsilon, \tag{7}$$

where $\theta \in \mathbb{R}^d$ is the vector of unknown model parameters. The additive noise $\varepsilon \in \mathbb{R}^m$ is assumed to be Gaussian, $\varepsilon \sim \mathcal{N}(0, \Gamma_\varepsilon)$ and independent of the parameters θ. For simplicity, we assume that the locations at which the measurements are obtained is known.

Unlike the sequential state estimation methods based on variants of Kalman or particle filtering [3, 12, 19, 27, 29], (7) is an off-line, stationary estimation problem. Embedded within the parameter observation operator h is the forward evolution of the velocity field governed by the CTM-v (3), which is mapped onto the location where GPS measurements are obtained.

Using the above observation model we can write the likelihood function as:

$$p(y|\theta) \propto \exp\left(-\frac{1}{2}\left(y - h(\theta)\right)^{\mathrm{T}} \Gamma_\varepsilon^{-1} \left(y - h(\theta)\right)\right).$$

The posterior density of the model parameters is written using Bayes rule as

$$p(\theta|y) \propto p_{\mathrm{pr}}(\theta)p(y|\theta),$$

where $p_{\mathrm{pr}}(\theta)$ is the prior probability density for the unknown parameters.

The prior probability density on the model parameters represents our initial beliefs about the unknown model parameters. Because the flow model parameters have physical interpretations, the prior distribution can be defined as a distribution of physically allowable values. For example, a prior distribution on the jam density $\rho_{\max}$ can be obtained by considering the minimum and maximum number of vehicles which can be stored on a lane, given a minimum and maximum vehicle length. Engineering references such as the *Highway Capacity Manual* [24] can also be used to help construct the priors. The prior distributions on the parameters chosen for our flow model are described in Section 5.

3.2. **Exploring the posterior using a Markov Chain Monte Carlo method.** We use a Markov Chain Monte Carlo method to generate an ensemble of samples from the posterior density. Using these samples we can compute estimates for the unknown model parameters such as conditional mean values. More specifically, we use a method called *Metropolis–Hastings* (MH) to generate the samples. Another popular MCMC technique is called *Gibbs sampler*, see for example [10].

The Metropolis–Hastings algorithm is a well known method for generating samples from the posterior, which can be summarized using following four steps [13]:

1. Select an initial parameter value $\theta_1 \in \mathbb{R}^d$ and set $k = 1$.
2. Draw $z \in \mathbb{R}^d$ from the (not necessarily symmetric) *proposal density* $q(\theta_k, z)$ and calculate the *acceptance ratio*
$$\lambda(\theta_k, z) = \min\left(1, \frac{p(z|y)q(z,\theta_k)}{p(\theta_k|y)q(\theta_k,z)}\right).$$
3. Draw $u \sim \mathcal{U}[0,1]$, where $\mathcal{U}$ is the uniform distribution.
4. If $\lambda(\theta_k, z) \geq u$, set $\theta_{k+1} = z$, else $\theta_{k+1} = \theta_k$. If k equals the desired sample size K, stop, else $k \leftarrow k+1$ and go to step 2.

For practical implementation of the algorithm, we choose the proposal density to be Gaussian:

$$q(\theta, z) \propto \exp\left(-\frac{1}{2}(\theta - z)^{\mathrm{T}}\Gamma_w^{-1}(\theta - z)\right),$$

where Γ_w is assumed to be of the form: $\mathrm{diag}(\gamma)$, $\gamma \in \mathbb{R}^d$. In other words, the random step from θ to z is distributed as white noise,

$$w = z - \theta \sim \mathcal{N}(0, \Gamma_w).$$

Note that the above proposal density is symmetric, which simplifies the computation of the acceptance ratio (also called the acceptance probability) $\lambda(\theta, z)$. By choosing the proposal density as Gaussian, we obtain a simple random walk model called Random Walk Metropolis–Hastings algorithm which is summarized in Algorithm 1 [13].

3.3. **Mixing of the chain.** In Algorithm 1, the diagonal elements of matrix Γ_w control the step length of the random walk on the parameter vector. The choice of these values affects the *acceptance rate* (how many samples are rejected versus accepted) and the *mixing* of the chain. Mixing refers to the speed at which the chain moves around the support of $p(\theta|y)$. Large steps lead to few accepted samples because large steps tend to propose moves from the body of the density to the

Algorithm 1 Random walk Metropolis–Hastings

```
Pick initial value θ_1
Set θ = θ_1
for k = 2 : K do
    Calculate p(θ|y)
    Draw w ~ N(0, Γ_w) and set z = θ + w (proposal step)
    Calculate p(z|y)
    Calculate λ(θ, z) = min(1, p(z|y)/p(θ|y))
    Draw u ~ U[0, 1]
    if u ≤ λ(θ, z) then
        Accept: Set θ = z, θ_k = θ
    else
        Reject: Set θ_k = θ
    end if
end for
```

tails of the density, resulting in small acceptance ratios. Such a chain will not move frequently and thus results in slow mixing. On the other hand, a small step length means that the process explores the posterior density slowly but with a high acceptance rate. A good step length should avoid both of these extremes. Techniques for step length selection have been proposed for example in [10].

4. **Description of the numerical experiment.** In this section, we describe a numerical experiment which we use to assess the potential of the MCMC approach for estimating flow model parameters. We consider a two–mile stretch of road, which consists of three links, each with an unknown number of lanes. The link lengths are given as $l_1 = 0.62$ mi, $l_2 = 0.49$ mi, and $l_3 = 0.89$ mi. We assume the maximum velocity $v_{\max}$ and the maximum backwards shock wave speed w_f is constant across the domain, but that the number of lanes on each of the links may vary. Because the number of lanes has a strong influence on the road capacity and the jam density $\rho_{\max}$, we assume the jam density on each link is modeled with a separate parameter.

We generate synthetic data by forward simulation of the CTM-v, assuming the *true* values of the parameters are as follows. We assume the shock wave speed $w_f = 16$ mph and $v_{\max} = 77$ mph. The number of lanes in link 1 is five and the jam density $\rho_{\max,1}$ is 5×180 veh/mi. Link 2 has four lanes and the jam density $\rho_{\max,2}$ is 4×170 veh/mi. The number of lanes in link 3 is five and the jam density $\rho_{\max,3}$ is 5×160 veh/mi. The initial velocity is assumed to be 77 mph on all links, and the velocity boundary conditions are time varying, as shown in Figure 1.

The time step ΔT used in the Godunov scheme when creating the synthetic data is set to 1 second in order to achieve CFL number ≤ 0.5. Each link is discretized in space such that link 1 contains 14 cells, link 2 contains 11 cells, and link 3 contains 20 cells. This choice of discretization guarantees the numerical stability of the model up to a maximum velocity of 80 mph, which is important since the CFL condition for numerical stability is a function of an unknown parameter ($v_{\max}$) to be estimated.

The velocity field obtained using the above parameters is shown in Figure 2, over a period of 17 minutes. The noise free vehicle trajectories are simulated using this velocity field. A vehicle enters the domain at postmile 0 every 30 seconds, and the

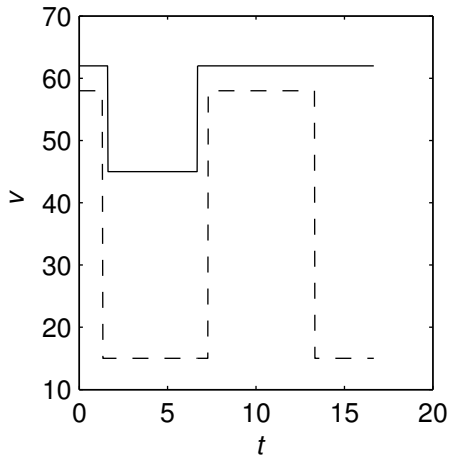

FIGURE 1. Upstream (solid) and downstream (dashed) velocity boundary conditions (mph) versus time (minutes) used in the model.

position of the vehicle is updated using the model described in Section 2.1. We assume that we obtain a GPS velocity and postmile reading every 2 seconds as long as vehicle is in the domain. The trajectories obtained using this method are shown in Figure 2. We add Gaussian noise to the GPS velocities with a standard deviation of 2 mph. We assume that there is no error in the postmile of the vehicle so that we know exactly in which cell of the computational domain the vehicle is at any given time. In the sequel, we call these noisy GPS velocities *measurements*.

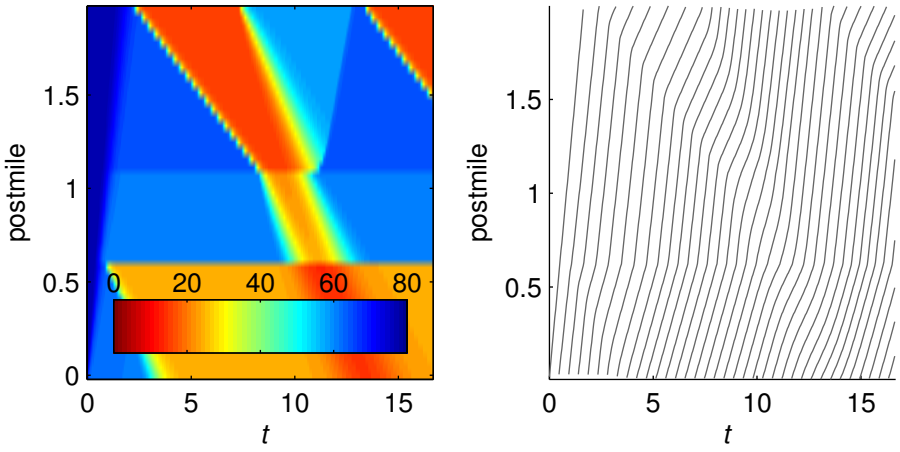

FIGURE 2. Left: True velocity field (mph) used to simulate vehicle trajectories. Right: Noise-free trajectories simulated using the true velocity field. Time is in minutes.

After data simulation we can proceed to the parameter estimation. We assume that the initial velocity and boundary conditions of the model are known during

the inverse modeling step. Clearly, estimation of the boundary conditions is also an important part of calibrating a flow model, but in this work we focus on the problem of estimating the flow model parameters, which is challenging even when the initial and boundary conditions are assumed to be known.

It is also important to notice that we use the same CTM-v model (including space discretization) and choice of flux function in the parameter estimation (inversion) as in the data simulation. This setting is referred to as an *inverse crime* [13], which represents a best case performance of the method.

The time step in the CTM-v model which is used in the parameter estimation is set to 2 seconds. This is done in order to avoid using the same time discretization as in the data generation and, thus, reducing the inverse crime.

Furthermore, we assume that the model predicted GPS velocity measurements $h(\theta)$ are obtained every 2 seconds as the average speed of the cell given by the CTM-v model. This means that the model predicting the GPS measurements during parameter estimation is different than the accurate model used to generate the synthetic data in Section 2.1. This setting also helps to reduce the inverse crime.

Despite the overly optimistic results which can be generated when inverse crimes are introduced into computational experiments, we purposefully use this setting to highlight the findings related to the efficient sampling of jam densities and the general solvability of the inverse problem.

4.1. **Sample generation.** In order for the MCMC methods to be effective, they should have a fast mixing time. The parameters that typically affect the mixing of the chain are the step length γ and the covariance of the measurement noise Γ_ε. Appropriate values for γ can be obtained, for example, using a trial and error method.

In [10], it is noted that posterior correlations also affect the mixing of the chain. Namely, strongly correlated model parameters in the posterior distribution can cause a lot of bad proposals thus preventing mixing of the chain.

Keeping our specific flow model application in mind, as we will demonstrate below, adjustments are needed in the selection of the model parameters which we estimate. Namely, one factor that affects the mixing in our application is the proposal related to the shock wave speed. This is because even small deviation from the true shock wave speed causes large discrepancy between the GPS measurements and the predicted measurements. This will further cause the value of the likelihood to be very small and thus the sample will most likely be rejected. In practice, the dependence of the solution on the ratio α makes the sampling difficult. Namely, unless the samples are drawn from the isoline $\rho_{\max,1} = \alpha\rho_{\max,2}$, the shock wave speed will be wrong and the samples get easily rejected.

In order to improve the mixing in Algorithm 1, we draw one value $\rho_{\max}$ and the ratios α for the remaining links instead of using random walk for all $\rho_{\max,i}$ $i \in \{1, 2, 3\}$. In our tests, this leads to significant improvement in the mixing of the chain.

5. **Results.** We present results for three different numerical experiments. In the first scenario, the parameters to be estimated include the jam densities on each link, and the resulting simulation suffers from the problem of poor mixing. In the second example, we show that improved mixing can be achieved by estimating ratios of jam densities instead of the jam densities directly. The third example shows that we can recover correct values for the model parameters although we do not use the same

(three–link) configuration in the parameter estimation as in the previous estimation cases and data simulation. Specifically, we will incorrectly split link l_3 into two links in our estimation algorithm, and we will recover the correct jam density ratio (one) between the two new links.

The common parameter values in both cases are: the measurement error covariance matrix $\Gamma_\varepsilon = (\sigma_\varepsilon)^2 \times I$, where $\sigma_\varepsilon = 3$ mph, I is the identity matrix, step length for maximum density $\gamma_{\rho_{\max}} = 10$, step length for ratios $\gamma_\alpha = 0.000075$, step length for shock wave speed $\gamma_{w_f} = 0.05$ and step length for maximum velocity $\gamma_{v_{\max}} = 0.1$.

We set the prior density for $\rho_{\max,2}$ to be the uniform density $\mathcal{U}[110 \times 4, 250 \times 5]$, which is constructed assuming a prior of $\rho_{\max} \in [110, 250]$ veh/mi per lane using jam density values observed in [8].

In order to ensure that we do not exceed the maximum velocity permitted by the CFL condition, 80 mph, we could set a prior for $v_{\max}$, for example $\mathcal{U}[70, 80]$. However, in our numerical experiments, we do not encounter problems with the sampling of $v_{\max}$ and, thus, omit the specification of the prior for $v_{\max}$ in this work. Also, for the shock wave speed, the construction of prior distribution is not needed in this study.

5.1. **Three jam densities.** The initial values for the Markov chain are: $v_{\max} = 75$ mph, $w_f = 17$ mph, $\rho_{\max,i} = 150 \times 5$ veh/mi, $i \in \{1, 2, 3\}$. The boundary conditions and the initial velocity are assumed to be known, and thus we use the same values as used in the measurement generation process. Furthermore, we assume the same configuration of the link lengths l_1, l_2, and l_3 as in the data simulation.

Figure 3 shows the results of the parameter estimation when all jam densities are drawn using random walk procedure. In other words the state vector θ is of the form

$$\theta = \begin{pmatrix} \rho_{\max,1} \\ \rho_{\max,2} \\ \rho_{\max,3} \\ v_{\max} \\ w_f \end{pmatrix}.$$

From the figure, two interesting conclusions can be made. First, we observe that the mixing is very poor. In this simulation, the acceptance ratio was close to zero after just 20,000 samples. In other words, the chain spends a lot of time stuck on a parameter realization, and new proposals are not accepted. Note, that although our parameter vector contained $\rho_{\max,1}$ and $\rho_{\max,3}$ explicitly, in Figure 3 we deliberately show the ratios of the jam densities.

The second observation is that the chain converges to correct values of jam density ratios, and parameters $v_{\max}$ and w_f quickly. However, the component of the chain corresponding to $\rho_{\max,2}$ shows very little movement, and the area near true value (gray line) is not sampled at all, despite the inverse crime setting.

The particular feature of our model, that the shock wave speed between links depends only on the ratio of jam densities, not the absolute values of them, suggested that the chain should not get stuck to some particular value of $\rho_{\max,2}$ but rather cover the posterior density more thoroughly.

5.2. **One jam density and two ratios.** In the second parameter estimation case, we estimate the maximum density $\rho_{\max,2}$, the ratios $\alpha_{1,2} = \rho_{\max,1}/\rho_{\max,2}$ and $\alpha_{3,2} = \rho_{\max,3}/\rho_{\max,2}$, the maximum velocity $v_{\max}$ and the shock wave speed w_f. Thus the

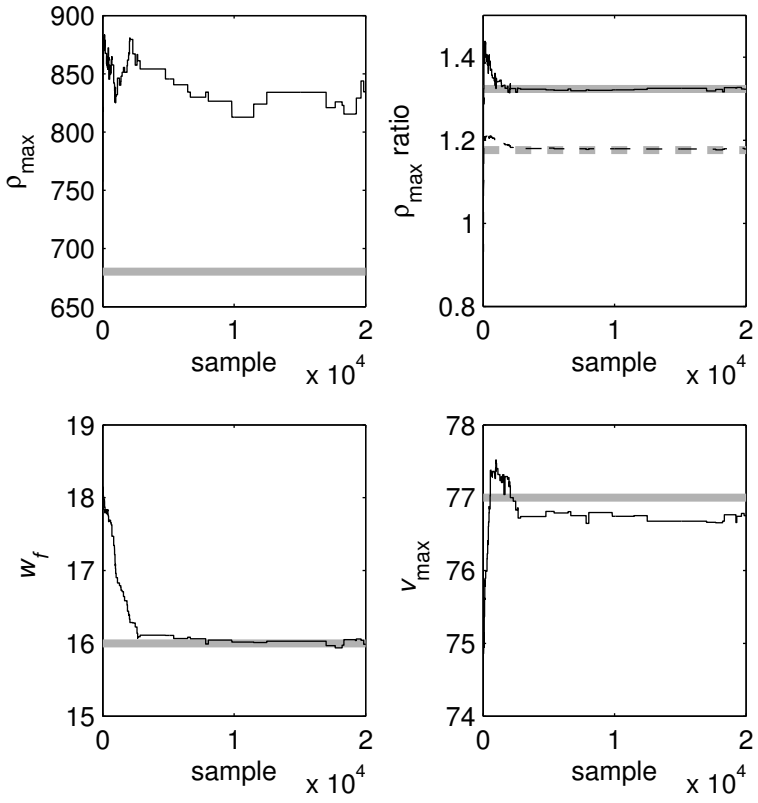

FIGURE 3. Example of poor mixing with Metropolis–Hastings algorithm. Upper left: Markov chain of estimated maximum density $\rho_{\max,2}$ (veh/mi) of the second link (black) and true value (gray). Upper right: estimated ratio (solid black) and true value (solid gray) for $\rho_{\max,1}/\rho_{\max,2}$. Estimated ratio (dashed black), true value (dashed gray) for $\rho_{\max,3}/\rho_{\max,2}$. Lower left: Markov chain for estimated shock wave speed (black) in mph and true value (gray). Lower right: Markov chain for estimated maximum velocity (black) in mph and true value (gray).

parameter vector θ can be written as

$$\theta = \begin{pmatrix} \alpha_{1,2} \\ \rho_{\max,2} \\ \alpha_{3,2} \\ v_{\max} \\ w_f \end{pmatrix}.$$

The initial values for the Markov chain are: $v_{\max} = 75$ mph, $w_f = 17$ mph, $\rho_{\max,2} = 150 \times 5$ veh/mi and ratios $\alpha_{1,2} = \alpha_{3,2} = 1$. The boundary conditions and the initial velocity are assumed to be known. We run the chain until we have 200,000 samples and we discard 100,000 first samples as the burn-in period. Furthermore, we assume the same configuration of the link lengths l_1, l_2, and l_3 as in the data simulation.

The results are shown in Figure 4. As can be seen, the parameter values are recovered with good accuracy, except for the $\rho_{\max,2}$. The high uncertainty regarding $\rho_{\max,2}$ reflected in our prior density is also visible in the diagnostics of the Markov chain. This suggests that when using GPS velocities as measurements, stronger priors may be needed in order to reduce the uncertainty in the estimates of the maximum densities.

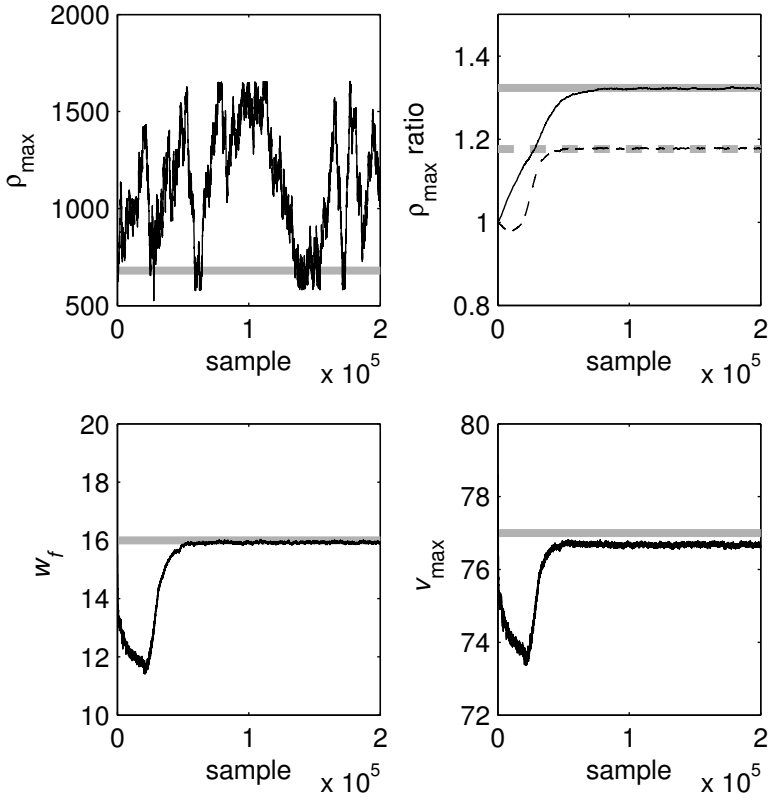

FIGURE 4. Upper left: Markov chain of estimated maximum density $\rho_{\max,2}$ (veh/mi) of the second link (black) and true value (gray). Upper right: estimated ratio (solid black) and true value (solid gray) for $\rho_{\max,1}/\rho_{\max,2}$. Estimated ratio (dashed black), true value (dashed gray) for $\rho_{\max,3}/\rho_{\max,2}$. Lower left: Markov chain for estimated shock wave speed (black) in mph and true value (gray). Lower right: Markov chain for estimated maximum velocity (black) in mph and true value (gray).

The computed mean values and standard deviations are summarized in Table 1 with the true values. Also, the absolute value of the error in the velocity field obtained using the mean values from Table 1 as flow model parameters is shown in Figure 5. Due to the accurate recovery of the ratios $\alpha_{1,2}$, $\alpha_{3,2}$, $v_{\max}$ and w_f, the overall discrepancy between the predicted velocity field and the true state is small, in spite of the poor estimate of the jam densities. However, it is worth noting that the use of a different time step in the synthetic data generation and parameter

estimation causes a noticeable error near the backward propagating shock wave boundary.

TABLE 1. Mean values, standard deviations and true values for the estimated parameters in the case of three links when ratios are sampled.

Parameter	True value	Mean value	Standard deviation
$\alpha_{1,2}$	1.32	1.32	0.001
$\rho_{\max,2}$	680	830	230
$\alpha_{3,2}$	1.18	1.18	0.0005
$v_{\max}$	77.0	76.7	0.03
w_f	16.0	15.9	0.02

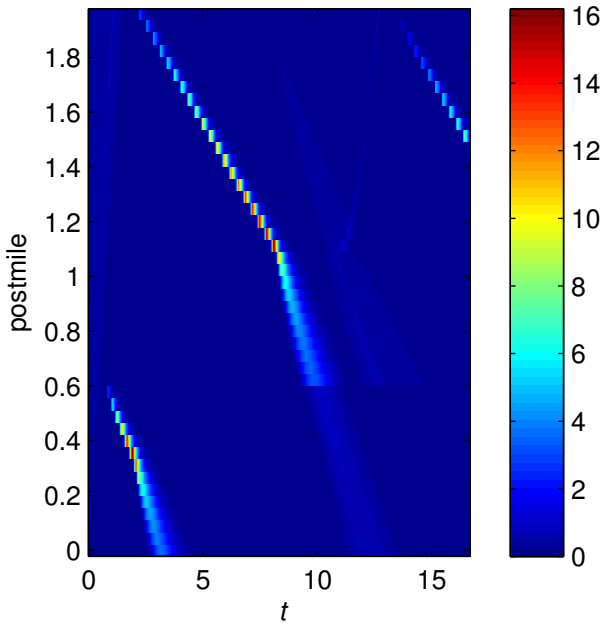

FIGURE 5. The absolute value of the error in the velocity field obtained using the posterior mean values as parameters. Time is in minutes. The unit of velocity error is mph.

5.3. **One jam density and three ratios.** In order to further test our method, we divide the domain into four links. We keep the links l_1 and l_2 the same as above but we split the link l_3 into two separate (but equal length) links l_3 and l_4. Hence, we do not change the length of the computational domain. In other words we assume a different link discretization than what is used in the generation of the true state. The lengths of the links are as follows: $l_1 = 0.62$ mi, $l_2 = 0.49$ mi, $l_3 = 0.445$, and $l_4 = 0.445$ mi. Since the true value of the $\rho_{\max,3} = \rho_{\max,4} = 5 \times 160$ veh/mi, we

should recover that the ratio $\alpha_{4,3} = \rho_{\max,4}/\rho_{\max,3}$ is 1. Thus the parameter vector θ can be written as

$$\theta = \begin{pmatrix} \alpha_{1,2} \\ \rho_{\max,2} \\ \alpha_{3,2} \\ \alpha_{4,3} \\ v_{\max} \\ w_f \end{pmatrix}.$$

The initial values for the Markov chain are: $v_{\max} = 75$ mph, $w_f = 17$ mph, $\rho_{\max,2} = 150 \times 5$ veh/mi and ratios $\alpha_{1,2} = \alpha_{3,2} = 1$ and $\alpha_{4,3} = 1.1$. The boundary conditions and the initial velocity are assumed to be known. We run the chain until we have 200,000 samples and we discard 100,000 first samples as the burn-in period.

From Figure 6 it is seen that the method recovers the value of ratio $\alpha_{4,3}$ accurately. This illustrates that it is possible to have redundancy when choosing the parameters for the estimation and the method can still recover the correct values for the ratios. As in the previous three–link-case, the value of $\rho_{\max,2}$ shows high uncertainty.

The high uncertainty in the values of the maximum density makes it difficult to interpret the lane configurations from GPS speed data through parameter estimation of a flow model. However, from a purely traffic velocity simulation point of view it is not necessary to know the true values in order to obtain good forward simulation accuracy. Instead, knowing the ratios between the maximum densities plays a crucial role.

The computed mean values and standard deviations are summarized in Table 2 with the true values.

TABLE 2. Mean values, standard deviations and true values for the estimated parameters in the case of four links when ratios are sampled.

Parameter	True value	Mean value	Standard deviation
$\alpha_{1,2}$	1.32	1.32	0.001
$\rho_{\max,2}$	680	750	230
$\alpha_{3,2}$	1.18	1.17	0.002
$\alpha_{4,3}$	1.00	1.00	0.001
$v_{\max}$	77.0	76.7	0.04
w_f	16.0	15.9	0.04

6. **Conclusions and future work.** In this paper we presented a calibration method for fundamental diagram parameters. The work serves as a first step toward a larger goal which is to be able to utilize GPS data in the calibration of flow models operating on a realistic road network. Our work focused on main line parameters. Our proposed Markov Chain Monte Carlo recovers important fundamental diagram parameters and enables accurate forward simulations of the traffic flow model.

The estimation of the true value of jam density remains a challenging task with GPS data alone. Without strong priors or additional data sources there is lot of uncertainty with respect to the physical value of the jam density. However, the value of the jam density does not play a crucial role in the forward simulation accuracy. Instead we showed that estimating the ratios of the jam densities between

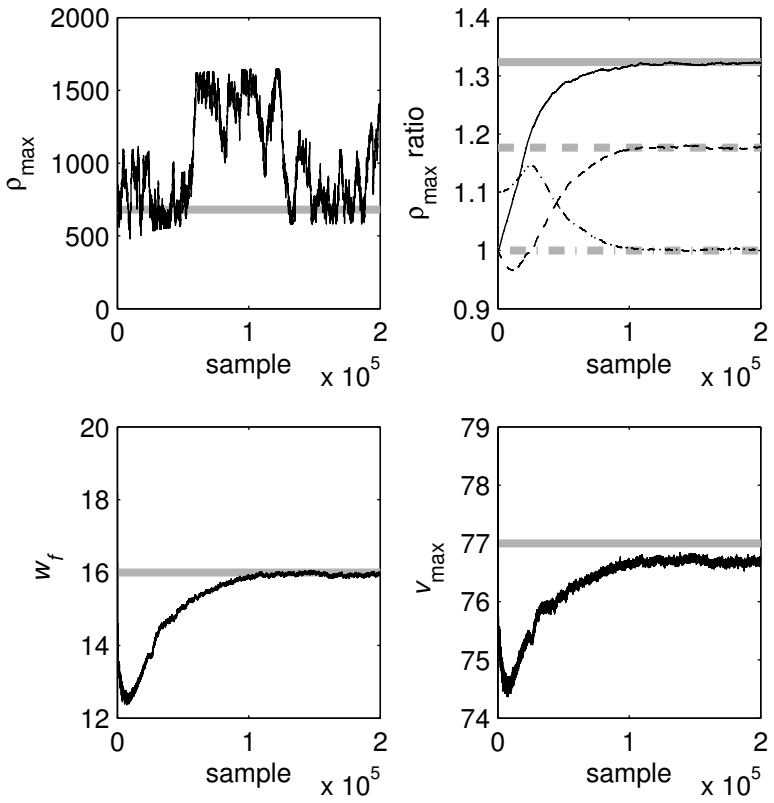

FIGURE 6. Upper left: Markov chain of estimated maximum density (veh/mi) of the second link (black) and true value (gray). Upper right: estimated ratio (solid black) and true value (solid gray) for $\rho_{\max,1}/\rho_{\max,2}$. Estimated ratio (dashed black), true value (dashed gray) for $\rho_{\max,3}/\rho_{\max,2}$. Estimated ratio (dash-dotted black), true value (dash-dotted gray) for $\rho_{\max,4}/\rho_{\max,3}$ Lower left: Markov chain for estimated shock wave speed (black) in mph and true value (gray). Lower right: Markov chain for estimated maximum velocity (black) in mph and true value (gray).

the consecutive road segments is sufficient for speed evolution prediction. Thus, the sensitivity of the flow model to errors in the number of lanes in a map database stems from the fact that these errors create incorrect estimates of the ratios of the jam densities between links.

Testing the method with experimental data is part of our future work. To complete the calibration in an network scale, our future work also includes the calibration of the parameters related to merging and diverging traffic. These parameters include quantities such as on-ramp flows and flow splitting ratios in the junctions.

Acknowledgments. The authors are grateful for the helpful suggestions by the anonymous referees.

REFERENCES

[1] T. Bellemans, B. D. Schutter and B. D. Moor, *Model predictive control with repeated model fitting for ramp metering*, in "Proceedings of the IEEE 5th Conference on Intelligent Transportation Systems," (2002), 236–241.

[2] T. Bellemans, B. D. Schutter, G. Wets and B. D. Moor, *Model predictive control for ramp metering combined with extended Kalman filter-based traffic state estimation*, in "Proceedings of IEEE Intelligent Transportation Systems Conference," (2006), 406–411.

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

[4] G. Bretti and B. Piccoli, *A tracking algorithm for car paths on road networks*, SIAM Journal on Applied Dynamical Systems, **7** (2008), 510–531.

[5] R. M. Colombo and A. Marson, *A Hölder continuous ode related to traffic flow*, Proceedings of the Royal Society of Edinburgh: Section A Mathematics, **133** (2003), 759–772.

[6] M. Cremer and M. Papageorgiou, *Parameter identification for a traffic flow model*, Automatica J. IFAC, **17** (1981), 837–843.

[7] E. Cristiani, C. de Fabritiis and B. Piccoli, *A fluid dynamic approach for traffic forecast from mobile sensor data*, Communications in Applied and Industrial Mathematics, **1** (2010), 54–71.

[8] G. Dervisoglu, G. Gomes, J. Kwon, R. Horowitz and P. Varaiya, *Automatic calibration of the fundamental diagram and empirical observations on capacity*, in "Proceedings of the 88th Annual Meeting," Washington, D.C., 2009, Transportation Research Board.

[9] M. Garavello and B. Piccoli, "Traffic Flow on Networks," Conservation laws models. AIMS Series on Applied Mathematics, 1. American Institute of Mathematical Sciences (AIMS), Springfield, MO, 2006. xvi+243 pp.

[10] W. Gilks, S. Richardson and D. Spegelhalter, "Markov Chain Monte Carlo in Practice," Interdisciplinary Statistics. Chapman & Hall, London, 1996.

[11] S. Godunov, *A difference method for the numerical calculation of discontinuous solutions of hydrodynamic equations*, (Russian) Mat. Sb. (N. S.), **47** (1959), 271–306.

[12] A. Hegyi, D. Girimonte, R. Babŭska and B. D. Schutter, *A comparison of filter configurations for freeway traffic state estimation*, in "Proceedings of the 2006 IEEE Intelligent Transportation Systems Conference," Toronto, Canada, 2006, ITSC, 1029–1034.

[13] J. Kaipio and E. Somersalo, "Statistical and Computational Inverse Problems," Springer, 2005.

[14] J. Lebacque, *The Godunov scheme and what it means for first order traffic flow models*, in "13th International Symposium on Transportation and Traffic Theory," (1996), 647–677.

[15] R. J. LeVeque, "Numerical Methods for Conservation Laws," Second edition. Lectures in Mathematics ETH Zürich. Birkhäuser Verlag, Basel, 1992. x+214 pp.

[16] M. Lighthill and G. Whitham, *On kinematic waves. II. A theory of traffic flow on long crowded roads*, Proc. Roy. Soc. London. Ser. A. **229** (1955), 317–345.

[17] J. V. Lint, S. Hoogendoorn and A. Hegyi, *Dual EKF state and parameter estimation in multi-class first-order traffic flow models*, in "Proceedings of the 17th World Congress," Seoul, Korea, (2008), The International Federation of Automatic Control.

[18] X.-Y. Lu, P. Varaiya and R. Horowitz, *Fundamental diagram modeling and analysis based on NGSIM data*, in "12th IFAC Symposium on Control in Transportation Systems," (2009).

[19] L. Mihaylova, R. Boel and A. Hegyi, *Freeway traffic estimation within particle filtering framework*, Automatica J. IFAC, **43** (2007), 290–300.

[20] L. Munoz, X. Sun, D. Sun, G. Gomez and R. Horowitz, *Methodological calibration of the cell transmission model*, in "Proceedings of the American Control Conference," **1**, 2004, 798–803.

[21] A. Muralidharan, G. Dervisoglu and R. Horowitz, *Probabilistic graphical models of fundamental diagram parameters for freeway traffic simulations*, in "Proceedings of the 90th Annual Meeting," Washington, D.C., (2011), Transportation Research Board.

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

[23] S. Smulders, *Control of freeway traffic flow by variable speed signs*, Transportation Research Part B: Methodological, **24** (1990), 111–132.

[24] Transportation Research Board, "HCM 2010: Highway Capacity Manual," (2010).

[25] `http://traffic.berkeley.edu`.

[26] Y. Wang, M. Papageorgiou and A. Messmer, *RENAISSANCE – A unified macroscopic model-based approach to real-time freeway network traffic surveillance*, Transportation Research Part C: Emerging Technologies, **14** (2006), 190–212.

[27] Y. Wang, M. Papageorgiou and A. Messmer, *Real-time freeway traffic state estimation based on extended kalman filter: A case study*, Transportation Science, **41** (2007), 167–181.
[28] D. Work, S. Blandin, O.-P. Tossavainen, B. Piccoli and A. Bayen, *A traffic model for velocity data assimilation*, Appl. Math. Res. Express. AMRX, **2010** (2010), 1–35.
[29] J. Yan, *Parameter identification of freeway traffic flow model and adaptive ramp metering*, in "2009 Second International Symposium on Electronic Commerce and Security," (2009), 235–238.

Appendix A. **Numerical scheme for simulating vehicle trajectories.** We explicate a numerical scheme for integrating vehicle trajectories for linear networks with the Smulders velocity function. Our implementation is inspired by the algorithm proposed in for the Greenshields velocity function in [4].

We define the absolute position of the vehicle at discrete time n as χ^n, and assume $i\Delta x \leq \chi^n < (i+1)\Delta x$. The vehicle position update scheme is of the form

$$\chi^{n+1} = \chi^n + d\left(v_i^n, v_{i+1}^n\right), \tag{8}$$

where v_i^n and v_{i+1}^n are the velocities computed by (3). The function d determines the distance traveled by the vehicle in the time step ΔT, and is computed by integrating the vehicle through the speed field obtained by solving a Riemann problem between cells i (running from $x \in [i\Delta x, (i+1)\Delta x)$) and $i+1$, (running from $x \in [(i+1)\,\Delta x, (i+2)\Delta x)$) (i.e. at the cell boundary located at $x = (i+1)\,\Delta x$). Because the vehicle speeds are always at least as large as the characteristic speed for a given density, we only need to concern ourselves with interactions between the vehicle and waves emanating from the cell boundary in front of the vehicle. Moreover, if the time step ΔT is chosen to satisfy $\frac{\Delta T}{\Delta x} v_{\max} \leq \frac{1}{2}$, then it is not necessary to consider interactions of waves emanating from $x = (i+1)\,\Delta x$ with waves from $x = (i+2)\,\Delta x$.

A.1. **Riemann problem and solution on a linear network.** This work considers the case of a linear network (one incoming link and one outgoing link at each junction), where the parameters θ of the flux function $q_\theta(\rho) = \rho V_\theta(\rho)$ are allowed to change between links. Therefore, the Riemann problem between the left link $x \in (-\infty, (i+1)\,\Delta x)$ and right link $x \in (-\infty, (i+1)\,\Delta x)$ can be written as:

$$\begin{cases} \frac{\partial \rho}{\partial t} + \frac{\partial q_{\theta_l}(\rho)}{\partial x} = 0 & (x,t) \in (-\infty, (i+1)\,\Delta x) \times (0,T) \\ \frac{\partial \rho}{\partial t} + \frac{\partial q_{\theta_r}(\rho)}{\partial x} = 0 & (x,t) \in ((i+1)\,\Delta x, \infty) \times (0,T), \end{cases} \tag{9}$$

with initial data given by:

$$\rho_0(x) = \begin{cases} \rho_l & \text{if } x < (i+1)\,\Delta x \\ \rho_r & \text{if } x > (i+1)\,\Delta x, \end{cases} \tag{10}$$

and where ρ_l and ρ_r are the left and right initial densities. An admissible solution to (9) and (10) can be obtained with the introduction of *new* or *internal* states $\tilde{\rho}_l$ and $\tilde{\rho}_r$ given by:

$$\tilde{\rho}_l \in \begin{cases} \{\rho_l\} \cup (\tau_l(\rho_l), \rho_{\max,l}] & \text{if } \rho_l \leq \rho_{c,l} \\ [\rho_{c,l}, \rho_{\max,l}] & \text{if } \rho_l \geq \rho_{c,l} \end{cases} \tag{11}$$

and

$$\tilde{\rho}_r \in \begin{cases} [0, \rho_{c,r}] & \text{if } \rho_r \leq \rho_{c,r} \\ \{\rho_r\} \cup [0, \tau_r(\rho_r)) & \text{if } \rho_r \geq \rho_{c,r}, \end{cases} \tag{12}$$

where the subscript l and r on the parameters are used to distinguish the change in the parameters on the left and right side of $(i+1)\,\Delta x$. The function τ is defined such that $q\left(\tau\left(\rho\right)\right) = q\left(\rho\right)$, and $\tau\left(\rho\right) \neq \rho$ for all $\rho \neq \rho_c$.

The maximal flow across the junction at $(i+1)\,\Delta x$, denoted $\tilde{q}$, can be computed as

$$\tilde{q} = \min\left\{S_{\theta_l}\left(\rho_l\right), R_{\theta_r}\left(\rho_r\right)\right\},$$

where S_{θ_l} and R_{θ_r} are the sending and receiving functions:

$$S_{\theta_l}\left(\rho\right) = \begin{cases} q_{\theta_l}\left(\rho\right) & \text{if } \rho \leq \rho_{c,l} \\ q_{\max,l} & \text{if } \rho \geq \rho_{c,l} \end{cases}$$

and

$$R_{\theta_r}\left(\rho\right) = \begin{cases} q_{\max,r} & \text{if } \rho \leq \rho_{c,r} \\ q_{\theta_r}\left(\rho\right) & \text{if } \rho \geq \rho_{c,r}. \end{cases}$$

The unique values of the internal states can be computed by inverting the relations

$$q_{\theta_l}\left(\tilde{\rho}_l\right) = \tilde{q}, \quad q_{\theta_r}\left(\tilde{\rho}_r\right) = \tilde{q}$$

on the sets given by (11) and (12). When internal velocity states are desired, they can be obtained according to $\tilde{v}_l = V_{\theta_l}\left(\tilde{\rho}_l\right)$ and $\tilde{v}_r = V_{\theta_r}\left(\tilde{\rho}_r\right)$.

With the internal states defined, the solution to (9) and (10) is separated into two cases, depending the type of connection between the states ρ_r and $\tilde{\rho}_r$. If $\tilde{\rho}_r < \rho_r$, the states are connected by a shock, while if $\tilde{\rho}_r > \rho_r$, the states are separated by an expansion fan. By the structure of the Smulders flux function, all waves in congestion travel at the same speed w_f, which simplifies the connections between ρ_l and $\tilde{\rho}_l$. The solution is given as follows.

- **Case 1.** If $\tilde{\rho}_r \leq \rho_r$,

$$\rho(x,t) = \begin{cases} \rho_l & \text{if } x - (i+1)\,\Delta x < s_1 t \\ \tilde{\rho}_l & \text{if } s_1 t < x - (i+1)\,\Delta x < 0 \\ \tilde{\rho}_r & \text{if } 0 < x - (i+1)\,\Delta x < s_2 t \\ \rho_r & \text{if } s_2 t < x - (i+1)\,\Delta x, \end{cases} \tag{13}$$

  where s_1 is the speed of the wave connecting ρ_l and $\tilde{\rho}_l$ when the wave exists, and zero otherwise:

$$s_1 = \begin{cases} \frac{q_{\theta_l}(\rho_l) - q_{\theta_l}(\tilde{\rho}_l)}{\rho_l - \tilde{\rho}_l} & \text{if } \rho_l \neq \tilde{\rho}_l \\ 0 & \text{if } \rho_l = \tilde{\rho}_l. \end{cases} \tag{14}$$

  The wave speed s_2 is defined similarly in terms of ρ_r and $\tilde{\rho}_r$:

$$s_2 = \begin{cases} \frac{q_{\theta_r}(\tilde{\rho}_r) - q_{\theta_r}(\rho_r)}{\tilde{\rho}_r - \rho_r} & \text{if } \rho_r \neq \tilde{\rho}_r \\ 0 & \text{if } \rho_r = \tilde{\rho}_r. \end{cases}$$

- **Case 2.** If $\tilde{\rho}_r \geq \rho_r$,

$$\rho(x,t) = \begin{cases} \rho_l & \text{if } x - (i+1)\,\Delta x < s_1 t \\ \tilde{\rho}_l & \text{if } s_1 t < x - (i+1)\,\Delta x < 0 \\ \tilde{\rho}_r & \text{if } 0 < x - (i+1)\,\Delta x < s_3 t \\ \frac{1}{2}\rho_{\max,r}\left(1 - \frac{x-(i+1)\Delta x}{t v_{\max,r}}\right) & \text{if } s_3 t < x - (i+1)\,\Delta x < s_4 t \\ \rho_r & \text{if } s_4 t < x - (i+1)\,\Delta x, \end{cases} \tag{15}$$

where s_1 is given by (14). The terms s_3 and s_4 are the speeds of the waves separating the states $\tilde{\rho}_r$ and ρ_r with the expansion fan when the wave exists, and zero otherwise. Recalling the characteristic speeds for the Smulders flux function when $\rho < \rho_c$ are given by $v_{\max}\left(1 - \frac{2\rho}{\rho_{\max}}\right)$, s_3 and s_4 can be computed as:

$$s_3 = \begin{cases} v_{\max,r}\left(1 - \frac{2\tilde{\rho}_r}{\rho_{\max,r}}\right) & \text{if } \tilde{\rho}_r \neq \rho_r \\ 0 & \text{if } \tilde{\rho}_r = \rho_r \end{cases}$$

and

$$s_4 = \begin{cases} v_{\max,r}\left(1 - \frac{2\rho_r}{\rho_{\max,r}}\right) & \text{if } \tilde{\rho}_r \neq \rho_r \\ 0 & \text{if } \tilde{\rho}_r = \rho_r. \end{cases}$$

A.2. **Vehicle trajectory evolution equations.** The position of a vehicle $\chi(t)$ traveling at the speed $V_\theta(\rho(x,t))$ with $\rho(x,t)$ defined by (13) and (15), is governed by the following ordinary differential equation:

- **Case 1.** If $\tilde{v}_r \geq v_r$,

$$\dot{\chi} = \begin{cases} v_l & \text{if } \chi - (i+1)\,\Delta x < +s_1 t \\ \tilde{v}_l & \text{if } s_1 t < \chi - (i+1)\,\Delta x < 0 \\ \tilde{v}_r & \text{if } 0 < \chi - (i+1)\,\Delta x < s_2 t \\ v_r & \text{if } s_2 t < \chi - (i+1)\,\Delta x, \end{cases} \tag{16}$$

with s_1 and s_2 defined in terms of velocities as:

$$s_1 = \begin{cases} \frac{V_{\theta_l}^{-1}(v_l)v_l - V_{\theta_l}^{-1}(\tilde{v}_l)\tilde{v}_l}{V_{\theta_l}^{-1}(v_l) - V_{\theta_l}^{-1}(\tilde{v}_l)} & \text{if } v_l \neq \tilde{v}_l \\ 0 & \text{if } v_l = \tilde{v}_l \end{cases}$$

and

$$s_2 = \begin{cases} \frac{V_{\theta_r}^{-1}(\tilde{v}_r)\tilde{v}_r - V_{\theta_r}^{-1}(v_r)v_r}{V_{\theta_r}^{-1}(\tilde{v}_r) - V_{\theta_r}^{-1}(v_r)} & \text{if } v_r \neq \tilde{v_r} \\ 0 & \text{if } v_r = \tilde{v_r}. \end{cases}$$

- **Case 2.** If $\tilde{v}_r \leq v_r$,

$$\dot{\chi} = \begin{cases} v_l & \text{if } \chi - (i+1)\,\Delta x < s_1 t \\ \tilde{v}_l & \text{if } s_1 t < \chi - (i+1)\,\Delta x < 0 \\ \tilde{v}_r & \text{if } 0 < \chi - (i+1)\,\Delta x < s_3 t \\ \frac{1}{2}\left(v_{\max,r} + \frac{\chi - (i+1)\Delta x}{t}\right) & \text{if } s_3 t < \chi - (i+1)\,\Delta x < s_4 t \\ v_r & \text{if } s_4 t < \chi - (i+1)\,\Delta x, \end{cases} \tag{17}$$

with s_3 and s_4 defined in terms of velocities as:

$$s_3 = \begin{cases} 2\tilde{v}_r - v_{\max,r} & \text{if } \tilde{v_r} \neq v_r \\ 0 & \text{if } \tilde{v_r} = v_r \end{cases}$$

and

$$s_4 = \begin{cases} 2v_r - v_{\max,r} & \text{if } \tilde{v_r} \neq v_r \\ 0 & \text{if } \tilde{v_r} = v_r. \end{cases}$$

By integrating the ODE (16) or (17) forward in time with the initial condition $\chi(0) = 0$, the distance d in (8) is given by $d(v_l, v_r) = \chi(\Delta T)$. To compute the solution, the times and positions of the vehicle interactions with the waves in (16) and (17) need to be computed. We detail these calculations next.

A.3. **Computation of vehicle–wave interactions.** Let us first consider case 1 given by (16). There are three possible interactions a vehicle located at $\chi^n < (i+1)\Delta x$ could have with different waves in case 1. The first interaction occurs with the wave with speed s_1 at the point (x_a, t_a):

$$x_a = \chi^n + v_l t_a = (i+1)\,\Delta x + s_1 t_a. \tag{18}$$

The interaction time t_a is given by:

$$t_a = \frac{(i+1)\,\Delta x - \chi^n}{v_l - s_1}. \tag{19}$$

The second interaction occurs with the boundary, at the point given by:

$$x_b = x_a + \tilde{v}_l\,(t_b - t_a) = (i+1)\,\Delta x. \tag{20}$$

The interaction time with the boundary is given by:

$$t_b = \frac{(i+1)\,\Delta x - x_a + \tilde{v}_l t_a}{\tilde{v}_l}. \tag{21}$$

The third interaction occurs with the wave s_2, at the point given by (x_c, t_c):

$$x_c = x_b + \tilde{v}_r\,(t_c - t_b) = (i+1)\,\Delta x + s_2 t_c. \tag{22}$$

The interaction time t_c is given by:

$$t_c = \frac{\tilde{v}_r t_b}{\tilde{v}_r - s_2}. \tag{23}$$

Let us now consider case 2 given by (17). There are four possible interactions a vehicle could have with different waves in case 2. The first and second interactions occur with a wave traveling at speed s_1 at (x_a, t_a), and the cell boundary at (x_b, t_b), which are computed by (18), (19), (20), and (21). The third interaction occurs with the left side of the expansion fan at (x_c, t_c), given by:

$$x_c = x_b + \tilde{v}_r\,(t_c - t_b) = (i+1)\,\Delta x + s_3 t_c \tag{24}$$

and

$$t_c = \frac{\tilde{v}_r t_b}{\tilde{v}_r - s_3}. \tag{25}$$

The fourth interaction occurs when the vehicle exits the expansion fan at (x_d, t_d), given by:

$$x_d = x\,(t_d) = (i+1)\,\Delta x + s_4 t_d, \tag{26}$$

where

$$x(t) = c_1\sqrt{t} + t v_{\max,r} + (i+1)\Delta x \tag{27}$$

and c_1 is a constant of integration. Applying $x\,(t_c) = x_c$ gives:

$$c_1 = \frac{x_c - t_c v_{\max,r} - (i+1)\,\Delta x}{\sqrt{t_c}}. \tag{28}$$

Thus, the time the vehicle exits the expansion fan can be obtained by solving

$$c_1\sqrt{t_d} + t_d v_{\max,r} = s_4 t_d \tag{29}$$

for t_d.

Algorithm 2 Vehicle position update for Smulders velocity function

Find the cell index i such that $i\Delta x \leq \chi^n < (i+1)\Delta x$
Set $v_l = v_i^n$ and $v_r = v_{i+1}^n$
Calculate internal states $\tilde{v}_l$ and $\tilde{v}_r$
Calculate x_a and t_a according to (18) and (19)
if $t_a \geq \Delta T$ **then**
  $\chi^{n+1} = \chi^n + v_l \Delta T$
else
  Calculate x_b and t_b according to (20) and (21)
  if $t_b \geq \Delta T$ **then**
    $\chi^{n+1} = x_a + \tilde{v}_l\,(\Delta T - t_a)$
  else
    if $\tilde{v}_r > v_r$ (case 1) **then**
      Calculate x_c and t_c according to (22) and (23)
      if $t_c \geq \Delta T$ **then**
        $\chi^{n+1} = x_b + \tilde{v}_r\,(\Delta T - t_b)$
      else
        $\chi^{n+1} = x_c + v_r\,(\Delta T - t_c)$
      end if
    else (case 2)
      Calculate x_c and t_c according to (24) and (25)
      if $t_c \geq \Delta T$ **then**
        $\chi^{n+1} = x_b + \tilde{v}_r\,(\Delta T - t_b)$
      else
        Solve (29) for t_d
        if $t_d \geq \Delta T$ **then**
          $\chi^{n+1} = x(\Delta T)$
        else
          $\chi^{n+1} = x\,(t_d) + v_r\,(\Delta T - t_d)$
        end if
      end if
    end if
  end if
end if

A.4. **Summary of the algorithm.** The algorithm for updating the vehicle positions is summarized in Algorithm 2. After determining the current cell of the vehicle, and computing the internal states $\tilde{v}_l$ and $\tilde{v}_r$, it remains to compute the distance traveled by the vehicle during the interval ΔT. This is achieved by integrating (16) or (17) from 0 to ΔT, which requires computation of the interaction times with the waves described in A.3.

Received October 2012; revised June 2013.

E-mail address: olli-pekka.tossavainen@microsoft.com
E-mail address: dbwork@illinois.edu