



Low-dimensional estimation and prediction framework for description of the oscillatory traffic dynamics

Jakub Krol^{1,*}, Bani Anvari^{1,*}, and Roberto Lot²

*Corresponding authors

¹Centre for Transport Studies, Faculty of Engineering, University College London, WC1E 6BT, UK

²Faculty of Engineering and the Environment, University of Southampton, SO17 1BJ, UK

j.krol@ucl.ac.uk

b.anvari@ucl.ac.uk

roberto.lot@soton.ac.uk

Abstract

Large majority of control methodologies used in traffic applications require short-time prediction of the environment. For instance, in widely-used Model Predictive Control [1] employed to reduce fuel and energy consumption of vehicles in a platoon, information about future velocity profiles of leading vehicles is necessary. In such case, the dynamic model should provide information more detailed than prediction of averaged and global quantities. Additionally, if the control input is to be applied at high-frequencies, traffic model must be solved in a short period of time.

We propose a novel framework which addresses aforementioned problems by estimating the vehicle velocity at any location in the domain based on the real-time information from induction loops downstream. Additionally, our formulation is linear and low-dimensional (i.e. consists of few degrees of freedom) meaning that the estimation can be executed at high frequencies. First a mapping is constructed from velocities at discrete locations to the smooth continuous field, which is subsequently projected onto its most significant principal components. Next, current state of such system is estimated using Kalman filter by combining the linear, wave-like dynamics of the traffic with the instantaneous information provided by induction loops. Short-term traffic prediction is then achieved by integration of the model forward in time.

The proxy methodology is validated using SUMO simulation on the test case of the vehicles approaching a traffic junction. The performance is evaluated based on sampling reconstructed continuous waveform at the locations and timestamps of the vehicles in the reference data and calculating velocity errors. Separate cases are considered where drivers follow Intelligent Driver Model perfectly and with varying levels of uncertainty.

1 Introduction

Dynamic prediction is vital for construction of velocity profiles that can be used for encouraging eco-driving to reduce travel time, pollution and fuel consumption [2, 3, 4]. Travel prediction

methodologies can be in general categorised based on the frame of reference relative to the driver or to the side of the road. In the first case, models are largely based on the car following models and deal with platoon dynamics. The latter case aims to describe traffic using quantities such as density and flow, and models inspired by fluid dynamics. The reconciliation of both is non-trivial. Some attempts in this regards are summarised and compared in [5].

Traffic data can be collected through various methods (e.g., loop detectors, GPS) with various spatio-temporal granularity. In traffic data analysis, one of the challenges is to map traffic data collected from stationary detectors to a continuous field. In scientific investigations, isotropic smoothing and anisotropic kernel-based smoothing are common approaches for processing and reconstructing velocity data collected from stationary detectors. Treiber and Helbing [6] proposed a kernel-based traffic-adaptive smoothing method to produce a detailed reconstruction of traffic conditions between stationary loop detectors, and to combine different data sources. Various studies demonstrated that a kernel-based interpolation is an accurate and robust approach in comparison to isotropic data smoothing methodologies [6, 7, 8, 9, 10]. Another challenge in traffic data analysis is to decompose the high-dimensional traffic data into principal variables that approximate different classes of traffic. Principal Component Analysis (PCA) [11, 12] is commonly used for traffic matrix analysis [13, 14, 15, 16]. Previous studies have shown that PCA can provide a good approximation of traffic classes with few principle components, however, its accuracy and efficiency decreases in the presence of large anomalies in the system [16, 17].

In this paper, we propose a proxy method which estimates full velocity field based on induction loop measurements and subsequently uses wave-like dynamics for velocity prediction. In the first step continuous data is constructed using weighted average of vehicles present in the link. The dimension of resulting data is dependent on the discretisation level set by the user, and can lead to high-dimensional data. Low-dimensional state is constructed by a projection of continuous velocity-density field onto few principle components. Subsequently, an estimation using the Kalman filter is performed.

The paper is organized as follows: Sections 2 and 3 explain the proposed methodology and the case study in-details. The results of the proposed framework on the case studies are presented in Section 4. The conclusions are summarised in Section 5.

2 Problem Formulation

Fixed point sensors such as inductive loop detectors allow collecting speed data at constant spatial locations. In order to construct a velocity field based on real-time traffic measurements from inductive loops, we analyse the system using an Eulerian reference frame. The network is therefore discretised into a series of spatial locations with the objective of estimating velocity at each grid point. Given such an estimated state, a model is constructed to dynamically predict the evolution of the high-dimensional velocity field forward in time.

In order to achieve this objective, state-space formulation of the dynamical system is utilised

$$\mathbf{a}_{t+1} = A(\mathbf{a}_t), \quad (1a)$$

$$\mathbf{y}_t = C(\mathbf{a}_t). \quad (1b)$$

In such a framework, $\mathbf{a} \in \mathbb{R}^r$ is a state vector which consists of variables representing some information about the system. Here, those quantities relate to the velocity field and can, for example, denote velocities at different locations. The values of $\mathbf{y}_t \in \mathbb{R}^p$ represent observables

of the system. In traffic modelling those can represent some measurements corresponding to a vehicle driving through the network or information from induction loops such as mean speed or flow. In this study, \mathbf{y}_t is a function of velocity measurements from induction loops at time t . The function $A : \mathbb{R}^r \rightarrow \mathbb{R}^r$ describes dynamics of the system, whilst $C : \mathbb{R}^r \rightarrow \mathbb{R}^p$ is mapping between state and observations. The broad aim of the proposed estimation methodology is to infer current state \mathbf{a}_t given, usually imperfect, forms of \mathbf{y}_t, A, C .

The estimation framework is constructed using the following steps:

- Constructing continuous velocity fields on the analysed network from discrete measurements associated with vehicles in the system (see Section 2.1).
- Reducing the number of degrees of freedom by projecting data onto low-dimensional vector space in order to make the problem tractable for larger networks (see Section 2.2).
- Defining a linear dynamic model on the low-dimensional state together with linear map between the state. Subsequently, combining both models together through a Kalman filter (see Section 2.3).

Detailed description of each step is given in the following subsections.

2.1 Construction of a continuous velocity field

The availability of a continuous form of the velocity field allows us to assign the speed at any point in the domain for a certain prediction horizon and consequently reconstruct future velocity profile of a vehicle. Nonetheless, traffic data is given in the discrete form, where vehicles speed is associated with only one spatial location. In order to map discrete measurements into continuous domain we use modified Adaptive Smoothing Method (ASM) [6].

For a given link at time t_j , there exist $n(t_j) \in \mathbb{N}$ measurements $z_{i,j} = z(x_i, t_j)$ representing a characteristic quantity and acquired at irregular locations x_i for $i \leq n(t_j)$. For example, they can be obtained using induction loops, where $z_{i,j}$ could denote average velocity or flow. In the analysed case, $z_{i,j}$ represent instantaneous velocity of the vehicles in the system. Note that $n(t_j)$ is not time-invariant and depends on the number of vehicles on the lane at each instance.

The ASM maps $z_{i,j}$ to a regular grid space using the weighted average of the measurements close to each grid point such that

$$z(x, t) = \frac{1}{\sum_{i=1}^{n(t_j)} \sum_{j=j_{min}}^{j_{max}} \Phi(\Delta x, \Delta t^*)} \sum_{i=1}^{n(t_j)} \sum_{j=j_{min}}^{j_{max}} \Phi(\Delta x, \Delta t^*) z_{i,j}, \quad (2)$$

where weights are generated by a nonlinear kernel $\Phi(\Delta x, \Delta t^*)$, the appropriate form of which will be discussed later. The values of $\Delta x = x_i - x$ and $\Delta t^*(c) = t_j - t - \frac{\Delta x}{c}$ correspond respectively to spatial and temporal distances of measurement $z_{i,j}$ at x_i, t_j from grid point at x, t . Note that Δt^* depends on speed c which defines a rate at which measurements are propagated in the spatio-temporal domain. The ASM is based on the assumption that a traffic perturbation will either propagate downstream at free flow speed c_{free} or upstream at congested speed c_{cong} . Consequently, two interpolated quantities are obtained $z_{free}(x, t)$ for $\Delta t^*(c_{free})$ and $z_{cong}(x, t)$ for $\Delta t^*(c_{cong})$.

The final interpolated value z_{interp} is calculated as

$$z_{interp} = w(V_{cong}, V_{free})z_{cong} + (1 - w(V_{cong}, V_{free}))z_{free}.$$

Parameter	Description	Assigned Value ($\frac{km}{h}$)
c_{free}	Propagation velocity of perturbations in free traffic	80
c_{cong}	Propagation velocity of perturbations in congested traffic	-15
V_c	Crossover from free to congested traffic	60
ΔV	Width of the transition region	20

Table 1: A summary of the parameters and their typical values in the Adaptive Smoothing Method as defined in [6].

Here, the weight $w(V_{cong}, V_{free})$ is a function of smoothed velocities, i.e. $V_{free} = z_{free}$ if $z_{i,j}$ denote speed measurement (with equivalent definition of V_{cong}), and is given by

$$w(V_{cong}, V_{free}) = \frac{1}{2} \left[1 + \tanh \left(\frac{V_c - \min(V_{cong}, V_{free})}{\Delta V} \right) \right],$$

where V_c and ΔV are re-scaling parameters. Note when $\Delta V \rightarrow 0$, $w \rightarrow 1/2$ such that V_{cong} and V_{free} have equal weighting. Table 1 shows a summary of the values assigned to the parameters in the ASM in [6]. It is noted that the methodology is robust with respect to small changes in those parameters [6].

The final mapped quantity heavily depends on the choice of kernel $\Phi(\Delta x, \Delta t^*)$. In [6],

$$\Phi(\Delta x, \Delta t^*) = \exp \left(-\frac{|\Delta x|}{\sigma} - \frac{|\Delta t^*|}{\tau} \right) \quad (3)$$

is used for spatial and temporal smoothing parameters σ and τ . In this study, the smoothed velocity field should take the same value as $z_{i,j}$ when $x = x_i$ and $t = t_j$, i.e.

$$\lim_{\Delta x \rightarrow 0, \Delta t^* \rightarrow 0} \Phi(\Delta x, \Delta t^*) \rightarrow \infty.$$

This is not the case in (3) and consequently the exponential kernel is replaced with

$$\Phi(\Delta x, \Delta t^*) = \left(\frac{|\Delta x|^2}{\sigma} + \frac{|\Delta t^*|^2}{\tau} \right)^{-1}. \quad (4)$$

The relative decay of measurements in space and time can be regulated through appropriate choice of the smoothing parameters σ and τ . In the following analysis $\sigma = \tau = 1$, however, further investigation is required to determine what values are the most beneficial for dynamic modelling purposes.

Note that $\Phi = 0$ only when either $\Delta x \rightarrow \infty$ or $\Delta t \rightarrow \infty$. Here, the modified ASM is applied separately for each edge and consequently if vehicle is outside of the spatial domain of interest, it is automatically assigned $\Phi = 0$. Furthermore, note that temporal summation in (2) is bounded by hard limits j_{min} and j_{max} thus ignoring $z_{i,j}$ corresponding to large values of Δt^* .

2.2 Dimensional reduction

In order to design an estimator, it is necessary to express a system in the state space formulation. Given continuous velocity field, a natural choice for a state space is to discretise every link in the network and set quantity corresponding to each spatial location to be one of the states.

Nonetheless, in a network with multiple links, the dimension of the problem can be too large for real-time applications and it is necessary to reduce the number of degrees of freedom.

The velocity field $\mathbf{u}(\mathbf{x}, t)$ is therefore defined as a sum of time-invariant structures $\Phi_i(\mathbf{x})$ such that

$$\mathbf{u}(\mathbf{x}, t) \approx \bar{\mathbf{u}}(x) + \sum_{i=1}^r a_i(t) \Phi_i(\mathbf{x}), \quad (5)$$

where the scalar coefficient $a_i(t)$ is associated with $\Phi_i(\mathbf{x})$ and describes its evolution in time. The term $\bar{\mathbf{u}}(x)$ is a time-constant part of the dynamics and can represent, for example, mean flow or steady-state solution to the underlying equations. Given (5), in order to reduce the dimension of the problem, a_i is set to be variables in state vector \mathbf{a} . Note that now the size r of $\mathbf{a} \in \mathbb{R}^r$ can be set freely allowing for drastic dimensional reduction.

There exists a large variety of possible choices for $\Phi_i(x)$ [18]. In the following analysis, $\Phi_i(x)$ will be set to the output of Principal Component Analysis (PCA) [11]. The PCA is a widely used technique which produces structures $\Phi_i(x)$ that give the optimal approximation of (5) with respect to l^2 -norm and which are orthonormal (i.e. $\Phi_i(x)^\top \Phi_j(x) = \delta_{ij}$ where δ_{ij} is the Kronecker delta). The orthonormality of $\Phi_i(x)$ is beneficial as it allows us to obtain projection coefficients $a_i(t)$ in a straightforward manner where $a_i(t) = \Phi_i(x)^\top \mathbf{u}(\mathbf{x}, t)$. Furthermore, principal components are ordered with respect to explained variance making it easy to select the r most important structures.

2.3 Dynamic model and estimator

In order to obtain the velocity field in the network, live information about PCA projection coefficients $a_i(t)$ are necessary. This can be obtained using the Kalman filter [19], which is state-space model-based estimation technique. Given imperfect dynamic model and some noisy measurements, Kalman filter can produce a state estimate which is more accurate based on a single set of measurements or integration of the dynamic model.

The estimated dynamical system is expressed in the *linear* state space formulation such that

$$\mathbf{a}_{t+1} = \mathbf{A} \mathbf{a}_t \quad (6a)$$

$$\mathbf{y}_t = \mathbf{C} \mathbf{a}_t \quad (6b)$$

where $\mathbf{a}_t = [a_0(t), \dots, a_r(t)]^\top$. Note that functions in (1) are replaced with matrices $\mathbf{A} \in \mathbb{R}^{r \times r}$, $\mathbf{C} \in \mathbb{R}^{p \times r}$. The Kalman filter works by recursively applying a two-step procedure: *prediction* and *update*. In the prediction step, the values at the next step are obtained by propagating a current state forward in time using dynamical model, i.e. by applying (6a). Subsequently, once a new measurement is available the predicted step is updated based on the difference between acquired and predicted measurements, i.e. $\mathbf{y}_t - \mathbf{C} \mathbf{a}_t$.

The Kalman filter algorithm was originally formulated for linear state dynamics and observation mappings (with nonlinear extensions such as Extended Kalman Filter [20] or Unscented Kalman Filter [21] developed later). Due to simplicity and low-computation cost, a linear form will be utilised here. The values of \mathbf{A} and \mathbf{C} are obtained using a solution to least squares problem on some training data, i.e.

$$\mathbf{A} = \arg \min_{\mathbf{A}^*} \sum_t \|\mathbf{a}_{t+1} - \mathbf{A}^* \mathbf{a}_t\|_2, \quad (7a)$$

$$\mathbf{C} = \arg \min_{\mathbf{C}^*} \sum_t \|\mathbf{y}_t - \mathbf{C}^* \mathbf{a}_t\|_2. \quad (7b)$$

Length	Minimum headway	Acceleration ability	Deceleration ability	Maximum velocity
5.0 m	2.0 m	0.8 m/s ²	4.5 m/s ²	25 m/s

Table 2: Vehicle parameters used in the simulation of Test Case 1.

Furthermore, the accuracy of Kalman filter estimates increases if the relative inaccuracies in the dynamical model and observation mapping are provided through process and observation noise covariance matrices, \mathbf{Q} and \mathbf{R} respectively. Using the same training data as in (7), \mathbf{Q} and \mathbf{R} are defined as

$$\mathbf{Q} = \mathbf{e}_p \mathbf{e}_p^\top \quad (8a)$$

$$\mathbf{R} = \mathbf{e}_m \mathbf{e}_m^\top \quad (8b)$$

where

$$\mathbf{e}_p = [\mathbf{a}_1 - \mathbf{A}\mathbf{a}_0 \quad \dots \quad \mathbf{a}_N - \mathbf{A}\mathbf{a}_{N-1}]$$

$$\mathbf{e}_m = [\mathbf{y}_0 - \mathbf{C}\mathbf{a}_0 \quad \dots \quad \mathbf{y}_N - \mathbf{C}\mathbf{a}_N]$$

and N is the size of the training data.

3 Case Studies - Cars approaching a traffic junction

The methodology outlined in Section 2 is analysed on the simple test case of the vehicles approaching a junction on a link with single lane. The analysed scenario will be referred to as Test Case 1 and corresponding SUMO network is presented in Figure 1. The investigated lane is 95.25m long and 3.20m wide. The induction loop is placed at $x = 89.25\text{m}$. The routes are generated such that the car enters the lane approximately every 10s leading to congested traffic conditions. First, it is assumed that each vehicles has identical parameters given in Table 2 and is driving according to Intelligent Driver Model (IDM) without any driver imperfections. Test Case 1 is simulated using SUMO 0.32.0 [22] for $t \leq 2\text{h}$. Data is sampled with the constant timestep of $\delta t = 0.1\text{s}$, resulting in 72,000 sets of measurements.

In order to test the robustness of the estimation approach Test Case 2 is constructed which consists of multiple separate simulations. In this case, every vehicle is assigned with a certain degree of imperfection. For desired speed $u^*(t)$ given by IDM, the speed of each vehicle is modified such that $u(t) = ku^*(t)$. The value of k is chosen using the following procedure: 1) first, k is drawn from Gaussian distribution, $k \sim \mathcal{N}(1, \sigma^2)$; 2) subsequently, in order to avoid excessively slow and fast vehicles in the network, lower and upper caps are introduced such that $0.2 < k < 2$. Separate simulations are performed for $\sigma^2 = \{i \in \mathbb{N} | i \leq 4\}$. It is noted that since the upper and lower caps are introduced, distribution of k is no longer Gaussian and as $\sigma^2 \rightarrow \infty$ it will converge to Bernoulli distribution. Consequently, σ^2 does not describe variance of k in a strict sense.

4 Results

In order to prevent over fit in the model construction, the data-set is split into two equal subsets corresponding to $t \leq 1\text{h}$ and $t > 1\text{h}$. The latter set is used to perform principal component analysis and obtain the dynamical model (see Section 4.2-4.3) whilst the first half is used to for estimation (see Section 4.4).

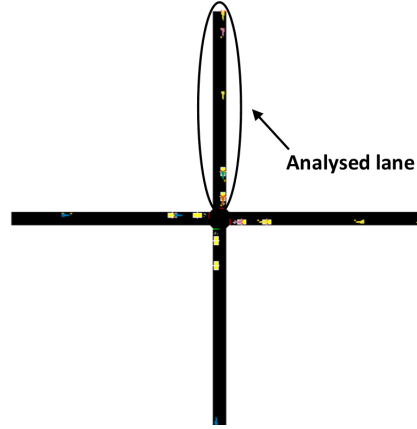


Figure 1: Test case of vehicles approaching a junction is used to analyse the methodology presented in Section 2.

4.1 Modified ASM analysis

The modified ASM methodology outlined in Section 2.1 is applied to data corresponding to all aforementioned Test Cases. The lane is discretised into 100 points and the temporal window in (2) is set such that j_{max} and j_{min} always satisfy $j_{max} - j_{min} = 10s$ and $t = \frac{1}{2}(j_{max} + j_{min})$. The value of 10s is chosen so that weight $\Phi(0, \Delta t^*)$ generated using kernel (4) associated with measurements temporally furthest away satisfies $\Phi(0, \Delta t^*) < 0.05$. The spatio-temporal diagram of obtained from the modified ASM together with temporal and spatial snapshots at fixed locations are presented in Figure 2.

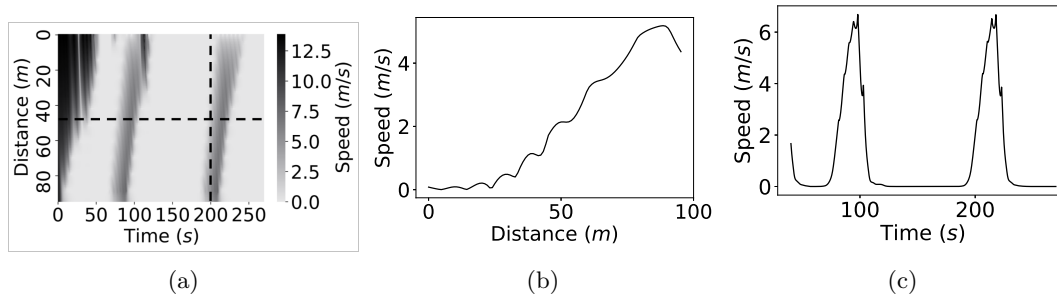


Figure 2: (a) Spatio-temporal diagram obtained from the modified Adaptive Smoothing Method, (b) spatial snapshot at $t = 200s$, and (c) temporal snapshot at $x = 48m$. Note that locations of the snapshots are represented on the spatio-temporal diagram.

Figure 2a shows that the vehicles enter the lane initially at high speeds until the lane becomes saturated, and congested traffic conditions develop. This can be observed especially well in the temporal snapshot in Figure 2c where the regular wave-like structure is visible travelling in time. The regularity of this feature implies that there are vehicles present in the lane at every change of traffic light. It is noted that associated timescale is controlled by the length of the traffic light stages. Furthermore, a waveform with a wave number corresponding to the vehicle length can be observed in the spatial snapshot in Figure 2b, implying that vehicles occupy the

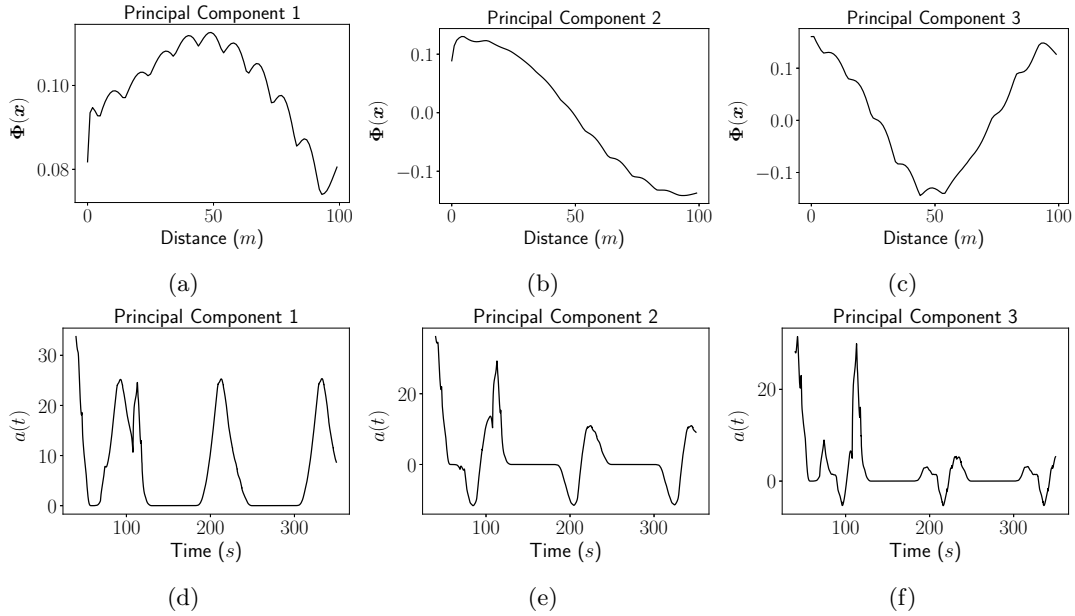


Figure 3: Three principal components of the data are presented in (a), (b) and (c), and their corresponding projection coefficients are illustrated in (d), (e) and (f) respectively.

whole length of the lane.

4.2 PCA analysis

Principal components of the data presented in Figure 2 are calculated in this section in order to obtain the separation of variable of the form (5). Note that, usually (5) contains a time-constant structure $\bar{u}(x)$. However, here, $\bar{\mathbf{u}}(\mathbf{x}) = 0$ is chosen based on the assumption that if other form of $\bar{\mathbf{u}}(\mathbf{x})$ is significant, it will emerge as $\Phi_i(\mathbf{x})$ for which $\dot{a}_i(t) \approx 0$.

The first three principal components $\Phi_i(\mathbf{x})$ and their corresponding projection coefficients $a_i(t)$ are presented in Figure 3. Although, principal component decomposition should be analysed taking into account all components at the same time, a certain level of interpretation can be achieved by investigating each component individually. The shape of the first principal component in Figure 3a demonstrates that the maximum velocity in the link is achieved in the middle of the lane. The second principal component in Figure 3b shows vehicles departing from the analysed lane in the situation where traffic light stage changes to green. When its projection coefficient becomes negative in Figure 3e, an increase in velocity downstream is observed. Subsequently, when its coefficient becomes positive, upstream velocity field increases implying that the vehicles in the back of the queue start to accelerate. It should be noted that projection coefficients exhibit progressively higher frequencies for higher orders of principal components, representing harmonics of the system.

In order to choose the appropriate size of the low-dimensional space (r in (5)) singular values s_i of the principal components are investigated. Singular values give us information regarding the explained variance σ_i^2 of each principal component such that $\sigma_i^2 = s_i^2 / \sum_i s_i^2$. The total explained variance, $\sigma^2(r) = \sum_{i=1}^r \sigma_i^2$ as a function of r is analysed and presented in Figure 4a. It can be seen that the value approaches $\sigma^2(r) \rightarrow 1$ at fast rate and $\sigma^2(r) > 0.99$ for $r \geq 6$.

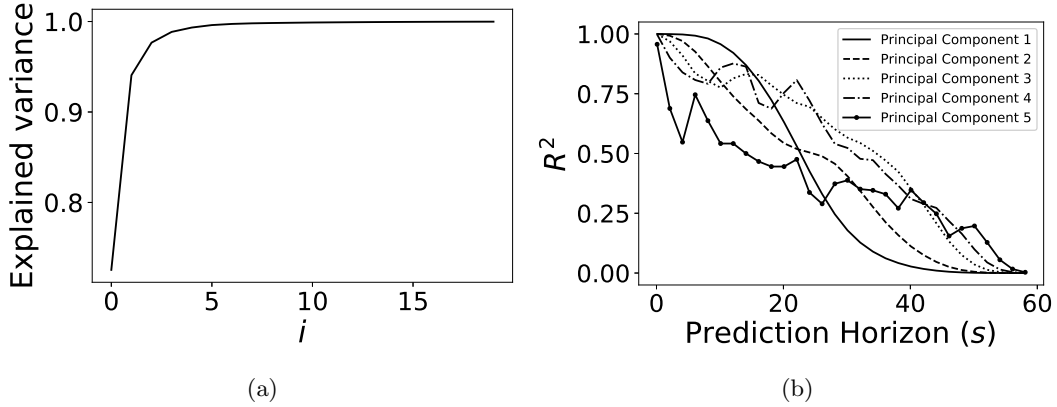


Figure 4: (a) Shows the explained variances of the principal components corresponding to the data in 2a, and (b) is the R^2 corresponding to linear model applied over a range of prediction horizons.

Consequently, $r = 6$ is chosen for further analysis.

4.3 Dynamical modelling

Linear model required in Kalman filter is obtained by solving (7a) on the training data set. Figure 4b shows the values of R^2 for each of the projection coefficients when linear model is applied in order to predict state values over a range of future prediction horizons t_h . It is showed that over a short period of time linear model provides a good description of the dynamics, where first four of the projection coefficients exhibits $R^2 > 0.6$ for $t_h > 20$ s. It can be also seen that as the order of the projection coefficients increases the relationship between R^2 and t_h becomes more irregular. This is due to the fact that PCA in many cases is able to separate noise and coherent structures into separate principal components, so that noise is primarily contained in higher order components. This feature is also visible in the analysed system where higher order projection coefficients contain high frequency oscillations.

In addition to dynamical model, it is necessary to provide measurements \mathbf{y} which will be used to update state \mathbf{a} . Using instantaneous velocity, $u(t)$ from induction loop obtained directly from SUMO simulation, time-delayed valued are introduced in the construction of \mathbf{y} such that $\mathbf{y}_t = [u(t), u(t-\delta t), \dots, u(t-n\delta t)]^\top$ for n delayed measurements. The mapping \mathbf{C} is obtained by solving (7b). Noting that measurements are not available when there are no vehicles occupying the induction loop, (7b) is only solved for t when $\|\mathbf{y}_t\|_0 = n + 1$. The value of n is chosen based on solution to the inverse problem

$$\min_{\mathbf{C}^{-1}} \sum_t \|\mathbf{C}^{-1} \mathbf{y}_t - \mathbf{a}_t\|_2$$

and investigation of R^2 corresponding to estimate of each state in \mathbf{a}_t . It was observer that relative increase in value of R^2 for $n = 6$ is less than 5% and consequently $n = 5$ is chosen. Despite omitting measurements where \mathbf{y}_t is not a full vector, $N \approx 30,000$ samples are used to solve (7b).

Given \mathbf{A} and \mathbf{C} obtained on the training data, noise covariance matrices are obtained using (8).

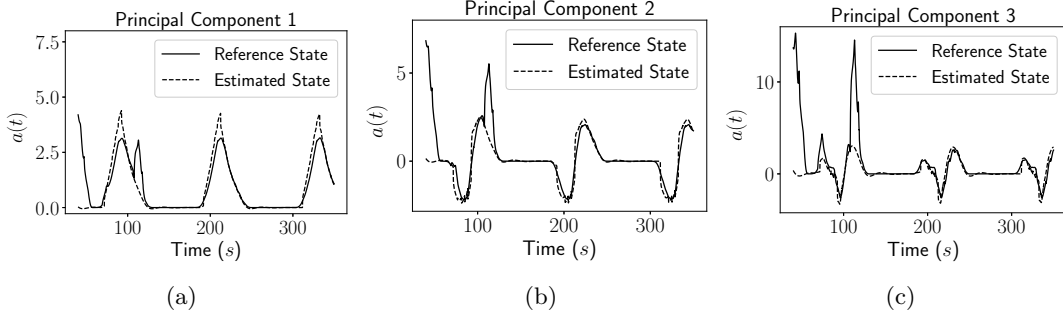


Figure 5: Estimation results for first three principal components.

Principal Component	1	2	3	4	5	6
ϵ_i	0.242	0.284	0.382	0.584	0.856	0.789

Table 3: The relative squared errors (10) corresponding to results presented in Figure 5.

4.4 Estimation results

4.4.1 Test Case 1

Using model obtained in Section 4.3 Kalman filter is applied to measurements \mathbf{y} outside the training domain, covering initial portion of the data. Standard Kalman filter is modified to take into account the fact that \mathbf{y} is not available at every timestamp. Recalling that Kalman filter consists of prediction and update stages, here, when there are no vehicles occupying the induction loop only the prediction step is applied and the update stage is omitted. The estimated coefficients together with the reference data are presented in Figure 5.

At the initial stage of the simulation vehicles enter the analysed domain for the first time, and the system is converging towards congested condition. The dynamics at that stage is highly nonlinear and such phenomena are not included in the training data. On the other hand for $t > 150$ s, the system exhibits oscillatory wave-like behaviour, where the linear Kalman filter formulation provides accurate estimates. The relative squared error,

$$\epsilon_i^2 = \frac{\sum_{t>150} (a_i(t) - \hat{a}_i(t))^2}{\sum_{t>150} (a_i(t))^2} \quad (10)$$

is calculated for each of the estimated projection coefficients \hat{a}_i with the values shown in Table 3. It is observed that dynamics of the first three principal components is estimated accurately with $\epsilon_i < 0.5$. The errors are larger for components of the higher order which correspond to frequencies higher than those which characterise the induction loop measurement. In those cases nonlinear mapping between state and measurement might be necessary. Nonetheless, it is noted that the contribution of the higher order components to explained variance is relatively small (see Figure 4a) and the estimate errors should have significantly smaller impact on the accuracy of the reconstructed velocity field $\hat{\mathbf{u}}(x, t)$.

Although analysis of (10) gives a beneficial insight into the accuracy of the estimation, the accuracy of the reconstructed velocity field $\hat{\mathbf{u}}(t)$ and its similarity to actual velocity profiles of the vehicles in the network should be used as the key performance metric. Consequently the following discrete error metric is defined. Given $\hat{\mathbf{u}}(x, t)$ and set of 3-tuples (v_i, x_i, t_i) containing respectively velocity and location of the vehicle and time at which measurement is obtained,

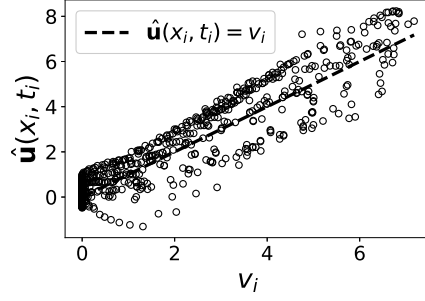


Figure 6: The scatter plot of velocities v_i and corresponding reconstructed velocity field $\hat{\mathbf{u}}(x_i, t_i)$.

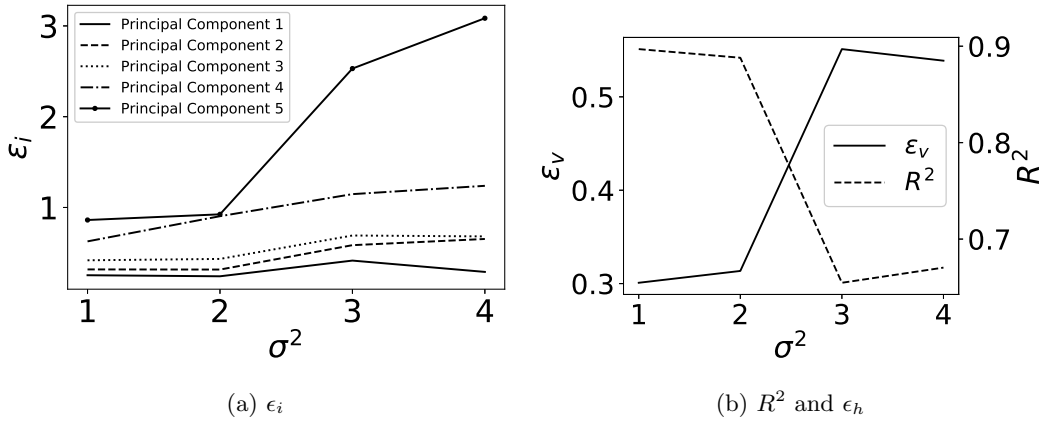


Figure 7: The error metrics for estimation framework applied to data with imperfect drivers for a range of σ^2 .

the error metric is given by

$$\epsilon_v^2 = \frac{\sum_i (\hat{\mathbf{u}}(x_i, t_i) - v_i)^2}{\sum_i v_i^2}. \quad (11)$$

It is noted that the same vehicle is used multiple times in the error calculation but for different values of x_i and t_i .

The scatter plot of velocities v_i and corresponding reconstructed velocity field $\hat{\mathbf{u}}(x_i, t_i)$ for $t > 150$ s is given in Figure 6. Note that, for the visualisation purposes, data in Figure 6 was subsampled and only randomly chosen 10% of all datapoints are presented. It is shown that the on average estimated velocity is in the same range as the reference value. The values of ϵ_v and R^2 associated with results in Figure 6 are 0.286 and 0.907 respectively.

4.4.2 Test Case 2

The results in Section 4.4.1 were obtained based on an idealistic scenario where every driver perfectly follows the IDM car-following model. The framework designed in Section 4.3 is subsequently applied to more realistic Test Case 2. Recall that in this case each driver in the SUMO simulation is associated with imperfection factor chosen using methodology described in Section 3. Four scenarios are analysed for different values of σ^2 . The equivalent error metrics,

i.e. error in estimation coefficients ϵ_i (10) and error against discrete measurements ϵ_v (11), are presented in Figure 7. It is shown that the estimation of the first three projection coefficients is robust with respect to σ^2 . On the other hand, higher order estimates give error of $\epsilon_i > 1$ for $\sigma^2 > 2$. Such values of ϵ_i implies that estimate of the coefficients is on average worse than a constant model, and estimation of those dynamical features in fact reduces accuracy of the reconstructed velocity field. This phenomenon is reflected in the rapid decrease of R^2 in Figure 7b.

5 Conclusions

The methodology which estimates velocity profiles based on the real-time information from induction loops was presented. The current implementation is linear and low-dimensional making the demonstrated approach suitable in situations where computational cost and simplicity are of high importance. The system was analysed on a simple problem of vehicles approaching the traffic junction in a congested traffic. Such a system exhibits semi-linear dynamics and can be well estimated using projection onto few structures and linear formulation of Kalman filter. Additionally, model calculated based on a perfect data was applied to dynamics with driver imperfections, where the estimation of key dynamical features corresponding to first few principal components was shown to be robust.

The future work will involve the application of the developed methodology to more realistic test cases; first with smaller and variable levels of congestion and second with traffic perturbations. In those cases non-linear effects are expected to be of significance and will need to be implemented in the dynamical model (1). The exact form of nonlinearity will be either obtained from detailed analysis of car-following models or from data-driven model inference techniques. Additionally, the network level test case will be developed and examined, because it is believed that proposed technique is more suitable in systems with multiple road link since it allows for radical reduction of degrees of freedom.

Acknowledgement

This research is funded by the Engineering and Physical Sciences Research Council (EPSRC) [grant No.: EP/N022262/1] whose support is gratefully acknowledged. The authors also express appreciation to the SUMO community for the development and maintenance of the SUMO micro-simulation software package, and to Craig Rafter from the University of Southampton for his assistance and support to get familiar with SUMO.

References

- [1] Carlos E. Garcia, David M. Prett, and Manfred Morari. Model predictive control: theory and practice—a survey. *Automatica*, 25(3):335–348, 1989.
- [2] Mark S. Young, Stewart A. Birrell, and Neville A. Stanton. Safe driving in a green world: A review of driver performance benchmarks and technologies to support ‘smart’driving. *Applied ergonomics*, 42(4):533–539, 2011.
- [3] Frank Lai, Oliver Carsten, and Fergus Tate. How much benefit does intelligent speed adaptation deliver: An analysis of its potential contribution to safety and environment. *Accident Analysis & Prevention*, 48:63–72, 2012.

- [4] Rich C. McIlroy, Neville A. Stanton, and Catherine Harvey. Getting drivers to do the right thing: a review of the potential for safely reducing energy consumption through design. *IET Intelligent Transport Systems*, 8(4):388–397, 2013.
- [5] Eleni I. Vlahogianni, Matthew G. Karlaftis, and John C. Golias. Short-term traffic forecasting: Where we are and where we’re going. *Transportation Research Part C: Emerging Technologies*, 43:3–19, 2014.
- [6] Martin Treiber and Dirk Helbing. An adaptive smoothing method for traffic state identification from incomplete information. In *Interface and Transport Dynamics*, pages 343–360. Springer, 2003.
- [7] J.W.C. Van Lint and Serge P. Hoogendoorn. A robust and efficient method for fusing heterogeneous data from traffic sensors on freeways. *Computer-Aided Civil and Infrastructure Engineering*, 25(8):596–612, 2010.
- [8] Martin Treiber, Arne Kesting, and R. Eddie Wilson. Reconstructing the traffic state by fusion of heterogeneous data. *Computer-Aided Civil and Infrastructure Engineering*, 26(6):408–419, 2011.
- [9] Felix Rempe, Philipp Franek, Ulrich Fastenrath, and Klaus Bogenberger. Online freeway traffic estimation with real floating car data. In *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, pages 1838–1843. IEEE, 2016.
- [10] Zhoutong Jiang, Xiqun Michael Chen, and Yanfeng Ouyang. Traffic state and emission estimation for urban expressways based on heterogeneous data. *Transportation Research Part D: Transport and Environment*, 53:440–453, 2017.
- [11] Ian Jolliffe. Principal component analysis. In *International encyclopedia of statistical science*, pages 1094–1096. Springer, 2011.
- [12] Michael E. Tipping and Christopher M. Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 61(3):611–622, 1999.
- [13] Theodore Tsekeris and Antony Stathopoulos. Measuring variability in urban traffic flow by use of principal component analysis. *Journal of Transportation and Statistics*, 9(1):49, 2006.
- [14] Qu Li, Hu Jianming, and Zhang Yi. A flow volumes data compression approach for traffic network based on principal component analysis. In *2007 IEEE Intelligent Transportation Systems Conference*, pages 125–130. IEEE, 2007.
- [15] Xuexiang Jin, Yi Zhang, Li Li, and Jianming Hu. Robust pca-based abnormal traffic flow pattern isolation and loop detector fault detection. *Tsinghua Science & Technology*, 13(6):829–835, 2008.
- [16] Xingxing Xing, Xiabing Zhou, Haikun Hong, Wenhao Huang, Kaigui Bian, and Kunqing Xie. Traffic flow decomposition and prediction based on robust principal component analysis. In *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, pages 2219–2224. IEEE, 2015.
- [17] Chenyi Chen, Yin Wang, Li Li, Jianming Hu, and Zuo Zhang. The retrieval of intra-day trend and its influence on traffic prediction. *Transportation research part C: emerging technologies*, 22:103–118, 2012.
- [18] Laurens Van Der Maaten, Eric Postma, and Jaap Van den Herik. Dimensionality reduction: a comparative review. *J Mach Learn Res*, 10:66–71, 2009.
- [19] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82(1):35–45, 1960.
- [20] Harold W/ Sorenson and Allen R. Stubberbud. Non-linear filtering by approximation of the a posteriori density. *International Journal of Control*, 8(1):33–51, 1968.
- [21] Eric A. Wan and Rudolph Van Der Merwe. The unscented kalman filter for nonlinear estimation. In *Adaptive Systems for Signal Processing, Communications, and Control Symposium 2000. ASSPCC. The IEEE 2000*, pages 153–158. Ieee, 2000.
- [22] Pablo Alvarez Lopez, Michael Behrisch, Laura Bieker-Walz, Jakob Erdmann, Yun-Pang Flötteröd, Robert Hilbrich, Leonhard Lücken, Johannes Rummel, Peter Wagner, and Evamarie Wießner. Microscopic traffic simulation using sumo. In *The 21st IEEE International Conference on Intelligent*

Transportation Systems. IEEE, 2018.