Estimating traffic conditions from smart work zone systems

Yanning Li*, Juan Carlos Martinez Mori*, and Daniel B. Work†

Abstract

This article evaluates the effectiveness of sensor network systems for work zone traffic estimation. The comparative analysis is performed on a work zone modeled in micro simulation and calibrated with field data from an Illinois work zone. Realistic error models are used to generate noisy measurements corresponding to Doppler radar sensors, *remote traffic microwave sensors* (RTMSs), and low energy radars. The velocity, queue length, and travel time are estimated with three algorithms based on *i*) interpolation, *ii*) spatio-temporal smoothing, and a *iii*) flow model based Kalman filter. A total of 396 sensor and algorithm configurations are evaluated and the accuracy of the resulting traffic estimates are compared to the true traffic state from the micro simulation. The nonlinear Kalman filter provides up to 30% error reduction over other velocity estimators when the RTMS sensor spacing exceeds two miles, and generally offers the best performance for queue and travel time estimation.

1 Introduction

Smart work zones (SWZs) aim to improve work zone safety and mobility through the integration of traffic sensors, estimation algorithms, and traffic management strategies. SWZ sensor networks have been deployed in a variety of applications such as the provision of realtime traveler information [1, 2, 3] including back of queue warnings [4, 5], and traffic management [6, 7, 8]. Many qualitative and quantitative benefits of smart work zones are reported from field deployments, such as a reduction in aggressive maneuvers and crashes [9, 10, 2], smoothed merging activities [11, 12], reduced speeding [13, 14], increased throughput [11], and reduced delay [15, 16]. For many SWZ applications, the effectiveness of the deployment relies on accurate and reliable traffic estimates. For example, the estimated traffic states used to produce safety critical messages on *portable*

*Department of Civil and Environmental Engineering, University of Illinois at Urbana-Champaign, 205 N. Mathews Ave., Urbana, IL 61801.

†Department of Civil and Environmental Engineering and Coordinated Science Laboratory, University of Illinois at Urbana-Champaign, 205 N. Mathews Ave., Urbana, IL 61801. Email: dbwork@illinois.edu

changeable message signs (PCMSs) must be accurate and credible to be effective [17].

Given the increasing number smart work zone deployments, cross-studies have been performed to summarize the lessons learned and the benefits of each smart work zone [10, 18, 9, 19]. Recently, the *Federal Highway Administration* (FHWA) published a work zone implementation guideline [20] to determine the feasibility and design of work zone ITS for a given application. The guideline also reports that trade offs typically exist between the number and type of work zone components, but it lacks a quantitative assessment of the trade offs. A main reason for the knowledge gap is due to the difficulty of collecting detailed performance data from a large number of configurations in an active work zone.

To address this gap, the main contribution of this article is the quantitative assessment of the influence of the sensor type and spacing, and the traffic estimation algorithms used to monitor the traffic state in a work zone. The evaluation framework used to identify the importance of each factor is illustrated in Figure 1. A microscopic traffic simulation software AIMSUN is calibrated with field data to model traffic dynamics around a work zone on I80 in Illinois. AIMSUN generates trajectory data for each vehicle traveling through the work zone, and is used to define the true state to be estimated by the traffic estimation algorithms. A total of 396 sensor network configurations are evaluated to measure the importance of the number and spacing of sensors, the types of sensors, the accuracy of individual sensors, and the estimation algorithms.

Three algorithms are implemented to estimate the traffic state. The algorithms are representative of the *i*) spatial interpolation approaches often used in practice by state Departments of Transportation, *ii*) spatial-temporal filtering algorithms that can incorporate the temporal dynamics of traffic, and *iii*) state-of-art nonlinear Kalman filtering methods conventionally applied on highways outside of work zone environments [21, 22]. The algorithms directly estimate the traffic velocity, from which the length of the queue and the travel time can be computed.

The main findings of the analysis are the following. Advanced algorithms such as nonlinear Kalman filtering can be used to significantly reduce the velocity, length of queue, and travel time error when flow is accurately measured by the sensor. It is also found that all algorithms are able to perform well with existing sensor technologies, and little additional benefit can expected from improvements to the quality of the individual sensors. Finally, it is noted that all classes of algorithms perform poorly on travel time estimation during the rapidly changing traffic conditions, which motivates the need for predictive analytics for smart work zone monitoring systems.

The remainder of this article is organized as follows. Section 2 presents the implementation of three traffic estimation algorithms used to estimate critical traffic quantities in work zones. Section 3 details the methods used to translate the micro simulation trajectory data into sensor measurements containing realistic measurement errors. Section 4 is devoted to the comparative analysis across sensor network configurations and algorithms. Finally, Section 5 summarizes the findings from the comparative analysis.

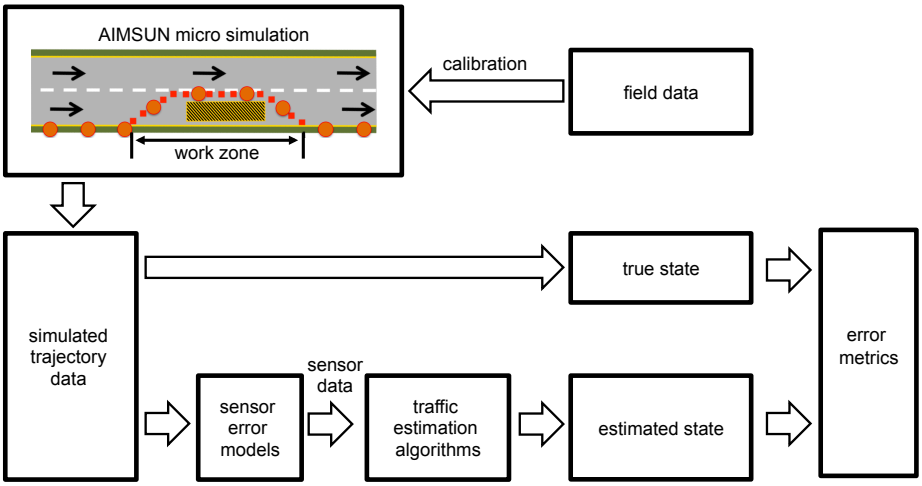


Figure 1: The framework for the evaluation of various sensor networks and algorithms.

2 Traffic State Estimation Algorithms

Consider a segment of roadway of length L containing a work zone, observed over a period of time T. The velocity along the roadway is denoted $v(t, x)$, where $t \in [0, T]$ and $x \in [0, L]$. If a queue develops on the segment due to the bottleneck created by the work zone, the length of the queue is denoted $l(t) \geq 0$. Finally, the travel time of a vehicle entering the roadway at time t traveling the length of the road segment L is denoted $\tau(t)$. In this work the velocity along the roadway is directly estimated, which is then used to estimate the travel time and the length of the queue when it occurs. The evaluated traffic estimation algorithms are described next.

2.1 Spatial Interpolation

A common class of algorithms infer the traffic conditions along the roadway by spatially interpolating the measurement data obtained from sensors. Such strategies include *constant interpolation* [23], *min interpolation* [24] adopted by the Texas Department of Transportation, *averaging interpolation* [25], the *mid-point algorithm* [26] adopted by the Illinois and Wisconsin Departments of Transportation, and the *three segment algorithm* [27]. More sophisticated interpolation strategies, such as *linear interpolation* [28] and *quadratic interpolation* [29] have also been proposed.

Considering the performance and ease of implementation, a linear interpolation is selected as a representative interpolation algorithm for further evaluation.

To estimate the velocity $v(t,x)$ between a pair of sensors located at x_1 and x_2 ($> x_1$) with corresponding measured velocities $\tilde{v}_1$ and $\tilde{v}_2$, the estimated velocity $\hat{v}(x,t)$ for $x \in [x_1, x_2]$ computed as:

$$\hat{v}(t,x) = \frac{x_2 - x}{x_2 - x_1}\tilde{v}_1(t) + \left(1 - \frac{x_2 - x}{x_2 - x_1}\right)\tilde{v}_2(t). \tag{1}$$

The same process is repeated for each pair of adjacent sensors to obtain the estimated velocity field throughout the spatial domain.

2.2 Spatio-temporal Smoothing

A criticism of the interpolation algorithms is that they do not directly account for the spatio-temporal dynamics of traffic. Approaches to exploit the structure of traffic include two-dimensional interpolation [30], trajectory reconstruction based methods [31], and kernel smoothing techniques [32, 33, 26, 34].

The main idea of the spatio-temporal estimator [32] considered in this work is to estimate the traffic using a weighted spatio-temporal average of the available measurements. Van Lint [26] showed that estimating the *pace* of traffic, $p(t,x) = 1/v(t,x)$, rather than the velocity is necessary to avoid the structural bias in the travel times constructed from the velocity field estimated by the spatio-temporal method. The pace estimate $\hat{p}(t,x)$ at time t and space x is:

$$\hat{p}(t,x) = \gamma(t,x)p_{\text{cong}}(t,x) + (1 - \gamma(t,x))p_{\text{free}}(t,x), \tag{2}$$

which is a convex combination of two pace estimates p_{free} and p_{cong} weighted by $\gamma(t,x)$. The freeflow and congested pace estimates are computed by averaging a set of measurements $\mathcal{M}(t,x)$ in the neighborhood of (t,x) along the freeflow and congested wave speeds of traffic flow respectively, given by kinematic wave theory [35, 36]. In order to generate realtime estimates required in work zone applications, the measurement set $\mathcal{M}(t,x)$ includes only measurements obtained up to time t.

Let $\tilde{p}_m = 1/\tilde{v}_m$ denote the corresponding pace of a velocity measurement obtained at (t_m, x_m). The congested pace p_{cong} is a weighted average of the measurements $\mathcal{M}(t,x)$:

$$p_{\text{cong}}(t,x) = \frac{1}{\sum\limits_{m \in M(t,x)} \phi_{\text{cong}}\left(t_m - t, x_m - x\right)} \sum_{m \in M(t,x)} \tilde{p}_m \phi_{\text{cong}}\left(t_m - t, x_m - x\right). \tag{3}$$

The weight of each measurement is computed with a decaying exponential aligned with maximum congested wave speed w:

$$\phi_{\text{cong}}(t,x) = \exp\left(-\frac{|t - x/w|}{\zeta} - \frac{|x|}{\kappa}\right), \tag{4}$$

where κ and ζ are parameters to control the decay rates. The freeflow pace estimate p_{free} traffic is obtained similarly, with the modification that traffic is

smoothed along the freeflow wave speed $v_{\max}$ instead of the congested wave speed w.

The weighting coefficient $\gamma(t,x)$ is defined as:

$$\gamma(t,x) = \frac{1}{2}\left[1 + \tanh\left(\frac{v_c - u^*(t,x)}{\Delta V}\right)\right]. \tag{5}$$

In (5), v_c is the critical velocity (i.e., where traffic transitions from freeflowing to congested), ΔV is a smoothing width for the traffic state transition, and $u^*(t,x)$ is given as:

$$u^*(t,x) = \min\left\{\frac{1}{p_{\text{cong}}(t,x)}, \frac{1}{p_{\text{free}}(t,x)}\right\}. \tag{6}$$

The purpose of $u^*(t,x)$ is to determine if p_{cong} or p_{free} indicate traffic conditions are congested. As a result, (5) produces coefficients that favor the congested estimate when either p_{cong} or p_{free} indicate congestion (i.e., $\gamma(t,x) \to 1$), and favors the freeflow estimate otherwise.

A final step to recover the velocity estimate is to invert the resulting pace estimate $\hat{v}(t,x) = 1/\hat{p}(t,x)$.

2.3 Kalman Filtering

The Kalman filter [37] is an algorithm used to estimate the state of a linear system by correcting a model based prediction with measurement data. The Kalman filter and its nonlinear extensions necessary for traffic estimation can be understood by posing the model and sensor measurement processes in state space form as follows:

$$\begin{cases} x^n = \mathcal{F}(x^n) + \eta^n, \\ \tilde{y}^n = \mathcal{H}(x^n) + \epsilon^n. \end{cases} \tag{7}$$

The first equation of (7) is known as the evolution equation, where x^n is the traffic state vector at time step n to be estimated, $\mathcal{F}$ denotes the traffic flow model used to predict the traffic state at time n given the traffic state at time $n-1$, and $\eta^n \sim (0, \mathbf{Q})$ is a white noise process with covariance $\mathbf{Q}$. The second equation of (7) is the observation equation which relates the vector of measurements $\tilde{y}^n$ received at time n with the traffic state variables x^n through the measurement model $\mathcal{H}$. The random variable $\epsilon^n \sim (0, \mathbf{R})$ denotes the measurement error distribution which is modeled as a white noise with covariance $\mathbf{R}$. The Kalman filter and its nonlinear variants are sequential state estimators that are optimal estimators (in the best linear unbiased sense) of the state x^n given a sequence of measurements $\tilde{y}^0, \cdots, \tilde{y}^n$.

Considering the nonlinearity of the traffic models, a number of variations of Kalman filter have been explored for traffic estimation, including the *extended Kalman filter* (EKF) [21], the *unscented Kalman filter* (UKF) [38], the *mixture Kalman filter* (MKF) [39, 40], the *particle Filter* (PF) [41, 42], and the *ensemble Kalman filter* (EnKF) [43]. The EKF, UKF, MKF, EnKF rely linearization of

the system $\mathcal{F}$ or closure assumptions on the state error distributions, while the particle filter is a fully nonlinear Bayesian estimator that can be computationally expensive for large systems. Consequently the EnKF is implemented as representative filtering algorithm for work zone traffic state estimation, and is briefly summarized next.

2.3.1 Evolution Equation

The traffic dynamics on a road segment is governed the LWR model [35, 36], which expresses conservation of vehicles on the roadway:

$$\frac{\partial \rho(t,x)}{\partial t} + \frac{\partial \psi\left(\rho(t,x)\right)}{\partial x} = 0, \tag{8}$$

where $\psi(\rho) = \rho\mathcal{V}(\rho)$ is the *fundamental diagram* describing the relationship between the density and flow on a highway. The model is closed by assuming an empirical relationship between the velocity and density of traffic, defined by the velocity function $\mathcal{V}$. In this work, the following velocity function is used:

$$\mathcal{V}(\rho) = \begin{cases} v_{\max}\left(1 - \frac{1}{\beta}\right) & \text{if } \rho \leq \rho_c, \\ w(\rho - \rho_{max})/\rho & \text{if } \rho > \rho_c, \end{cases} \tag{9}$$

where the parameter $v_{\max}$ is the maximum free flow velocity, β controls the slope of the velocity function in free flow, ρ_c and ρ_{max} are the critical and jam density, and w is the maximum backwards wave speed.

The LWR PDE is discretized using the Godunov scheme [44] resulting in the cell transmission model [45]. Suppose the time and space domain $[0,T] \times [0,L]$ is evenly discretized into time steps of size ΔT indexed by $n \in \{0,\ldots,n_{max}\}$, and spatial cells indexed by $i \in \{0,\ldots,i_{max}\}$ with cell length Δx. By the conservation of vehicles, at time step n, the traffic density ρ_i^n in cell i evolves according to:

$$\rho_i^n = \rho_i^{n-1} + \frac{\Delta T}{\Delta x}\left(q_{i-1/2}^{n-1} - q_{i+1/2}^{n-1}\right), \tag{10}$$

where $q_{i-1/2}^n$ represents the flow between cell $i-1$ and cell i during time step n. The flow over the cell boundary is computed as the minimum flow that can be sent from the upstream cell $i-1$ and the flow that can be received by the downstream cell i:

$$q_{i-1/2}^n = \min\left\{S(\rho_{i-1}^n), R(\rho_i^n)\right\}, \tag{11}$$

where S and R are known as the sending and receiving functions. The functions are constructed from the fundamental diagram as:

$$S(\rho) = \begin{cases} \psi(\rho) & \text{if } \rho \leq \rho_c, \\ q_{max} & \text{otherwise}, \end{cases} \qquad R(\rho) = \begin{cases} q_{max} & \text{if } \rho \leq \rho_c, \\ \psi(\rho) & \text{otherwise}. \end{cases} \tag{12}$$

In the present application, the system state at time n consists of the density in each cell. The upstream inflow to the road $q_{-1/2}^n$ and the downstream

outflow $q^n_{i_{max}+1/2}$ are also modeled as state variables, and are assumed to have stationary dynamics. The concatenated state vector is given as $x^n := [\rho^n_0, \rho^n_1, \cdots, \rho^n_{i_{max}}, q^n_{-1/2}, q^n_{i_{max}+1/2}]^T \in \mathbb{R}^{N\times 1}$. The forward model $\mathcal{F}$ is constructed from (10),(11), and the stationary dynamic assumed for the inflow and outflow states.

The model noise variance selected for the density states in cells with an onramp or offramp, and the noise variance associated with the inflow and outflow $q^n_{-1/2}$, $q^n_{i_{max}+1/2}$ are elevated to account for the larger uncertainty in the state evolution.

2.3.2 Observation Equation

The flow measurements $\tilde{q}^n_m$ and velocity measurements $\tilde{q}^n_m$ time n and indexed by m are related to the traffic state variables as follows. The flow measurements at the boundaries are direct observations of the flow state variables. For example, the inflow measurement is related to the inflow state variable by $\tilde{q}^n_m = q^n_{-1/2}$ for the appropriate flow measurement m. Consider a flow measurement m on the interior of the road segment located at the boundary between cells $i-1$ and i. The measurement is related to the density state variable in cell $i-1$ or cell i depending on if traffic is free flowing or congested:

$$\tilde{q}^n_m = \min\left\{S(\rho^n_{i-1}), R(\rho^n i)\right\} + \epsilon^n_m, \tag{13}$$

where ϵ^n_m is the noise associated with the flow measurement.

Velocity measurements are related to the state variables as follows. Let $\tilde{v}^n_m$ denote the a velocity measurement at time n at the boundary between cells $i-1$ and i. The velocity is linked to the density state variable in either the upstream or downstream cell depending on the traffic conditions as:

$$\tilde{v}^n_m = \begin{cases} \mathcal{V}(\rho^n_{i-1}) + \epsilon^n_m & \text{if } S(\rho^n_{i-1}) < R(\rho^n_i), \\ \mathcal{V}(\rho^n_i) + \epsilon^n_m & \text{if } R(\rho^n_i) < S(\rho^n_{i-1}), \\ \mathcal{V}(\rho_c) + \epsilon^n_m & \text{if } R(\rho^n_i) = S(\rho^n_{i-1}), \end{cases} \tag{14}$$

where ϵ^n_m is represents the noise on the velocity measurement.

The complete set of measurements at time n is written as a vector

$$\tilde{y}^n := [\tilde{q}^n_1, \cdots, \tilde{q}^n_{M_q}, \tilde{v}^n_1, \cdots, \tilde{v}^n_{M_v}]^T \in \mathbb{R}^{M_v+M_q}, \tag{15}$$

where M_q and M_v denote the number of flow and velocity measurements respectively. The nonlinear observation equations (13),(14) and the appropriate measurement error models are used to construct the observation equation in (7).

2.3.3 Ensemble Kalman Filter

The general algorithmic steps of the ensemble Kalman filtering algorithm are briefly summarized following [46]. The interested reader is referred to [47] for the theoretical foundation and detailed interpretation of the technique.

Consider the discrete-time nonlinear system (7). Since the observation equation is nonlinear, an augmented state approach [46] is adopted to linearize the observation equation. In the augmented system, the state is denoted by $s^n = \begin{bmatrix} (x^n)^T & (\mathcal{H}(x^n))^T \end{bmatrix}^T$, which has the following observation equation:

$$\tilde{y}^n = \mathbf{H}s^n + \epsilon^n = \begin{bmatrix} \mathbf{0}_{(M_v+M_q)\times N} & \mathbf{I}_{(M_v+M_q)\times(M_v+M_q)} \end{bmatrix} s^n + \epsilon^n. \tag{16}$$

Given the augmented state s with a linear observation equation (16), the EnKF algorithm consists of the following steps, namely, initialization, prediction, and correction. After the algorithm is initialized, it predicts the best estimate at time n with measurements through time $n-1$, denoted $\hat{s}^{n|n-1}$, using the traffic evolution equation and the best estimate $\hat{s}^{n-1|n-1}$. After measurements are received at time n, the predicted state $\hat{s}^{n|n-1}$ is updated to $\hat{s}^{n|n}$ by incorporating the new information contained in the measurements. The algorithm is as follows:

1) Initialize a set of state ensembles $\hat{s}^{0|0}(k), \forall k \in \{1, \cdots, k_{max}\}$ from the initial state distribution with covariance $\mathbf{P}^{0|0}$.

2) Predict the traffic states $\hat{s}^{n|n-1}(k)$ for all ensembles using $\mathcal{F}$ and $\mathcal{H}$ where the model noise $\eta^n(k)$ is independently sampled for each ensemble k from a distribution with model noise covariance $\mathbf{Q}$.

   Then the predicted state error covariance can be computed as

$$\mathbf{P}^{n|n-1} = \frac{1}{k_{max}-1}\mathbf{E}^{n|n-1}\left(\mathbf{E}^{n|n-1}\right)^T. \tag{17}$$

   In (17), $\mathbf{E}^{n|n-1}$ is the state error matrix which is computed by subtracting the estimated state $\hat{s}^{n|n-1}(k)$ for each ensemble k by the mean of estimated states $\bar{s}^{n|n-1}$:

$$\mathbf{E}^{n|n-1} = \left[\hat{s}^{n|n-1}(1) - \bar{s}^{n|n-1}, \ldots, \hat{s}^{n|n-1}(k_{max}) - \bar{s}^{n|n-1}\right], \tag{18}$$

   where $\bar{s}^{n|n-1} = \frac{1}{k_{max}}\sum_{k=1}^{k_{max}} \hat{s}^{n|n-1}(k)$.

3) Given the measurement data $\tilde{y}^n$, the predicted state $\hat{s}^n(k)$ for each ensemble is corrected by the prediction error on the output multiplied by a Kalman gain $\mathbf{K}^n$,

$$\hat{s}^{n|n}(k) = \hat{s}^{n|n-1}(k) + \mathbf{K}^n(\tilde{y}^n - \mathbf{H}\hat{s}^{n|n-1}(k)), \tag{19}$$

   where the Kalman gain is computed using the predicted state error covariance (17):

$$\mathbf{K}^n = \mathbf{P}^{n|n-1}\left(\mathbf{H}^n\right)^T\left(\mathbf{H}\mathbf{P}^{n|n-1}\mathbf{H}^T + \mathbf{R}\right)^{-1}. \tag{20}$$

The second and third steps are repeated sequentially for each time step n until the estimated state over the entire time horizon is obtained. The final velocity estimate is directly computed from the density estimate by applying the velocity function $\mathcal{V}$ to the estimated density field.

2.4 Back of Queue and Travel Time Estimation

Given an estimate of the velocity $\hat{v}(t,x)$ in space and time, a standard approach across algorithms is used to estimate the length of the queue and the travel time. The estimated length of the queue $\hat{l}(t)$ is determined as the maximum length of any consecutive segment in the velocity field such that $\hat{v}(t,x) < \alpha$, where α is a threshold denoting congested traffic velocities. The travel time is computed using an *instantaneous* [48] travel time estimate:

$$\hat{\tau}(t) = \int_0^L 1/\hat{v}(t,x)dx. \tag{21}$$

The instantaneous travel time is valid under the assumption that the velocity field is constant along over the time interval $[t, t+\tau(t)]$, which may fail in scenarios with rapid queue growth or dissipation. The primary benefit of the instantaneous travel time is that it does not require the future traffic state to be predicted.

3 Construction of a Virtual Testbed

With the algorithms defined, it is now shown how micro simulation trajectories are used to generate noisy traffic sensor data that reflects the true errors observed in the field.

The microscopic simulation environment is developed in a traffic simulation software AIMSUN, which uses a modified Gipps' car-following model [49] to generate individual vehicle trajectories at a 0.2 second resolution. AIMSUN allows error free sensors to be plased in the simulation environment, but lacks realistic traffic sensor error models representative of sensors deployed in work zone environments.

3.1 Overview of Sensor Types

To better assess how different types of sensors featuring distinct measurement errors affect the traffic estimation accuracy, realistic sensor models are developed to degrade the simulated traffic measurements to be consistent with the data quality observed in practical field deployments. Sensor models are developed for the three commonly used traffic sensors, including the Doppler radars, *low energy radar* (LER) units (e.g., the iCone sensor), and *remote traffic microwave sensors* (RTMSs).

Doppler radars are widely used in traffic sensing due to the maturity of the technology and low cost. Doppler radars rely on the Doppler effect for measuring the velocity of vehicles, and provides accurate velocity measurement for vehicles (less than one mph [50]), although performance degrades at lower velocities where the Dopplar shift is not as prominent. The traffic flow data is obtained by counting the number of velocity measurements recorded during each detection cycle. For best performance, Doppler radars are also commonly

mounted relatively low to the ground (e.g., at minimum three ft [50]). Consequently the sensor is prone to occlusion issues (e.g., where one vehicle blocks another vehicle from being detected by the sensor), which may result in biased velocity and flow measurements when deployed on multi-lane freeways.

Low energy radar units are deployed in construction barrels for ease of deployment, but with limited energy available for detection. For long term deployments, LER sensors operate for only a portion of the detection cycle (e.g., 30 seconds of detection in a 60 second detection window). Consequently, these sensors have similar types of measurement errors as radar units for velocity and count data, and an additional sampling error due to the discontinuous operation.

An RTMS measures the distance to objects in the path of its microwave beam, hence is able to detect moving and stationary vehicles in multiple detection zones (lanes). The RTMS is commonly mounted in an elevated position (i.e., at least 17 ft [51]), which reduces the occlusion potential and increases the counting accuracy compared to lower sensors. With proper field calibration, the RTMS produces velocity measurements with 10% error, with larger velocity errors in high congestion [51].

3.2 Generation of Noisy Measurements

A multistep process is used to convert the detailed trajectory data from AIMSUN into noisy sensor data to mimic the field data collected from the sensors described above. The first step is to determine which vehicles pass through the detection zone of a given sensor. The detection zones for each sensor are modeled based on the recommended installation guidelines [50] and the reported field of view. For example, the radar and LER sensors are aimed at oncoming traffic, resulting in a detection zone approximately 140 ft upstream of the sensor as shown in Figure 2. The detection zone of the LER is assumed active for only the first half of the detection cycle, resulting in the detection of only a subset of vehicles. The RTMS sensor detection zone is located at the installation point. For a given detection cycle, the vehicles that pass through the detection zone are potentially available to contribute to the average velocity or count measurements.

The next step is to discard any vehicle that passes through the detection zone whose trajectory is occluded by another vehicle. As illustrated in Figure 2, two vehicles, labeled A and B, travel through the sensing area in the outside and inside lane respectively. Vehicle A is considered occluded if more than p_o percent of the trajectory of vehicle A overlaps with the trajectory of B in the detection zone. The parameter p_o is selected as $p_o = 0.3$, which results in the occlusion of approximately 40% of vehicles in heavy congestion. Occluded vehicles are removed for radar and LER sensors, while RTMS sensors are assumed to be mounted in a position to prevent occlusion, but the count is still perturbed by a counting error consistent with the reported accuracy [51].

After occluded vehicles are removed, any vehicle with a velocity outside the measurement range of the sensor is also discarded. The radar and LER sensors have a measurement range $[5, 99]$ mph [50], while the RTMS has a range of

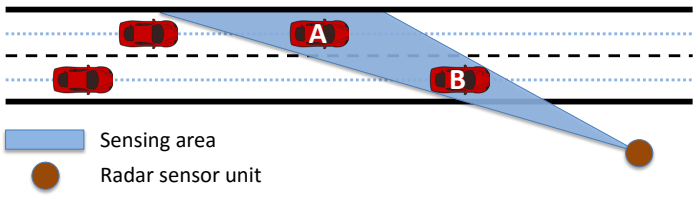


Figure 2: Illustration of an occluded vehicle A caused by the presence of vehicle B in the detection zone.

$[0, 110]$ mph [51].

For vehicles that remain, the measured velocity is assumed to be a noisy reading of the true vehicle velocity. Measured velocities are constructed by adding a measurement error generated from $\mathcal{N}(0, \sigma)$ to the true velocity, where σ takes two distinct values in freeflow and congested traffic. For the radar and LER sensors, σ is chosen such that the true vehicles are measured with an accuracy of 1 mph, while the RTMS velocities are measured to within 10%. At low vehicle velocities, the errors are increased to 2 mph for radar and LER, and 15% for the RTMS devices. Note that although the measurement noise is assumed unbiased, the average velocities and counts are biased due to the sample set (i.e., removing occluded vehicles, which are predominantly from the faster lane of traffic). Finally, the harmonic mean of the noisy velocity measurements is taken as the average velocity reported by the sensor, and the number of measurements is the count.

To model the realistic missing data rates that occur in field deployments, a subset of measurement detection cycles are also discarded resulting in no data available for estimation during the cycle. Up to fifteen percent of the data from the radar and LER sensors and three percent of the data from RTMS sensors is dropped in congestion, based on the missing data rates observed from work zone field data in Illinois.

4 Comparative Analysis

4.1 Experimental Setup and Micro Simulation Calibration with Field Data

A work zone on the I-80 freeway near the Des Plaines River in Illinois is modeled in micro simulation. The work zone included an active sensor network that recorded velocity and flow data at sensors placed approximately once every half mile, which is used to calibrate the micro simulation. Severe congestion was observed in the field from 3:30 PM and 6:00 PM on May 1, 2015, which is the time period selected for calibration. This period provides varying traffic conditions, including queue formation and dissipation, and is suitable for testing

the performance of various sensor networks and algorithms.

Calibration of a micro simulation model from field data is a challenging problem due to the large number of correlated microscopic parameters. Automated calibration of the microscopic parameters using a variety of optimization programs has been explored [52, 53, 54, 55], and a comprehensive comparison of nonlinear optimization programs on the effectiveness of calibrating microscopic model parameters was reported in the MULTITUDE project [55]. Recently, an advanced *nonlinear optimization by mesh adaptive direct search* (NOMAD) [56] software was also used for the calibration of microscopic traffic models [57]. Considering the software availability and the implementation effort, this article adopts NOMAD as the nonlinear optimization software in the calibration process. The *root-mean-square error* (RMSE) is selected as the error metric considering its reliable performance for measuring the goodness of fit [55].

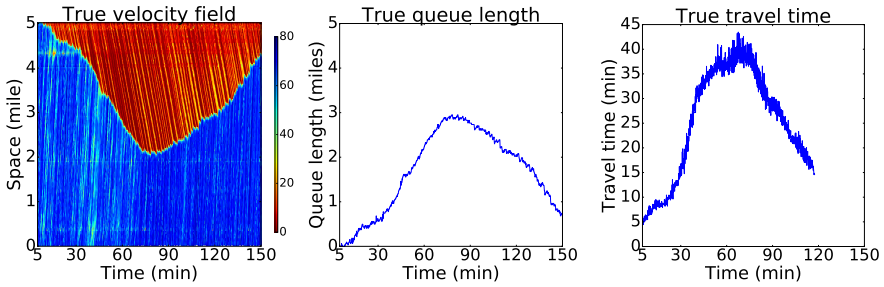
Because of the lack of high quality flow data from the work zone sensor network, only the velocity data is used in the error metric to calibrate the micro simulation. Based on a literature review and a sensitivity analysis of the model, a total of eight parameters were selected for automatic calibration. Using the calibrated parameters, the RMSE error of the model is reduced by approximately 40% from the default values in AIMSUN. A detailed description of the calibrated parameters is available in the supplemental source code associated with this article (`https://github.com/Lab-Work/IDOT-SmartWorkzone`).

4.2 Traffic Estimation Error Metrics

The error metrics used to assess the performance of the various estimators are briefly described. In order to calculate the errors, the true state to be estimated is first calculated from the AIMSUN trajectory data. The true velocity field (Figure 3(a)), true queue length (Figure 3(b)), and true travel time (Figure 3(c)), are constructed on a finely discretized spatio-temporal grid with cells of length 50 m and duration 5 s and within each grid the true velocity is computed using Edie's definitions [58]. The *mean absolute error* (MAE) of the velocity, the travel time, and the length of the queue are reported. The true travel time is taken as the average travel time of all vehicles entering the roadway during the time step, and the true length of the queue is computed as the maximum length of a segment with a velocity of less than 40 mph. For each reported experiment, the MAE is the average MAE of five simulations with different micro simulation seed values.

4.3 Estimation Algorithm Implementation

To calibrate the fundamental diagram (9), a variety of traffic conditions are simulated in AIMSUN, where two RTMS sensors are modeled to extract the velocity and flow measurements. Using this data, the fundamental diagram parameters in (9) calibrated are $v_{\max} = 60.82$ mph, $\beta = 1000$ veh/mile, $w = -9.29$ mph, and $\rho_{max} = 500$ veh/mile. The parameters are determined following



(a) (b) (c)

Figure 3: True state of the (a) velocity field, (b) queue length, and (c) travel time. The true travel time was not obtained in the last period because vehicles that entered the road after approximately 120 min did not exit before the simulation stopped.

the calibration procedure [59] except the maximum density is computed directly from the minimum spacing of stopped vehicles.

The parameters of the spatio-temporal algorithm w, $v_{\max}$ and v_c are also set as -9.29 mph, 60.82 mph and 56.52 mph, respectively, based on the calibrated fundamental diagram. The parameter ΔV is set to 12.43 mph as in [32, 34]. The decay rate controllers κ and ζ are fixed at 0.75 times the sensor spacing and sensor aggregation, respectively, based on an optimization procedure that evaluated the performance of multiple combinations of values that fell in the ranges recommended in [34]. To avoid using excessively old measurements and unnecessarily long computation times, the measurements older than 2.5 minutes are ignored.

4.4 Algorithms and Sensor Spacings

In the first set of experiments, the influence of the sensor spacing and traffic estimation algorithm on the travel time, queue length, and velocity estimation error are analyzed. For each algorithm, eleven spacings ranging from five miles to 1/8 mile are considered. All sensors are assumed to be RTMS sensors in these experiments.

For each experiment, the MAE on the velocity estimate is computed both as an average over the entire spatio-temporal domain, as well as in the area immediately around the true back of queue (+/-0.5 mile) as identified in AIMSUN. The resulting errors as a function of the algorithm type and sensor spacing is shown in Figures 4(a) and Figure 4(b). As expected, as the density of sensors increases, all algorithms result in lower velocity errors, with the best performance (about 5 mph error overall and 6 mph around the queue) by the linear interpolation algorithm with sensors placed every 1/8th of a mile. Not surprisingly, all algorithms perform worse in the neighborhood of the queue compared

to the MAE reported over all space and time, which is unfortunately where the errors may be most safety critical. The spatio-temporal algorithm has the highest error around the queue, which results in higher total MAE compared to the spatial interpolation algorithm. Note the algorithm was originally proposed as an offline algorithm [26], and the performance may change if additional measurement data is available for smoothing. At very dense sensor spacings, the EnKF performs worse than the interpolation and smoothing algorithms due to the fact that the velocity is computed from the density estimate through (9), rather than directly estimated as in the better performing algorithms. However, in the neighborhood of the queue, the EnKF localizes the queue more accurately which generally results in higher quality velocity estimates in the neighborhood of the queue.

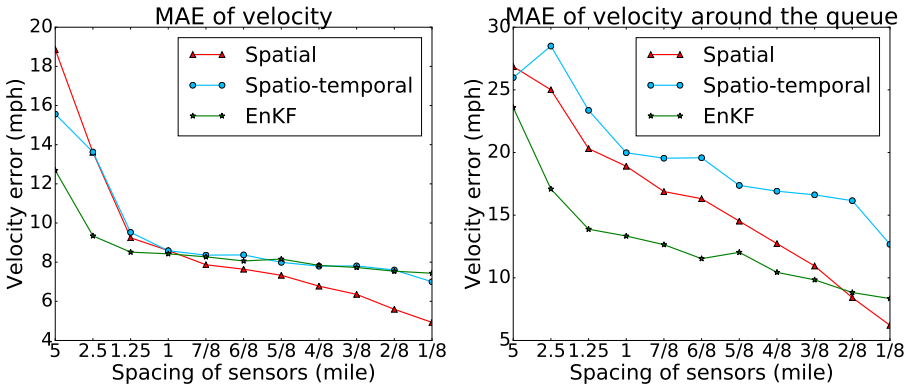
The general trend of the MAE for the queue length estimation (Figure 5(a)) is similar to the velocity MAE. This is a direct result of the fact that the queue length is estimated from the velocity field, and consequently improvements on the velocity field result in better queue length estimates. The true maximum true queue length during the simulation is approximately 3 miles. The EnKF uses a traffic model and utilizes the flow measurements from RTMS sensors, to consistently outperform the other estimators across a wide range of sensor spacings.

Finally, the MAE of the travel time estimation is shown in Figure 5(b). The travel time of the road section in free flow is around five minutes and the longest travel time during congestion is 40 minutes. The EnKF is generally the best performing travel time estimator, and offers slightly improved performance over the spatio-temporal smoother. Recall the spatio-temporal algorithm directly estimates the pace of traffic, which results in a better estimate of the travel time than the purely spatial interpolation algorithm. The MAE for travel time is relatively large for all algorithms independent of the spacing, and the largest source of error is due to the use of the instantaneous travel time calculation (21), not the underlying velocity estimate. In fact, the true instantaneous travel time has a MAE of over six minutes, which is comparable to the MAE observed in the travel time estimates of the best performing algorithms. In traffic conditions with slower dynamics (e.g., in free flow or in complete congestion), the use of the instantaneous travel time may result in lower errors.

4.5 Type of Sensors

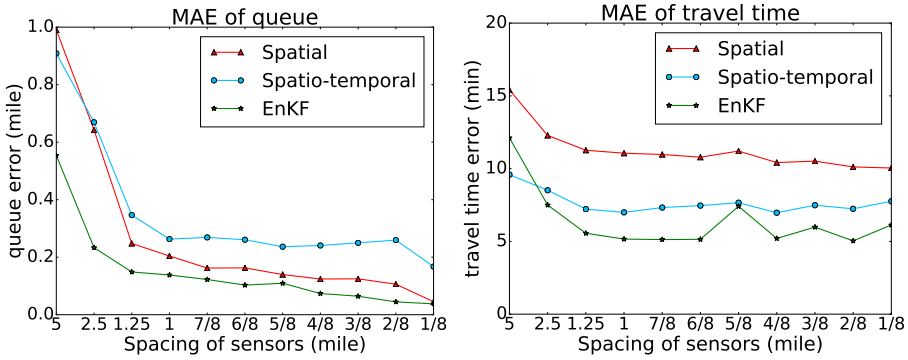
In the next set of experiments, the influence of the sensor type (i.e., RTMS, the radar, and the LER) and the errors they introduce is compared across algorithms. Recall that the RTMS offers the lowest quality velocity measurement of individual vehicles but provides more reliable count data compared to the radar based sensors. The MAE for the estimated traffic velocity, the queue length, and the travel time are shown in Figure 6(a), Figure 6(b), Figure 6(c) respectively, for sensors placed at a spacing of one mile.

To understand the potential benefit of improved sensor technologies, traffic estimates using an ideal sensor are also generated. The ideal sensor is assumed



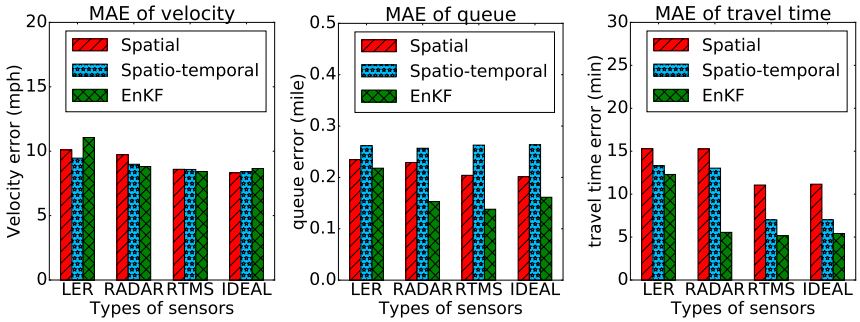
(a) (b)

Figure 4: MAE of velocity estimation over (a) the entire time space horizon and (b) around the queue using RTMS sensors.



(a) (b)

Figure 5: MAE of the (a) queue length estimation and (b) travel time estimation using RTMS sensors.



(a) (b) (c)

Figure 6: MAE of the (a) velocity estimation, (b) queue estimation, and (c) travel time estimation at 1 mile spacing.

to have zero error (i.e., it measures the velocity of every vehicle exactly, and has no occlusion or dropped packets). It does have a quantization error due to the fact that the count and velocity is computed within a single detection cycle. For example, if the aggregation interval is 30 s, the quantization error (e.g., including or excluding a single vehicle near the cycle boundary) introduces a change in the flow of 120 veh/hr/lane, which is around five percent of the lane capacity. Algorithms running with measurements from the ideal sensor have velocity, queue, and travel time errors that are similar to the RTMS sensors, which indicates that quantization error is the largest source of remaining error from the existing sensors. Indeed, the existing commercial RTMS sensors already achieve flow measurements to within five percent error, which indicates that the measurement accuracy will be dominated by the quantization error even if the sensor accuracy is improved. Marginal benefit will be obtained if the quality of the sensing device is improved beyond the current market technologies, and the general trends are similar at other sensor spacings.

The analysis also indicates that the EnKF algorithm is the most sensitive algorithm to the sensor type, and provides the poorest velocity estimates when combined with LER sensors. This is due to the reliance of the EnKF algorithm on accurate flow data, which is degraded in the LER devices due to the fact that vehicles are recorded only for a portion of the detection cycle (resulting in increased quantization error), and the counts are prone to larger occlusion errors. Across all traffic quantities and all algorithms, the radar and RTMS sensors offer at least as good or better performance compared to the LER devices. The use of LER devices may still be warranted if the deployment cost, maintenance, and portability make the sensor competitive with other sensing technologies.

5 Conclusion

This article evaluated a variety of sensor network configurations and algorithms with different levels of sophistication to estimate the traffic velocity, from which back of queue and travel time were estimated. The main findings are as follows. The spacing of sensors is an important factor for improving the accuracy of traffic estimation, especially at sparse sensor spacings. When the sensor spacing is smaller than 0.5 miles, the benefit of additional sensors or the choice of algorithm is marginal (i.e., less than five percent except in the neighborhood of the queue). A major benefit from the RTMS sensor is the significant reduction in error on the measured flow. For the flow model based Kalman filter, the improved flow measurements allow the estimator to perform well, even when sensors are sparsely placed. For spatio-temporal and spatial interpolation algorithms, the flow data is ignored and consequently the improved flow data offers no estimation algorithm performance benefit. Finally, the fact that all algorithms perform relatively poorly for travel time estimation highlights the fact that the instantaneous travel time calculation is a poor estimator of the true travel time in a dynamic traffic environment. Further predictive analytics are needed to reduce the travel time error.

The main limitation of the study is that the analysis is conducted in a micro simulation environment. Further field testing is needed to validate the findings. Furthermore, the present analysis is restricted to the accuracy of the traffic state estimate only, and did not consider deployment and maintenance costs, which may also play a critical role in the development of practical sensor network configuration guidelines.

6 Acknowledgement

This publication is based on the results of ICT-R27-155, Improving the effectiveness of smart work zone technologies. ICT-R27-155 was conducted in cooperation with the Illinois Center for Transportation; the Illinois Department of Transportation, Division of Highways; and the U.S. Department of Transportation, Federal Highway Administration. The contents of this publication reflect the view of the authors, who are responsible for the facts and the accuracy of the data presented herein. The contents do not necessarily reflect the official views or policies of the Illinois Center for Transportation, the Illinois Department of Transportation, or the Federal Highway Administration. This publication does not constitute a standard, specification, or regulation.

References

[1] FHWA, Intelligent transportation systems in work zones: A cross-cutting study, Tech. Rep. FHWA-OP-02-025, Federal Highway Administration (2002).

[2] L. Tudor, A. Meadors, R. Plant, Deployment of smart work zone technology in Arkansas, Transportation Research Record: Journal of the Transportation Research Board 1824 (2003) 3–14.

[3] FHWA, Real-time work zone traffic control system, using an automated traffic information system to reduce congestion and improve safety during reconstruction of the I-55 lake Springfield bridge in Illinois, Tech. Rep. FHWA-HOP-04-018, Federal Highway Administration (2004).

[4] T. Roelofs, C. Brookes, Synthesis of intelligent work zone practices, Tech. Rep. ENT-2014-1, Athey Creek Consultants, and Michigan Department of Transportation (2014).

[5] G. Ullman, J. Schroeder, Mitigating work zone safety and mobility challenges through Intelligent Transportation Systems: Case Studies, Tech. Rep. FHWA-HOP-14-007, Federal Highway Administration (2014).

[6] FHWA, Dynamic lane merge system, reducing aggressive driving and optimizing throughput at work zone merges in Michigan, Tech. Rep. FHWA-HOP-04-033, Federal Highway Administration (2004).

[7] E. Radwan, R. Harb, S. Ramasamy, Evaluation of safety and operational effectiveness of dynamic lane merge system in Florida, Florida Department of Transportation, 2009.

[8] FHWA, Reducing congestion with the use of a traffic management contract incentive during the reconstruction of Arizona State Route 68, Tech. Rep. FHWA-HOP-04-032, Federal Highway Administration (2004).

[9] T. Luttrell, M. Robinson, J. A. Rephlo, R. Haas, J. Srour, R. F. Benekohal, J. Oh, T. Scriba, Benefits of using intelligent transportation systems in work zones-A summary report, Tech. Rep. FHWA-HOP-08-021, Federal Highway Administration (2008).

[10] R. P. Maccubbin, B. L. Staples, F. Kabir, C. F. Lowrance, M. R. Mercer, B. H. Philips, S. R. Gordon, ITS benefits, costs, deployment, and lessons learned, Tech. Rep. FHWA-JPO-08-032, Federal Highway Administration (2008).

[11] T. K. Datta, K. L. Schattler, P. Kar, A. Guha, Development and evaluation of an advanced dynamic lane merge traffic control system for 3 to 2 lane transition areas in work zones, Tech. Rep. RC-1451, Michigan Department of Transportation (2004).

[12] L. Grillo, T. Datta, C. Hartner, Dynamic late lane merge system at freeway construction work zones, Transportation Research Record: Journal of the Transportation Research Board 2055 (2008) 3–10.

[13] R. F. Benekohal, A. Hajbabaie, J. C. Medina, M. Wang, M. V. Chitturi, Speed photo-radar enforcement evaluation in Illinois work zones, Tech. Rep. ICT-10-064 UILU-ENG-2010-2004, Illinois Center for Transportation (2010).

[14] P. J. Carlson, M. D. Fontaine, H. G. Hawkins, Evaluation of traffic control devices for rural high-speed maintenance work zones, Tech. Rep. FHWA/TX-00/1879-1, Texas Transportation Institute, Texas A & M University System (2000).

[15] E. Lee, C. Kim, Automated work zone information system on urban freeway rehabilitation: California implementation, Transportation Research Record: Journal of the Transportation Research Board 2006 (2006) 77–85.

[16] R. Bushman, C. Berthelot, Estimating the benefits of deploying intelligent transportation systems in work zones, in: Proceedings of Transportation Research Board 83rd Annual Meeting Compendium of Papers, 2004.

[17] A. Haghani, M. Hamedi, R. Fish, A. Nouruzi, Evaluation of dynamic message signs and their potential impact on traffic flow, Tech. Rep. MD-13-SP109B4C, Maryland Department of Transportation (2013).

[18] C. Dankocsik, M. Zarean, D. Register, K. Timpone, Best practices of rural and statewide ITS strategic planning, Tech. Rep. FHWA-OP-02-037, Federal Highway Administration (2002).

[19] T. Luttrell, M. Robinson, J. A. Rephlo, R. Haas, J. Srour, R. F. Benekohal, J. Oh, T. Scriba, Comparative analysis report: The benefits of using intelligent transportation systems in work zones, Tech. Rep. FHWA-HOP-09-002, Federal Highway Administration (2008).

[20] G. Ullman, J. Schroeder, D. Gopalakrishna, Work zone intelligent transportation systems implementation guide: Use of technology and data for effective work zone management, Tech. Rep. FHWA-HOP-14-008, Battelle, and Texas A&M Transportation Institute (2014).

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

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

[23] J. Oh, R. Jayakrishnan, W. Recker, Section travel time estimation from point detection data, in: Proceedings on 82nd Transportation Research Board Annual Meeting, 2002.

[24] B. Fariello, ITS America RFI travel time projects in North America, Report of San Antonio TransGuide Program.

[25] C. Cortes, R. Lavanya, J.-S. Oh, R. Jayakrishnan, General-purpose methodology for estimating link travel time with multiple-point detection of traffic, Transportation Research Record: Journal of the Transportation Research Board 1802 (2002) 181–189.

[26] J. Van Lint, Empirical evaluation of new robust travel time estimation algorithms, Transportation Research Record: Journal of the Transportation Research Board 2160 (2010) 50–59.

[27] E. Kwon, Development of operational strategies for travel time estimation and emergency evacuation on a freeway network, Tech. Rep. MN/RC-2004-49, Minnesota Department of Transportation (2004).

[28] J. Van Lint, N. Van der Zijpp, Improving a travel-time estimation algorithm by using dual loop detectors, Transportation Research Record: Journal of the Transportation Research Board 1855 (2003) 41–48.

[29] L. Sun, J. Yang, H. Mahmassani, Travel time estimation based on piecewise truncated quadratic speed trajectory, Transportation Research Part A: Policy and Practice 42 (1) (2008) 173–186.

[30] D. Ni, H. Wang, Trajectory reconstruction for travel time estimation, Journal of Intelligent Transportation Systems 12 (3) (2008) 113–125.

[31] B. Coifman, Estimating travel times and vehicle trajectories on freeways using dual loop detectors, Transportation Research Part A: Policy and Practice 36 (4) (2002) 351–364.

[32] M. Treiber, D. Helbing, Reconstructing the spatio-temporal traffic dynamics from stationary detector data, Cooper@tive Tr@nsport@tion Dyn@mics 1 (2002) 3.1–3.24.

[33] J. Van Lint, S. Hoogendoorn, A robust and efficient method for fusing heterogeneous data from traffic sensors on freeways, Computer-Aided Civil and Infrastructure Engineering 24 (1) (2009) 1–17.

[34] M. Treiber, A. Kesting, R. Wilson, Reconstructing the traffic state by fusion of heterogeneous data, Computer-Aided Civil and Infrastructure Engineering 26 (6) (2011) 408–419.

[35] M. Lighthill, G. Whitham, On kinematic waves. II. A theory of traffic flow on long crowded roads, Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences 229 (1955) 317–345.

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

[37] R. E. Kalman, A new approach to linear filtering and prediction problems, Journal of Basic Engineering 82 (1) (1960) 35–45.

[38] A. Hegyi, D. Girimonte, R. Babuska, B. De Schutter, A comparison of filter configurations for freeway traffic state estimation, in: Proceedings of the International Conference on Intelligent Transportation Systems, 2006, pp. 1029–1034.

[39] X. Sun, L. Munoz, R. Horowitz, Highway traffic state estimation using improved mixture Kalman filters for effective ramp metering control, in: Proceedings of the 42nd IEEE Conference on Decision and Control, Vol. 6, 2003, pp. 6333–6338.

[40] Y. Sun, D. B. Work, A distributed local Kalman consensus filter for traffic estimation, in: 53rd IEEE Conference on Decision and Control, 2014, pp. 6484–6491.

[41] L. Mihaylova, R. Boel, A. Hegyi, Freeway traffic estimation within particle filtering framework, Automatica 43 (2) (2007) 290–300.

[42] R. Wang, D. B. Work, R. Sowers, Multiple model particle filter for traffic estimation and incident detection, IEEE Transactions on Intelligent Transportation Systems (2016) 1–10.

[43] D. Work, S. Blandin, O. Tossavainen, B. Piccoli, A. M. Bayen, A traffic model for velocity data assimilation, Applied Mathematics Research eXpress 2010 (1) (2010) 1–35.

[44] S. K. Godunov, A difference method for numerical calculation of discontinuous solutions of the equations of hydrodynamics, Matematicheskii Sbornik 89 (3) (1959) 271–306.

[45] C. F. Daganzo, The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory, Transportation Research Part B: Methodological 28 (4) (1994) 269–287.

[46] R. Lorentzen, G. Nævdal, An iterative ensemble Kalman filter, IEEE Transactions on Automatic Control 56 (8) (2011) 1990–1995.

[47] G. Evensen, The ensemble Kalman filter: Theoretical formulation and practical implementation, Ocean Dynamics 53 (4) (2003) 343–367.

[48] P.-E. Mazare, O.-P. Tossavainen, D. B. Work, Computing travel times from filtered traffic states, Discrete & Continuous Dynamical Systems-Series S 7 (3).

[49] P. Gipps, A behavioural car-following model for computer simulation, Transportation Research Part B: Methodological 15 (2) (1981) 105 – 111.

[50] Houston Radar LLC, Installation and User Manual For K-Band Doppler Radar DR500 (2011).

[51] RTMS G4 User Guide, Image Sensing Systems, Inc (2012).

[52] J. Maryak, D. Chin, Global random optimization by simultaneous perturbation stochastic approximation, in: Proceedings of the 2001 IEEE American Control Conference, 2001, pp. 756–762.

[53] K. Kim, R., Simplex-based calibration of traffic microsimulation models with intelligent transportation systems data, Transportation Research Record: Journal of the Transportation Research Board 1855 (2003) 80–89.

[54] J. Ma, H. Dong, H. Zhang, Calibration of microsimulation with heuristic optimization methods, Transportation Research Record: Journal of the Transportation Research Board 1999 (2007) 208–217.

[55] B. Ciuffo, V. Punzo, M. Montanino, The calibration of traffic simulation models: Report on the assessment of different goodness of fit measures and optimization algorithms, MULTITUDE Project: COST Action TU0903. JRC68403, 2012.

[56] S. Le Digabel, Algorithm 909: NOMAD: Nonlinear optimization with the MADS algorithm, ACM Transactions on Mathematical Software 37 (4) (2011) 1–15.

[57] L. Gauthier, N. Saunier, S. Le Digabel, G. Cao, Calibration of driving behavior models using derivative-free optimization and video data for Montreal highways, in: Proceedings on 95th Transportation Research Board Annual Meeting, 2016.

[58] L. Edie, Discussion of traffic stream measurements and definitions, in: Proceedings of the 2nd International Symposium on the Theory of Road Traffic Flow, 1965, pp. 139–154.

[59] G. Dervisoglu, G. Gomes, J. Kwon, R. Horowitz, P. Varaiya, Automatic calibration of the fundamental diagram and empirical observations on capacity, in: Proceedings of Transportation Research Board 88th Annual Meeting, Vol. 15, 2009.